



VIGO UNIVERSITY
DEPARTMENT OF ELECTRONICS TECHNOLOGY

DISSERTATION SUBMITTED FOR THE DEGREE OF
EUROPEAN DOCTOR OF
PHILOSOPHY IN ELECTRICAL ENGINEERING

SPACE VECTOR PULSE-WIDTH MODULATION FOR
MULTILEVEL MULTIPHASE VOLTAGE-SOURCE
CONVERTERS

Author: Óscar López Sánchez

Director: Jesús Doval Gandoy

Vigo, 9 January 2009

PH.D. COMMITTEE MEMBERS:

Carlos Martínez Peñalver

Emil Levi

Frede Blaabjerg

Iustin Radu Bojoi

Josep Pou i Félix

Keith A. Corzine

Leopoldo García Franquelo

Manuel Pérez Donsión

Mariusz Malinowski

Olorunfemi Ojo

9 January 2009

University of Vigo

Vigo, Spain

Vita

Óscar López Sánchez was born in A Coruña, Spain, in 1975. He received the M.Sc. degree in Mechanical Engineering and the M.Sc. degree in Electrical Engineering in 2001 and 2004, respectively, from the University of Vigo, Spain.

From October 2003 to September 2004, he was Assistant Professor with the University Carlos III of Madrid, Spain. He is currently an Assistant Professor with the University of Vigo. From March 2006 to June 2006, he had a research stay at the Institute of Energy Technology in the University of Aalborg in Denmark. From September 2008 to December 2008, he joined the RCEEE Electric Machines and Drives group of Liverpool John Moores University in the United Kingdom for a research stay. His research interests include control of multilevel power converters and renewable energy systems. He has authored more than 20 published technical papers in the fields of power electronics and systems.

He is member of the IEEE Industrial Electronics Society since 2005.

Space Vector Pulse-Width Modulation for Multilevel Multiphase Voltage-Source Converters

Óscar López Sánchez

(ABSTRACT)

The space vector pulse-width modulation (SVPWM) for multilevel multiphase power converters is studied in this dissertation. Multilevel converters have a high voltage capability with voltage limited devices, high efficiency, low harmonic distortion and good electromagnetic compatibility. Multiphase machines have a high reliability, high efficiency and low torque ripple with a low per-phase power rating. The multilevel multiphase technology combines the benefits of both technologies, but new modulation techniques must be developed to take advantage of multilevel multiphase converters.

The SVPWM technique has been deeply studied for multilevel converters due to its performance benefits when compared with other modulation techniques. However, even though it has a general acceptance in three-phase applications it has hardly been studied in relation to multiphase systems. Most of multiphase SVPWM algorithms presented in literature deal with two-level converters. Multilevel multiphase SVPWM algorithms are rare and most of them have been developed for a particular number of phases or levels, and it is difficult to extend them to other cases. There are some generic algorithms available but they are complex and not suitable for real-time implementation. The main objective of this work is to develop a generic SVPWM algorithm for multiphase converters, valid for any number of levels and phases that can be implemented in real-time. The main contributions are summarized in the following.

Research starts with the study of the relationships between the switching states of the transistors and the output voltage in the typical multilevel topologies: diode clamped, flying capacitor and cascaded full-bridge converters. Some common characteristics have been identified in those relationships, which allow formulating the SVPWM problem in a generic form for all these topologies. Novel expressions to calculate the number of redundant switching states corresponding to the same output level are also provided.

Next, two well-known multilevel three-phase SVPWM techniques, one with and another without switching state redundancy, are studied to identify the key issues in the implementation of SVPWM algorithms. The most significant are two: the high number of power switches that must be controlled in parallel in multilevel topologies; and the extra modulation complexity due to the switching state redundancy present in converters with no neutral wire. Redundancy can be used to achieve certain control goals such as modulation index extension, capacitor voltage balancing, dc source current control, switching frequency reduction, fault tolerance or common-mode voltage elimination. Field-programmable gate arrays (FPGAs), with their concurrent processing capability, are identified as appropriate devices for implementing multilevel modulation algorithms.

A new modular design of the hardware implementation and a comparative analysis of both modulation algorithms are carried out.

Finally, two novel SVPWM algorithms for multilevel multiphase voltage source converters, with and without switching state redundancy, are developed. Both are formulated in a multidimensional space and can be applied to converters with any number of levels and phases. It is demonstrated, in both cases, that the multilevel SVPWM can be carried out by means of a two-level space vector modulator without redundancy. Therefore, a new two-level multiphase SVPWM algorithm is developed. In the case of a multilevel P -phase converters without redundancy, the SVPWM algorithm is based on a displacement plus the new two-level SVPWM for P -phases. In the case, of a multilevel P -phase converter with switching state redundancy the SVPWM is based on a more complex transformation and the new two-level SVPWM algorithm for $(P - 1)$ -phases. This algorithm includes a new technique that makes it easy to choose the switching states among the set of redundant switching states. Both modulation algorithms can be applied to the typical multilevel topologies; they have a low computational complexity and they are suitable for hardware implementation. The new SVPWM algorithms are implemented in a low-cost FPGA and they are tested in laboratory with a five-level five-phase inverter. The modulation techniques are also applied to multilevel three-phase converters with three and four legs. The resulting algorithms are compared with existing SVPWM techniques and tested in laboratory.

Major contributions of this dissertation have been published in four journal papers, and minor contributions have been presented at two international conferences.

Acknowledgments

First of all, I would like to thank my advisor, Dr. Jesús Doval, for his trust, guidance and support. He introduced me to the interesting field of power electronics.

I would like to gratefully acknowledge Dr. Jacobo Álvarez who carried out the VHDL description of the modulation algorithms presented in this dissertation. Without his help this work would not be possible.

I would like to express my appreciation to my colleagues of the Electronics Technology Department of the Vigo University. In particular, I would like to thank Dr. Lucía Costas and Francisco Freijedo for their continued support and good time spent with them.

I would like to express my gratitude to Prof. Frede Blaabjerg for giving me the opportunity to be at the Institute of Energy Technology of the Aalborg University. I wish to thank Dr. Remus Teodorescu for his help. I enjoyed and learned a lot by attending his courses.

I would like to thank Prof. Emil Levi for allowing me to join to the RCEEE Electric Machines and Drives research group of the Liverpool John Moores University for a research stay.

I would like to thank the company *LETAG, S.A.* for collaborating in the design and construction of the ac machine used in tests.

I would like to express my gratitude to members of the Ph.D. Committee for their unselfish collaboration in the evaluation of my work.

I would like to thank my parents and my brother for everything that they have done for me.

Finally, I would like to dedicate this dissertation to Marta for her love, unconditional support and continued encouragement.

*A Marta,
a miña pequena*

Contents

| | |
|--|------------|
| Contents | i |
| List of Figures | v |
| List of Tables | vii |
| List of Abbreviations and Acronyms | ix |
| Nomenclature | xi |
| 1 Introduction | 1 |
| 1.1 Motivation and Objectives | 1 |
| 1.2 Review of Previous Research | 2 |
| 1.2.1 Multilevel Technology | 2 |
| 1.2.2 Multiphase Motor Drives | 4 |
| 1.2.3 Multilevel Multiphase Voltage Source Inverters | 6 |
| 1.2.4 Multilevel Multiphase PWM | 7 |
| 1.3 Major Results | 11 |
| 2 Per-Phase Redundancy in Multilevel Converters | 13 |
| 2.1 Introduction | 13 |
| 2.2 Diode-Clamped Converter | 14 |
| 2.2.1 Switching State Redundancy | 16 |
| 2.2.2 Characteristics | 16 |
| 2.3 Flying-Capacitor Converter | 17 |
| 2.3.1 Switching State Redundancy | 19 |
| 2.3.2 Characteristics | 20 |
| 2.4 Cascaded Full-Bridge Converter | 20 |
| 2.4.1 Switching State Redundancy | 22 |
| 2.4.2 Characteristics | 23 |
| 2.5 Redundant Switching State Selection | 24 |
| 2.6 Conclusion | 29 |
| 3 Implementation of Multilevel SVPWM Algorithms | 31 |
| 3.1 Introduction | 31 |
| 3.2 Space Vector PWM Algorithms | 32 |
| 3.3 Implementation Design | 34 |

| | | |
|----------|--|-----------|
| 3.3.1 | Specifications | 34 |
| 3.3.2 | Description of the 2D SVPWM Algorithm Components | 35 |
| 3.3.3 | Description of the 3D SVPWM Algorithm Components | 38 |
| 3.4 | Hardware Implementation | 39 |
| 3.4.1 | Component Implementation | 39 |
| 3.4.2 | Resources | 44 |
| 3.5 | Experimental Results | 44 |
| 3.6 | Conclusion | 46 |
| 4 | Multilevel Multiphase SVPWM Algorithm | 49 |
| 4.1 | Introduction | 49 |
| 4.2 | Algorithm Development | 50 |
| 4.2.1 | Algorithm Formulation | 50 |
| 4.2.2 | Algorithm Decomposition | 52 |
| 4.2.3 | Two-level Multiphase SVPWM Algorithm | 53 |
| 4.2.4 | Multilevel Multiphase SVPWM Algorithm | 56 |
| 4.2.5 | Example | 57 |
| 4.2.6 | Algorithm Features | 59 |
| 4.3 | Algorithm Hardware Implementation | 60 |
| 4.4 | Experimental Results | 61 |
| 4.5 | Application to Three-Phase Three-Leg Converters | 63 |
| 4.5.1 | 3D SVPWM Algorithm | 63 |
| 4.5.2 | Example | 68 |
| 4.5.3 | Comparison With Existing 3D Algorithm | 70 |
| 4.5.4 | Experimental Results | 71 |
| 4.6 | Application to Three-Phase Four-Leg Converters | 74 |
| 4.6.1 | 4D SVPWM Algorithm | 74 |
| 4.6.2 | Example | 79 |
| 4.6.3 | Experimental Results | 84 |
| 4.7 | Conclusion | 87 |
| 5 | Multilevel Multiphase SVPWM Algorithm With Switching State Redundancy | 91 |
| 5.1 | Introduction | 91 |
| 5.2 | Algorithm Development | 92 |
| 5.2.1 | Algorithm Formulation | 92 |
| 5.2.2 | Equivalent Modulation Law | 94 |
| 5.2.3 | Modulation Problem Solution | 96 |
| 5.2.4 | Selection of the Vectors of the Switching Sequence | 97 |
| 5.2.5 | New Multilevel Multiphase SVPWM Algorithm | 101 |
| 5.2.6 | Example | 102 |
| 5.2.7 | Algorithm Features | 104 |
| 5.3 | Hardware Implementation | 106 |
| 5.4 | Experimental Results | 107 |
| 5.5 | Application to Three-Phase Three-Leg Converters | 111 |
| 5.5.1 | 2D SVPWM Algorithm With State Redundancy | 111 |

| | | |
|----------|--|------------|
| 5.5.2 | Example | 114 |
| 5.5.3 | Comparison With Existing 2D Algorithm | 116 |
| 5.5.4 | Experimental Results | 118 |
| 5.6 | Application to Three-Phase Four-Leg Converters | 119 |
| 5.6.1 | 3D SVPWM Algorithm With State Redundancy | 119 |
| 5.6.2 | Example | 122 |
| 5.6.3 | Comparison With Existing 3D Algorithm | 124 |
| 5.6.4 | Experimental Results | 125 |
| 5.7 | Conclusion | 129 |
| 6 | Conclusion and Future Research | 131 |
| 6.1 | Conclusion | 131 |
| 6.2 | Future Research | 133 |
| | References | 135 |
| A | Further Information About Testing Facilities | 151 |
| A.1 | dSPACE Platform | 151 |
| A.2 | Field-Programmable Gate Array | 153 |
| A.3 | Optical Link | 154 |
| A.4 | Inverters | 155 |
| | A.4.1 Three-Leg Inverter | 155 |
| | A.4.2 Four- and Five-Leg Inverters | 156 |
| A.5 | Loads | 156 |
| A.6 | Measurements | 157 |

List of Figures

| | | |
|------|---|----|
| 2.1 | Five-level diode-clamped converter. | 14 |
| 2.2 | Five-level flying capacitor converter. | 18 |
| 2.3 | Five-level cascaded full-bridge converter. | 21 |
| 2.4 | Trigger generator adaptation between flying capacitor and cascaded full-bridge topologies. | 25 |
| 2.5 | Trigger generator for diode-clamped converter used with other topologies. | 27 |
| 2.6 | Simulation results of using a trigger generator for a diode-clamped converter with a flying capacitor and a cascaded full bridge converter. | 28 |
| | | |
| 3.1 | Control regions. | 34 |
| 3.2 | Three-phase NPC inverter. | 35 |
| 3.3 | Implementation diagrams of the SVPWM algorithms. | 36 |
| 3.4 | Switching vector sequence selection. | 37 |
| 3.5 | Dead time implementation. | 38 |
| 3.6 | Component <code>freq</code> | 40 |
| 3.7 | Simulink implementation of the transformation components. | 41 |
| 3.8 | Flow diagram of the VHDL description of the modulation components. | 42 |
| 3.9 | Flow diagram of the VHDL description of the <code>gh2abc</code> component. | 42 |
| 3.10 | Implementation diagram of the component <code>PWM4</code> | 43 |
| 3.11 | Component <code>dead_time</code> description. | 44 |
| 3.12 | Trigger signals. | 45 |
| 3.13 | Experimental test setup. | 47 |
| 3.14 | Experimental results. | 47 |
| 3.15 | Neutral voltage. | 48 |
| | | |
| 4.1 | Functional diagram of a multilevel multiphase converter. | 52 |
| 4.2 | Example of decomposition in the 2D problem. | 54 |
| 4.3 | Block diagram of the multilevel multiphase SVPWM. | 54 |
| 4.4 | Block diagram of the two-level multiphase SVPWM. | 56 |
| 4.5 | Algorithm flow chart. | 57 |
| 4.6 | Five-level five-phase cascaded full-bridge inverter. | 60 |
| 4.7 | Simulink model of the hardware implementation of the SVPWM algorithm. | 61 |
| 4.8 | Experimental test setup. | 62 |
| 4.9 | Comparison of experimental measurements with simulation results. | 64 |
| 4.10 | Experimental results. | 65 |

| | | |
|------|---|-----|
| 4.11 | Trajectories of the output voltage and the current vectors in stationary dq axes. | 66 |
| 4.12 | 3D algorithms comparison. | 72 |
| 4.13 | Trigger signals. | 73 |
| 4.14 | Experimental test setup. | 74 |
| 4.15 | Inverter output voltage. | 75 |
| 4.16 | Functional diagram of multilevel four-leg converter. | 75 |
| 4.17 | Simulation results. | 85 |
| 4.18 | Trigger signals. | 86 |
| 4.19 | Four-leg five-level cascaded full-bridge inverter. | 86 |
| 4.20 | Experimental test setup. | 87 |
| 4.21 | Inverter output voltage. | 88 |
| 4.22 | Low order harmonics of phase voltage. | 88 |
| 4.23 | Spectrum of phase voltage. | 89 |
| 5.1 | Multiphase system without neutral conductor. | 93 |
| 5.2 | Two vectors with the same projection onto different planes. | 95 |
| 5.3 | All switching vectors arranged in a table. | 98 |
| 5.4 | Block diagram of the new multilevel multiphase SVPWM algorithm. | 101 |
| 5.5 | Five-level five-phase cascaded full-bridge inverter. | 106 |
| 5.6 | Simulink model of the hardware implementation of the SVPWM algorithm. | 106 |
| 5.7 | Experimental test setup. | 108 |
| 5.8 | Comparison of experimental measurements with simulation results. | 109 |
| 5.9 | Experimental results. | 109 |
| 5.10 | Experimental results. | 110 |
| 5.11 | Trajectories of the output voltage and the current vectors in stationary dq axes. | 111 |
| 5.12 | 2D algorithms comparison in the gh space. | 117 |
| 5.13 | Trigger signals. | 118 |
| 5.14 | Inverter output voltage. | 119 |
| 5.15 | 3D algorithms comparison. | 126 |
| 5.16 | Simulation results. | 127 |
| 5.17 | Trigger signals. | 128 |
| 5.18 | Inverter output voltage. | 130 |
| A.1 | Block diagram of the setups used for testing the algorithms. | 151 |
| A.2 | S3 board with the TTL/LVTTL adaptation boards connected. | 154 |
| A.3 | Block diagram of the circuits implemented in the FPGA. | 154 |
| A.4 | Five-vector sequence example. | 155 |
| A.5 | Full-bridge module. | 156 |
| A.6 | Three-phase motor. | 157 |
| A.7 | Five-phase motor. | 157 |

List of Tables

| | | |
|-----|--|-----|
| 2.1 | Five-level diode-clamped converter relationships. | 16 |
| 2.2 | Five-level flying capacitor converter relationships. | 19 |
| 2.3 | Number of per-phase redundant switching states in multilevel flying capacitor converters. | 20 |
| 2.4 | Five-level cascaded full-bridge converter relationships. | 22 |
| 2.5 | Number of per-phase redundant switching states in multilevel cascaded full-bridge converters. | 23 |
| 2.6 | Equivalence between the switching states of the flying capacitor converter and the cascaded full-bridge converter. | 26 |
| 3.1 | Resources summary. | 46 |
| 4.1 | Resources summary. | 62 |
| 4.2 | Permutation matrix in the 3D SVPWM algorithm. | 67 |
| 4.3 | Vector sequence and switching times of the 3D SVPWM algorithm. | 69 |
| 4.4 | Resources summary. | 73 |
| 4.5 | Permutation matrix in the 4D SVPWM algorithm. | 77 |
| 4.6 | Vector sequence and switching times of the 4D SVPWM algorithm. | 80 |
| 4.7 | Resources summary. | 84 |
| 5.1 | Switching vector string of the example. | 105 |
| 5.2 | Resources summary. | 107 |
| 5.3 | Permutation matrix in the 2D SVPWM algorithm. | 112 |
| 5.4 | Displaced space vector sequence and dwell times of the 2D SVPWM algorithm. | 113 |
| 5.5 | Resources summary. | 118 |
| 5.6 | Displaced space vector sequence and dwell times of the 3D SVPWM algorithm. | 121 |
| 5.7 | Resources summary. | 128 |

List of Abbreviations and Acronyms

| | |
|------------------|---|
| 2D | two-dimensional |
| 3D | three-dimensional |
| 4D | four-dimensional |
| 5D | five-dimensional |
| ADC | analog-to-digital converter |
| BRAM | block random access memory |
| CAN | controller-area network |
| DSP | digital signal processor |
| FPGA | field-programmable gate array |
| GCLK | global clock |
| GTO | gate turn-off thyristor |
| IEEE | Institute of Electrical and Electronics Engineers |
| IGBT | insulated-gate bipolar transistor |
| IGCT | integrated gate-commutated thyristor |
| IOB | input/output block |
| I/O | input/output |
| LED | light-emitting-diode |
| LUT | lookup table |
| LVTTL | low-voltage transistor-transistor logic |
| MULT18X18 | 18 × 18 bit multipliers |
| NPC | neutral-point clamped |
| PC | personal computer |

| | |
|----------------|--|
| PLD | programmable logic device |
| PWM | pulse-width modulation |
| RAM | random access memory |
| RCEEE | Research Centre for Electrical and Electronic Engineering |
| STATCOM | static compensator |
| SVC | static VAr compensator |
| SPWM | carrier-based sinusoidal pulse-width modulation |
| SVPWM | space vector pulse-width modulation |
| THD | total harmonic distortion |
| TTL | transistor-transistor logic |
| UPFC | unified power-flow controller |
| VHDL | very-high-speed integrated circuit hardware description language |
| VSI | voltage-source inverter |

Nomenclature

| | |
|-----------------|---|
| $\tilde{\pi}$ | Plane in which all vectors do not have homopolar component. |
| $\tilde{\pi}$ | Plane of projection. |
| τ_j | Dwell time corresponding to the space vector ω_{sj} . |
| ω_{dj} | j th vector in the displaced space vector sequence. |
| ω_{dj}^k | k component of vector ω_{dj} . |
| ω_f | Fractional part of the reference space vector ω_r . |
| ω_f^k | k component of space vector ω_f . |
| ω_i | Integer part of the reference space vector ω_r . |
| ω_i^k | k component of space vector ω_i . |
| ω_r | Reference space vector. |
| ω_r^k | k component of the reference space vector ω_r . |
| ω_{sj} | j th vector in the space vector sequence. |
| ω_{sj}^k | k component of space vector ω_{sj} . |
| θ | Angle formed by the d -axis of the rotating frame dq and the g -axis of the static frame gh . |
| A | Sinusoidal voltage amplitude. |
| B^k | Cell number of phase k in the cascaded full-bridge topology. |
| C_w | Binary word that gathers many C_{xy} bits. |
| C_{xy} | Bit result of comparing v_f^x with v_f^y . |
| D | Coefficient matrix of the two-level switching law. |
| \hat{D} | Ordered coefficient matrix of the two-level switching law, i.e., upper triangular matrix. |

| | |
|-----------------------------|---|
| I_s^k | Output current of phase k of the converter. |
| j | Position in the switching sequence. |
| k | Phase number. |
| l | Length of the switching vector sequence. |
| m | Modulation index. |
| m_1 | Normalized amplitude of the fundamental. |
| m_3 | Normalized amplitude of the third harmonic. |
| N | Number of levels of the multilevel converter. |
| N^k | Number of output levels of phase k of the multilevel converter. |
| N_{\max}^k | Maximum achievable output level by the phase k of the multilevel converter. |
| N_{\min}^k | Minimum achievable output level by the phase k of the multilevel converter. |
| P | Converter phase number. |
| \mathbf{P} | Permutation matrix that puts the elements of vector \mathbf{v}_f in descending order. |
| $\tilde{\mathbf{P}}$ | Projection matrix onto the plane $\tilde{\pi}$ along the direction of \mathbf{u} , i.e., matrix to calculate the non homopolar component of a vector. |
| $\check{\mathbf{P}}$ | Projection matrix onto the plane $\check{\pi}$ along the direction of \mathbf{u} . |
| q | Index that identifies every vector in a switching string. |
| q_{\max} | Maximum q index in the range of indices corresponding to the switching vectors that the converter is able to generate. |
| q_{\max}^k | Maximum q index in the range of indices corresponding to the switching vectors in which the phase k can be generated by the converter. |
| q_{\min} | Minimum q index in the range of indices corresponding to the switching vectors that the converter is able to generate. |
| q_{\min}^k | Minimum q index in the range of indices corresponding to the switching vectors in which the phase k can be generated by the converter. |
| $R_d(v_s^k)$ | Number of per-phase redundant states corresponding to the output level v_s^k . |
| $S_{w_{i \rightarrow j}}^k$ | Number of switchings needed to change the switching state of the converter from the level $v_{s_i}^k$ to the level $v_{s_j}^k$. |
| t | Time. |

| | |
|----------------------|---|
| \mathbf{t} | Vector of switching times. |
| T | Switching period. |
| T_d | Trigger signal dead time. |
| t_j | Normalized switching time corresponding to the switching vector \mathbf{v}_{sj} . |
| T_j | Switching time corresponding to the switching vector \mathbf{v}_{sj} . |
| T_i^k | Trigger signal corresponding to the transistor i in the phase k of diode-clamped and flying capacitor topologies. |
| \bar{T}_i^k | Trigger signal corresponding to the complementary transistor of T_i^k . |
| \mathbf{T}_v | Matrix to calculate switching vectors from space vectors. |
| \mathbf{T}_ω | Matrix to calculate space vectors from switching vectors. |
| \mathbf{u} | Projection vector. |
| V_{C_i} | Voltage of capacitor i in a diode-clamped topology. |
| $V_{C_i^k}$ | Voltage of capacitor i in phase k of a flying capacitor topology. |
| V_{dc} | Voltage step of a multilevel converter. |
| V_{DC} | Voltage of the full dc link in diode-clamped and flying capacitor topologies. |
| $V_{dc_i}^k$ | Voltage of the dc source i in phase k of a cascaded full-bridge topology. |
| \mathbf{v}_{dj} | j th vector of the displaced switching vector sequence. |
| v_{dj}^k | k component of vector \mathbf{v}_{dj} . |
| \mathbf{v}_f | Fractional part of the normalized reference vector \mathbf{v}_r . |
| $\hat{\mathbf{v}}_f$ | Vector that results by sorting the components of vector \mathbf{v}_f in descending order. |
| v_f^k | k component of vector \mathbf{v}_f . |
| V_{fund} | Peak fundamental of the output voltage. |
| \mathbf{v}_i | Integer part of the normalized reference vector \mathbf{v}_r . |
| v_i^k | k component of vector \mathbf{v}_i . |
| V_n | Voltage of the neutral point of the load. |
| \mathbf{v}_r | Normalized reference voltage vector. |

| | |
|---------------------------|---|
| $\check{\mathbf{v}}_r$ | Projection onto the plane $\check{\pi}$ of the normalized reference vector \mathbf{v}_r . |
| \mathbf{V}_r | Reference voltage vector. |
| $\tilde{\mathbf{V}}_r$ | Non homopolar component of the reference voltage vector \mathbf{V}_r . |
| v_r^d | d component of the normalized reference vector in the rotating dq frame. |
| v_r^g | g component of the normalized reference vector in the static gh frame. |
| V_r^g | g component of the reference vector in the static gh frame. |
| v_r^h | h component of the normalized reference vector in the static gh frame. |
| V_r^h | h component of the reference vector in the static gh frame. |
| v_r^k | k component of reference vector \mathbf{v}_r . |
| \tilde{v}_r^k | k component of vector $\tilde{\mathbf{v}}_r$. |
| \check{v}_r^k | k component of vector $\check{\mathbf{v}}_r$. |
| V_r^k | k component of the reference vector \mathbf{V}_r , i.e., reference voltage of phase k . |
| v_r^q | q component of the normalized reference vector in the rotating dq frame. |
| \mathbf{v}_s | Normalized converter output voltage vector. |
| $\tilde{\mathbf{v}}_s$ | Non homopolar component of the output vector \mathbf{v}_s . |
| $\check{\mathbf{v}}_s$ | Projection onto plane $\check{\pi}$ of the output vector \mathbf{v}_s . |
| \mathbf{V}_s | Converter output voltage vector. |
| $\bar{\mathbf{V}}_s$ | Homopolar component of the output voltage vector \mathbf{V}_s . |
| $\tilde{\mathbf{V}}_s$ | Non homopolar component of the output voltage vector \mathbf{V}_s . |
| v_{sj}^g | g component of the j th vector in the space vector sequence in the static gh frame. |
| \mathbf{v}_{sj} | j th vector in the switching vector sequence. |
| $\tilde{\mathbf{v}}_{sj}$ | Non homopolar component of the switching vector \mathbf{v}_{sj} . |
| $\check{\mathbf{v}}_{sj}$ | Projection onto the plane $\check{\pi}$ of the switching vector \mathbf{v}_{sj} . |
| \mathbf{V}_{sj} | Converter output voltage vector corresponding to the switching vector \mathbf{v}_{sj} . |
| v_s^k | k component of the output vector \mathbf{v}_s . |
| \tilde{v}_s^k | k component of vector $\tilde{\mathbf{v}}_s$. |

| | |
|--------------------|---|
| \check{v}_s^k | k component of vector $\check{\mathbf{v}}_s$. |
| V_s^k | k component of the output voltage vector \mathbf{V}_s , i.e., converter output voltage of phase k . |
| \bar{V}_s^k | k component of vector $\bar{\mathbf{V}}_s$. |
| \tilde{V}_s^k | k component of vector $\tilde{\mathbf{V}}_s$. |
| \check{v}_{sj}^k | k component of vector $\check{\mathbf{v}}_{sj}$. |
| \tilde{v}_{sj}^k | k component of vector $\tilde{\mathbf{v}}_{sj}$. |
| v_{sj}^k | k component of the switching vector \mathbf{v}_{sj} . |
| V_{sj}^k | k component of the vector \mathbf{V}_{sj} , i.e., output voltage of the converter corresponding to the level v_{sj}^k . |
| $V_{T_i^k}$ | Voltage across transistor T_i^k . |
| w | Angular frequency. |
| Z^k | Load impedance of phase k . |

Chapter 1

Introduction

1.1 Motivation and Objectives

The field of high-power drives has been one of the most active areas in research and development of power electronics in the last decades. Several industrial processes have increased their power-level needs, driven mainly by economy of scale (production levels and efficiency), triggering the development of new power semiconductors, converter topologies, and control methods.

In high power applications that require high voltages and high currents the maximum ratings of power semiconductors become a real handicap. The series connection of power switches is the solution for dealing with larger voltages. Nevertheless, achieving static and dynamic voltage sharing among those switches becomes a problem which led to the development of the new family of multilevel converters. On the other hand, paralleling of subsystems is the solution for dealing with larger currents. If more than three phases are used to deliver power to the load then the per-phase current rating is lower and low-current devices can be used. As a consequence multilevel multiphase converters are good candidates to be used in high-power drive applications. Nevertheless, this advantage comes at the price of a greater complexity in the inverter and in an increased control difficulty. The number of devices that must be controlled goes up from the only six switches of the two-level three-phase converters to the tens of switches of multilevel multiphase converters. At present, there are no commercial digital signal processors (DSPs) having enough appropriate built-in pulse-width modulation (PWM) units to control all those switches and a software implementation of them is very time-consuming. The field-programmable gate arrays (FPGAs) are the best candidates for the modulation implementation of multilevel multiphase converters due to their concurrent processing capability and their high number of output pins.

Two classical high frequency modulation techniques for power converters are the carrier-based sinusoidal pulse-width modulation (SPWM) and the space vector pulse-width modulation (SVPWM). The multilevel SPWM technique considers each converter leg individually; therefore, it can be applied easily to converters with any number of phases. On the contrary, the SVPWM technique treats all phases as a whole. This modulation technique is very popular in three-phase systems because it offers significant flexibility to optimize switching waveforms and is well suited for digital implementation. Three phase SVPWM algorithms have been developed by means of a graphical represen-

tation of the space vectors of the converter. A two-dimensional (2D) representation is used in three-wire converters and a three-dimensional (3D) one in four-wire converters. For systems with more than three phases, the SVPWM has not been much studied. This is mainly due to the fact that the graphical tools successfully used in three-phase systems become cumbersome in multiphase systems. Hence, only a few multiphase SVPWM algorithms have been presented in literature. Most of them have been developed for a particular number of phases or levels, and it is difficult to extend them to other cases. There are also some generic SVPWM algorithms; nevertheless, they have a high computational cost that makes them unsuitable for real-time hardware-implementation.

The main objective of this dissertation is the development of a generic SVPWM algorithm for multilevel multiphase converters with a low-computational cost suitable for hardware implementation in real time.

1.2 Review of Previous Research

1.2.1 Multilevel Technology

Even though the idea of utilizing multiple voltage levels to perform electric power conversion was patented almost thirty years ago [1, 2], the term multilevel starts with the three-level inverter introduced by Nabae et al. in 1981 [3]. Nowadays, multilevel converters are becoming increasingly popular in power applications, thanks to their ability to meet the increasing demand of power ratings and power quality associated with reduced harmonic distortion and lower electromagnetic interference [4]. The series combination of a large number of semiconductors to achieve high power ratings is well established. Choosing an arrangement where all devices are individually controlled, rather than switched together, provides multiple output voltage levels and more control opportunities. Voltage source multilevel inverter topologies [3, 5–9] synthesize a voltage waveform from several voltage levels, typically obtained from several capacitors or dc sources. The commutation of the power switches permits the addition of the voltages of these sources, which reach high voltage at the output, while the power semiconductors must withstand only reduced voltages. As the number of levels increases, the synthesized output voltage gets more steps and produces a waveform which approaches the reference more accurately. Major advantages of using multilevel inverters are [8, 9]:

- High voltage capability with voltage limited devices.
- Low harmonic distortion.
- Reduced switching losses.
- Increased efficiency.
- Good electromagnetic compatibility.

Switching State Redundancy

In multilevel converters there are switching states that produce the same output voltage. Those states are referred as redundant switching states. The switching state redun-

dancy provides some flexibility in multilevel systems and can be used to achieve certain control objectives [9] such as:

- Modulation index extension [10].
- Capacitors voltage balancing [11–15].
- Current control of dc-sources [16].
- Switches power loss balancing [16–18].
- Fault tolerant systems design [19].
- Common-mode voltage elimination [14, 20]

It is important to point out that two types of redundancy are possible [9]:

- Joint-phase redundancy.
- Per-phase redundancy.

Joint-phase redundancy is present in converters with no neutral wire, in which there are redundant switching states that provide the same line-to-line voltage. Hence, joint-phase redundancy involves changing the common-mode output voltage [11]; that is increasing or decreasing the voltage level of all phases. Per-phase redundancy refers to certain converter topologies that have redundant switching states within each phase [8, 18]; where several transistor switching combinations lead to the same line-to-ground voltage. Per-phase switching state in each topology is studied in chapter §2. The joint-phase redundancy is taken into account in the new multilevel multiphase SVPWM algorithm presented in chapter §5.

Topologies

Research in multilevel converters is focused on three main topologies [8, 9]:

- Diode-clamped converter [2, 3, 21]
- Flying capacitors converter [6, 7, 22, 23].
- Cascaded full-bridge converter [1, 7, 24, 25].

A patent search shows that the first multilevel topology introduced was the cascaded full-bridge design by Baker and Bannister in 1975 [1]. However, the multilevel technology is considered to begin around 1981 when the three-level diode clamped inverter was presented by Nabae et al. [3]. This topology clamps the voltage of the power switches by connecting them through diodes to the nodes of a series bank of capacitors that split the dc bus. Later, in 1992, the flying capacitor converter was invented by Meynard and Foch [6]. This topology directly makes use of capacitors to clamp the voltage of the power switches. More details of these topologies with a discussion of their benefits and drawbacks are given in chapter §2.

Applications

Multilevel converters have been extensively studied in a wide variety of applications such as induction machine drives [26], active rectifiers [27–29], connection of renewable energy sources to the utility grid [30–32] and static synchronous compensators [33]. Most of multilevel high-power practical applications use the diode-clamped structure. The first unified power-flow controller (UPFC) in the world [34] comprises the back-to-back connection of two identical three-level converters based on gate turn-off thyristors (GTOs), each rated at 160 MVA. It was commissioned in 1998 at the Inez Station of American Electric Power in Kentucky for voltage support and power-flow control. In [35] two back-to-back-connected diode-clamped converters perform a 10 MVA rolling-mill motor-drive application based on integrated gate-commutated thyristors (IGCTs). Unity power factor and low harmonic distortion of grid currents are major advantages of this technology compared to cycloconverters. Another example is the drive system used to transport ore from the mine to the concentrator plant in the Los Pelambres copper mine, Chile [36]. The mine is located at 1700 m and the belt conveyor system transports the material downhill. Eight 2.5 MVA motors are driven by diode-clamped converters with GTOs. The use of three-level active front ends permits regeneration of energy to the electrical network with nearly unity power factor. The system transports 5800 Tons per hour and generates a total power of 15 MW.

The cascaded full-bridge converter inverter is best suited for harmonic and reactive compensation since each bridge unit can balance its dc voltage without requiring additional isolated power sources. Alstom (formerly Alsthom) has commercialized the cascaded multilevel inverter as static compensator (STATCOM) for reactive power compensation [37]. Its first application was the ± 75 MVar STATCOM ordered by the National Grid Company as part of a 0–225 MVar relocatable static VAr compensator (SVC) for initial installation at its East Claydon 400 kV substation located to the northwest of London [38].

1.2.2 Multiphase Motor Drives

One of the first works in multiphase technology dates back to 1969, when a five-phase voltage source inverter-fed induction motor drive was proposed [39]. In the following years the multiphase motor drives have attracted a limited attention. However, the developments in three very specific application areas, namely electric ship propulsion, traction (including electric and hybrid electric vehicles) and the concept of “more-electric” aircraft, focused the attention of the research community on multiphase motor drives. In fact, the significant works published during the last years in the field rendered old survey papers [40, 41] rather obsolete. Reference [40] provides a treatment of the stator winding layouts for various phase numbers, and a discussion of space harmonics of the magnetomotive force in multiphase induction machines. Multiphase drive control schemes were reviewed in [41]. Recently, three new up-to-date surveys of the state-of-the art in this area have been published [42–44]. A survey of control schemes for asymmetrical six-phase induction motor drives and associated methods of voltage-source inverter (VSI) PWM control is given in [42]. Multiphase induction machines and drives are covered in a considerable detail in [43]. This includes basic models, control schemes in developed form, and exper-

imentally obtained illustrations of performance for various multiphase induction motor drives (asymmetrical and symmetrical six-phase, and five-phase machines). Reference [44] addresses types and advantages of multiphase machines for variable-speed applications, modeling of multiphase machines, control of variable-speed multiphase motor drives, PWM methods for multiphase VSI control, fault-tolerant operation, multiphase multimotor drive systems with a single multiphase supply, multiphase machines for electric power generation and other multiphase motor drive solutions.

Characteristics

Most of the variable-speed electric drives use three-phase machines because they have become a standard. However, increasing numbers of induction motors are not connected directly to three-phase supplies. Instead, they are fed by a power electronic converter which is connected to a three-phase supply. The output stage of the converter and the stator winding of the motor must have the same number of phases, but provided this simple requirement is met, any number of phases may be used. Major advantages of using a multiphase machine instead of a standard three-phase one are [43, 44]:

- Higher efficiency because the stator excitation produces a field with a lower space-harmonic content.
- Improved reliability and greater fault tolerance. If one phase of a multiphase-phase machine becomes open circuited, it will still self-start and will run with less derating.
- Higher torque density and reduced torque pulsations, because they are less susceptible to time-harmonic components in the excitation waveform.
- Lower per phase power handling requirements, which reduces the required rating of power electronic switches for a given motor output power.
- Improved noise characteristics as a consequence of the previous items.

Since control of an ac machine, regardless of the number of phases, requires only two currents, additional degrees of freedom that exist in multiphase machines can be used to enhance the torque production by stator harmonic current injection [45, 46] or to improve the fault tolerance. The machine can still continue to operate in post-fault conditions with the rotating magneto-motive force, provided that an appropriate post-fault current control strategy is developed [47–50]. An alternative use of these additional degrees of freedom can be found in multi-motor drive systems with single inverter supply. Stator windings of the machines are connected in series, using phase transposition, so torque producing currents of one machine appear as non-torque producing currents for all the other machines. Vector control techniques enable completely decoupled and independent control of the machines, although they are connected in series and a single VSI is used as the supply [51, 52].

Applications

Variable-speed multiphase drives are of industrial interest in applications where their advantages outweigh the lack of the off-the-shelf availability of both machines and power electronic converters. Multiphase machines are currently being used where both the machine and its control electronics are designed as a system, rather than as individual components. They have been found to be ideally suited, for example for the direct drives in marine applications, where their fault tolerance, high efficiency, low acoustic noise and the ability to distribute the drive electronics are seen as particularly advantageous [53–55]. Drive systems for safety or critical applications, such as more-electric aircraft, normally use the available additional degrees of freedom for achieving fault-tolerant operation of the drive [56–59]. Such drives utilize a modular design of both the machine and the supply system; therefore, they use individual full-bridge (single-phase) inverters. Other multiphase applications include electric vehicles propulsion [60–63] and locomotive traction [64, 65].

1.2.3 Multilevel Multiphase Voltage Source Inverters

Most of existing considerations of multiphase motor drives are related to two-level VSI supply. Multilevel inverters for multiphase variable speed drives are rare. Topologies considered in literature are typically either cascaded full-bridge or diode clamped inverters [54, 64, 66–70]. An interesting technique to realize a multilevel supply for open-end stator winding machines consists of supplying both ends of the winding with two VSIs. Both inverters can be of the same or different number of levels, which can be two or more. Such an approach has been considered for a six-phase machine in [71] where four two-level three-phase VSIs, configured into two six-phase VSIs, are connected at each side of the stator winding.

Multilevel inverters for multiphase variable speed drives appear to be a natural solution for high-power motor drives, such as those aimed at electric ship propulsion [54, 66, 67] or locomotive traction [64]. Some preliminary considerations related to multilevel inverter applications for five-phase induction motor drives are available in [66]. Multilevel VSIs are also considered in [54] as one of the options for the supply of a 36.5 MW electric ship propulsion drive. Some theoretical considerations are given in conjunction with four- and five-level nine-phase VSIs as potential solutions, while more detailed simulation study has been reported for the three-level nine-phase VSI [54]. Theoretical and simulation considerations, related to multilevel operation of three five-phase full-bridge inverters aimed at supplying the 15-phase induction motor for ship propulsion [67], show that the increase in the number of levels can considerably improve the current and torque ripples, when compared to the simplest two-level mode of operation. An attempt to develop a multilevel multiphase induction motor drive for a locomotive application is described in [64]. A nine-phase induction machine is used, configured into three three-phase windings with isolated neutral points. These are supplied from three-phase three-level inverters. Since the neutrals are isolated, three-level inverter control is an extension of what is used in three-phase drives. The same applies to the considerations related to the six-phase induction motor drive in [68], where two five-level three-phase inverters are used.

1.2.4 Multilevel Multiphase PWM

Multilevel PWM control of VSI has been focused on three-phase systems because, nowadays, multiphase induction motor drives are invariably supplied from two-level inverters. Modulation methods used in multilevel inverters can be classified according to switching frequency, as fundamental switching frequency methods, and high switching frequency PWM methods [8, 72].

Fundamental switching frequency methods generally perform one or two commutations of the power semiconductors during one cycle of the output voltages generating a staircase waveform. Hence, the switching losses are kept low but the output voltages have a high low-order harmonic content. These unwanted low-order harmonics lead to the flow of stator harmonic currents in motor drive applications. Representative techniques of this family are the multilevel selective harmonic elimination [73, 74] and the space vector control [75, 76]. In the former technique, the switching instants are calculated to eliminate the most significant low-order harmonics whereas the high-frequency harmonics must be removed by using additional filtering circuits. The objective in the latter technique is to deliver to the load a voltage vector that minimizes the space error or distance to the reference vector. This method is simple and attractive for high number of levels, nevertheless, when the number of levels decreases, the error increases and the current ripple becomes higher.

If a high switching frequency can be accommodated then the low-order voltage harmonics can be reduced at expense of higher switching losses. Methods that work with high switching frequencies have many commutations for the power semiconductors in one period of the fundamental output voltage. Two very popular methods in industrial applications are the SPWM [77–79] and the SVPWM [72, 80, 81].

Carrier-Based Sinusoidal PWM

SPWM generates the trigger signal for the power switches by comparing the voltage reference (typically a sinusoidal wave) with a triangular carrier signal. Based on the classical SPWM with triangular carriers, several multicarrier techniques have been developed to reduce the harmonic distortion in multilevel inverters. Some methods use carrier disposition and others use phase shifting of multiple carrier signals [78, 79].

A very common practice when SPWM is used in three-phase industrial applications for the multilevel inverter is the injection of a third harmonic to increase the output voltage and improve the dc bus utilization [24, 77, 82]. Likewise, in multiphase VSIs it is also possible to improve the dc bus utilization by injecting the appropriate zero-sequence harmonic (or adding an offset) into leg voltage references [83, 84]. Nevertheless, this improvement reduces as the number of phases increases [85]. The gain in maximum fundamental in the linear modulation region is only 5.15% for a five-phase VSI and 2.57% for a seven-phase VSI, while it is 15.47% in the three-phase VSI [43].

When compared with other techniques, the main advantage of the SPWM technique is the easiness of implementation. This advantage becomes more and more pronounced as the phase number increases because this technique controls each phase separately. Hence its implementation in multiphase converters is rather straightforward [86]. Another advantageous feature of multilevel SPWM is that the switching frequency of each transistor

can be lower than the effective converter output switching frequency, thus reducing the switching losses and filter requirements.

The SPWM has been used with multiphase series-connected induction motor drives. In this case, the multiphase VSI output voltage is required to contain several voltage components of different phase sequence and in general different magnitudes and frequencies. If SPWM is used, the modulation scheme is straightforward and very simple to implement because the inverter leg voltage references are simply formed by summing the appropriate sinusoidal signals. It is still possible to utilize zero-sequence addition to improve the dc bus utilization. This implementation has been discussed in [87] for a five-phase two-motor drive.

Space Vector PWM

SVPWM arises from the vectorial description of the switching states of power converters. Since those switching states constitute a discrete set of space vectors, the continuous reference vector must be approximated by the time averaging of a space vector sequence. The SVPWM technique is used to determine this sequence and to calculate the time interval corresponding to each vector. The way that the space vectors are selected and ordered in the sequence has a great effect on the output voltage harmonics and on the converter switching losses. Usually, the vector sequence is made with the space vectors nearest to the reference vector to reduce the output current ripple.

This modulation technique considers all phases of the converter as a whole, this being the main difference with carrier based methods. The number of available switching states in a P -phase converter changes according to the law N^P where N is the number of levels. This means that, as the number of phases and levels increases, the problem of devising an adequate SVPWM scheme becomes more and more involved.

Multilevel Three-Phase Converters The SVPWM technique offers significant performance benefits and has proved to be very popular in three-phase systems [88]. The modulation algorithms can be classified as algorithms that cannot handle the joint-phase switching state redundancy and algorithms that can handle it. The former algorithms are valid for converters with and without neutral wire. Nevertheless, in the case of converters without neutral wire the latter algorithms are more suited because they allow choosing the best switching states among the redundant ones to carry out a certain switching strategy. As the number of levels or phases increases, the amount of redundant switching states increases dramatically; and consequently, choosing the switching state sequence becomes more involved. Hence, in multilevel multiphase systems it is necessary to devise a technique that makes easy the selection of the switching states. In [72], a very simple SVPWM algorithm for multilevel three-phase topologies with state redundancy was presented. However, it does not address the redundant switching states selection. The method introduced in [81] for three-phase inverters without state redundancy was later extended to four-wire topologies with state redundancy in [89]. Recently, in [90] a new SVPWM method with state redundancy for single-phase converters has been presented.

For three phase inverters, some researchers [91–95] have proposed to perform the multilevel modulation by means of a two-level modulator. A new method for the switching time calculation, where the three-level space vector diagram is divided into six two-level

space vector diagrams, is introduced in [91]. However, this paper does not include the extension of the method for a number of levels higher than three. In [92] a similar scheme is also presented for a three-level inverter. This scheme cannot be directly applied to a multilevel inverter, nevertheless the principle explained in the paper permits to make an N -level SVPWM from six $(N - 1)$ -level SVPWM. Therefore, that implies that as the number of levels increases, both complexity and computation cost increase exponentially. In [93], the multilevel space vector diagram is divided into all possible two-level space vector hexagons. After that, a linear transformation is used to find out the center of one of those hexagons to calculate the switching times. In [94], a general solution to adapt an existing two-level modulator to a multilevel inverter is presented. This technique requires the storage of switching states with a memory requirement that grows exponentially with the number of inverter levels. A new method for the duty cycle calculation in two-level inverters is proposed in [95] that can be extended to multilevel inverters by adding an offset in the duty-cycle expression.

Two-Level Multiphase Converters Existing multiphase SVPWM algorithms are two-level algorithms that use the decomposition of switching states in multiple 2D planes [96–100]. The decomposition of every switching state in all those planes provides the space vectors, which therefore map simultaneously into all planes. Those planes are orthogonal sub-spaces of a multidimensional space so there is no coupling among them. This constitutes the basis of the switching states decomposition approach discussed in [101]. Early implementations of SVPWM for five-phase ac motor drives [102, 103] attempt to make the control by considering only one of those planes (by analogy with three-phase converters). Nevertheless, such an approach excites the other planes leading to the flow of large low-order stator current harmonics. To avoid the existence of unwanted low-order harmonics in the output voltage it is necessary to consider all orthogonal 2D sub-spaces. As a consequence, at least a number of space vectors equal to the number of phases must be applied during every switching period [83, 104]. It then becomes possible to set the average applied space vector in all planes. Once when the space vectors are selected, it is necessary to calculate their dwell times.

If requirement is just to generate only sinusoidal output voltages, the reference voltage space vector is nonzero only in one plane and the SVPWM scheme has to ensure that zero average voltage space vector is applied in all the other planes, so that undesirable low order harmonics do not appear in the output. This is most easily done utilizing the analytical expressions for computation of dwell times of the space vectors, such as those developed in [96, 97, 105] for a five-phase VSI, in [100] for a seven-phase VSI, and in [106] for a nine-phase VSI.

In some applications it is desirable to utilize higher stator current harmonic injection for the purpose of the torque enhancement. Thus, the reference voltage space vectors now have nonzero values in all the planes. If the space vectors are selected on the basis of the reference in only one plane, then the achievable voltages in all the other planes are automatically limited [98, 107]. This is, however, not the problem in this case since voltage references in all the other planes are considerably smaller than the reference in the first plane. Dwell times calculation of the active vectors is typically done by solving online a system of equations that relates the axis components of the selected VSI space vectors with the corresponding components of the references [98, 107]. This imposes a significant

computational burden and makes this approach rather complicated for implementation purposes. Some improvements with regard to the implementation complexity have been reported in [108]. A different method, which relies on summation of device turn-on times for creating the desired references in the two planes, but still considers in the initial vector selection only neighboring vectors in one plane, has been developed in [109]. A general approach to find the dwell times, once when the space vectors have been selected, has been developed in [110] and further successfully applied in [111] to control a five-phase machine with third harmonic injection.

The most demanding situation arises where there is one independent reference voltage space vector in each plane, such as in multimotor drive system with a single VSI supply. The first attempt to develop an SVPWM method to cope with such situation was based on using the concept of multiple 2D spaces [112] for a five-phase two-motor structure. It uses two planes, and completely independently selects in each one a set of space vectors neighboring the corresponding reference. Then it becomes possible to create two voltage space vector references independently, using the same approach and the same analytical expressions as for the case of purely sinusoidal output voltage generation. However, the application of the selected vectors is done in such a way that the reference voltage in one plane is applied in one switching period, while the voltage reference in the other plane is applied in the next switching period. Such an approach to realization of the two space vector voltage references has automatically restricted the available fundamental for each of the two machines to only 50% of its maximum value for the given dc-link voltage (since only every second period is used for application of the voltage to any of the two machines). A similar approach is used in [113] to generate the output voltage with two independent space vector components for a five-phase VSI. It selects in each plane a set of space vectors neighboring the corresponding reference and creates two voltage space vector sequence independently, using the same approach as in the previous case. Both sequences are manipulated to obtain the final sequence that is applied within a single switching period, thus avoiding the need for doubling the dc bus.

Multidimensional Approach The SVPWM for multiphase inverters is in essence a multidimensional problem and space vector decomposition is a relatively simple way of dealing with it in 2D subspaces. The attempts to formulate the SVPWM directly by using a multidimensional space are rare. Such an approach was investigated for a five-phase VSI supplying two series-connected five-phase induction machines in [114] and was deemed to be of little practical value because of the huge complexity involved in the selection of the space vectors. Some improvement with regard to the dwell time calculation has been reported in [110]; however, the problem of the space vector selection is the main obstacle in the application of the multidimensional SVPWM. A new method for a P -phase inverter that can be used to aid the selection of the most appropriate space vectors in the multidimensional space is presented in [115]. Nevertheless, even for a five-phase drive, online implementation of this method has high memory requirements and it is very time consuming [116].

Multilevel Multiphase Converters Multilevel multiphase SVPWM has not been studied much. In [68] two five-level three-phase inverters are used to feed a six-phase machine. Since the neutrals are isolated, the three-level inverter control is an extension of

what is used in three-phase drives. Some preliminary considerations related to multilevel SVPWM for five-phase induction motor drives are given in [66]. The SVPWM algorithm proposed in [69] only takes into account one of the 2D planes in switching states decomposition. Therefore, it is not suitable for higher stator current harmonic injection or multimotor drive control.

Problem of SVPWM for multilevel multiphase VSI has not been solved. An algorithm valid for any number of levels and phases suitable for online implementation and that can effectively synthesize a generic reference vector voltage is required.

1.3 Major Results

The work in this dissertation is divided into the following major parts:

Chapter 2: “Per-Phase Redundancy in Multilevel Converters”

This chapter extends the introduction with the description and a discussion of main multilevel converter topologies: diode-clamped converter, flying capacitor converter and cascaded full-bridge converter. Per-phase redundancy is studied and the switching laws for each topology are derived. In addition, analytical expressions to calculate the number of redundant states for each output level are provided. Finally, the similarities in those switching laws are studied and the principles to adapt the trigger signals among the multilevel topologies are discussed.

Chapter 3: “Implementation of Multilevel SVPWM Algorithms”

This chapter studies the hardware implementation of two well-known multilevel three-phase SVPWM techniques. The key issues in hardware implementation of SVPWM algorithms are identified and discussed through the chapter. Because of the high number of power switches that must be controlled in parallel in multilevel topologies the utilization of FPGA is discussed. The first algorithm is a 3D SVPWM algorithm that does not take into account the switching state redundancy present in converters without neutral wire. The second one is a 2D SVPWM algorithm with the capability to handle joint-phase switching state redundancy present in such converters. The extra implementation complexity of algorithms with state redundancy is identified and studied. Finally, test results with a neutral-point clamped (NPC) VSI are presented.

Chapter 4: “Multilevel Multiphase SVPWM Algorithm”

This chapter introduces a novel SVPWM algorithm for multilevel multiphase voltage source converters. This algorithm is computationally extremely efficient and thus suitable for online implementations, it is almost independent of the number of levels and phases of the converter and it can be applied to most multilevel topologies. The chapter starts with the presentation of a novel multidimensional formulation of the modulation problem. In this formulation the state redundancy is not considered. Next, the problem is solved and it is demonstrated that a multilevel multiphase modulator can be realized from a two-level multiphase modulator. Then a new two-level multiphase SVPWM algorithm is developed.

The algorithm was implemented in a low-cost FPGA and it was tested in laboratory with a real prototype using a five-level five-phase VSI. This chapter also covers the application of this algorithm to three-phase converters with and without neutral wire. The resulting algorithms are compared with other existing three-phase SVPWM techniques.

Chapter 5: “Multilevel Multiphase SVPWM Algorithm With Switching State Redundancy”

This chapter includes the joint-phase redundancy in the problem formulation presented in the previous chapter. The result is a new SVPWM algorithm for multilevel multiphase voltage source converters without neutral wire. This algorithm also makes use of the previously developed two-level multiphase algorithm. As the number of levels increases the number of redundant states increases dramatically; hence, the algorithm includes a tool that simplifies the redundant states selection. Even though management of redundancy adds some complexity to the algorithm, it is still suitable for online implementation, it is almost independent of the number of levels and phases of the converter and it can be also applied to the main multilevel topologies. As in previous chapter, the new algorithm was implemented in a low-cost FPGA and it was tested in laboratory by using the same converter. This chapter also covers the application of this algorithm to multilevel three-phase converters with and without neutral wire. The resulting algorithms are compared with other existing SVPWM techniques.

Chapter 6: “Conclusion and Future Research”

The main conclusions of this dissertation are summarized in this chapter and some recommendations for future research topics are provided.

Appendix A: “Further Information About Testing Facilities”

Original research results, described in conjunction with chapters 2-5, have been verified using various laboratory prototypes. Each of the relevant chapters contains brief description of the corresponding experimental facility used in the algorithm verification. Further information on laboratory prototypes is provided in this Appendix.

Chapter 2

Per-Phase Redundancy in Multilevel Converters

***Abstract** — Switching states redundancy provides flexibility in multilevel systems and may be used to achieve some control objectives such as capacitors voltage balancing, current control of dc-sources, switch power loss balancing and fault tolerance. This chapter studies the per-phase redundancy in three basic multilevel topologies: diode-clamped converter, flying capacitor converter and cascaded bull-bridge converters. The number of per-phase redundant switching states, the rules for triggering each topology and the relationships among them are studied through the chapter.*

2.1 Introduction

Redundancy is referred to as the capability of power converters to produce the same output voltages with different switching states. Two types of redundancy are possible: per-phase redundancy and joint-phase redundancy [9].

The joint-phase redundancy is present in converters with no neutral wire and does not depend on the converter topology. It is based on the fact that several different line-to-ground voltage combinations provide the same line-to-line voltage [11]. Joint-phase redundancy has been studied in depth and it is well known in three-phase converters. In fact, space vector pulse-width modulation (SVPWM) techniques [118] have become very popular in three-phase applications because their aptitude to manage it. Classical carrier-based sinusoidal pulse-width modulation (SPWM) techniques deal with joint-phase redundancy by means of a zero sequence offset added to the reference waveform of every phase [16, 119, 120]. Different zero sequences lead to different modulation strategies [10, 120] with different performance levels. The equivalence between this zero sequence in SPWM and the SVPWM is studied in [118].

Per-phase redundancy refers to certain power converter topologies that have redundant switching states within each phase [8, 18] so that several transistor switching combinations lead to the same line-to-ground voltage. Per-phase redundancy can be used or combined with joint-phase redundancy in multilevel converters to achieve certain control objectives

Research work included in this chapter has been published in the journal *International Review of Electrical Engineering* [117].

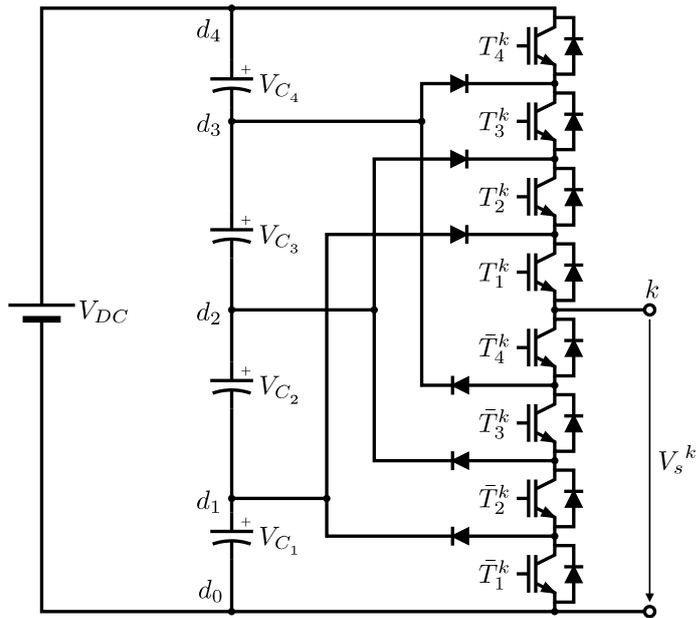


Figure 2.1: Five-level diode-clamped converter.

such as capacitors voltage balancing [11–15], current control of dc-sources [16], switch power loss balancing [16–18] and increase in fault tolerance [19].

This chapter focuses on per-phase redundancy and redundant switching state selection in the main multilevel topologies. Sections §2.2, §2.3 and §2.4 review the triggering constraints in the diode-clamped converter, the flying-capacitor converter and the cascaded full-bridge converter respectively. Switching laws for triggering each topology are presented. Analytical expressions to calculate the number of redundant states for each level and the minimum number of switchings between two consecutive states are also provided. The similarities among those switching laws are analyzed in section §2.5 and the basics to adapt the trigger signals among the three topologies are provided. The conclusions of the chapter are given in section §2.6.

2.2 Diode-Clamped Converter

The diode-clamped converter [2, 3, 5] provides multiple voltage levels through the connection of the phases to a series bank of capacitors. According to the original invention [2], the concept can be extended to any number of levels by increasing the number of capacitors. Early descriptions of this topology were limited to three-levels [3] where two capacitors are connected across the dc bus resulting in one additional level. The additional level was the neutral point of the dc bus, so the terminology neutral-point clamped (NPC) inverter was introduced [3]. However, with more than three levels the term diode-clamped topology is applied [8, 33, 121, 122]. Due to capacitor voltage balancing issues, the diode-clamped inverter implementation has been mostly limited to three-levels, and because of industrial developments over the past several years, it is now used extensively in industry applications [26].

Figure 2.1 shows the topology of one phase of a five-level diode-clamped converter. In

summary, the dc-bus is split by means of series connected capacitors, and the phase node k can be connected to any node in the capacitor bank (d_0, d_1, d_2, d_3 or d_4). In this representation, the labels T_i^k are used to identify the transistors as well as the semiconductor logic (1 = on and 0 = off). Since the transistors are always switched in pairs, to avoid short-circuiting of dc-link capacitors, the complementary pairs are labeled as T_i^k and \bar{T}_i^k accordingly. In practical implementations, some dead time is inserted between the transistor signals and their complements meaning that both transistors in a complementary pair may be switched off for a small amount of time during a transition [123]. However, for the discussion herein, the dead time will be ignored and it will be considered that the complementary trigger signals always sum to one:

$$T_i^k + \bar{T}_i^k = 1. \quad (2.1)$$

Hence, \bar{T}_i^k is the inverse of T_i^k and only one of the transistors in the pair is independent. The same consideration applies to the flying capacitor and the cascaded full-bridge topologies.

If the power losses are neglected then the transistor blocking voltage $V_{T_i^k}$ in a multilevel topology is given by

$$V_{T_i^k} = (1 - T_i^k)V_{C_i} \quad (2.2)$$

where V_{C_i} is the voltage of the capacitors in the bank. For every converter level, it must be taken into account that if $T_i^k = 0$ then $T_{i+1}^k = 0$. The output voltage of one phase V_s^k can be calculated from transistor voltages by means of

$$V_s^k = V_{DC} - \sum_{i=1}^{N^k-1} V_{T_i^k} \quad (2.3)$$

where N^k is the number of levels of the phase k of the converter, and V_{DC} is the full dc link voltage. If (2.2) is substituted in the above expression then the output voltage can be obtained from the trigger signals as

$$V_s^k = \sum_{i=1}^{N^k-1} T_i^k V_{C_i}, \quad \text{where if } T_i^k = 0 \text{ then } T_{i+1}^k = 0. \quad (2.4)$$

Usually, all capacitors in the bank are equally charged

$$V_{C_i} = \frac{V_{DC}}{N^k - 1} = V_{dc}, \quad \text{for all } i. \quad (2.5)$$

In this case (2.2) and (2.4) can be rewritten as

$$V_{T_i^k} = (1 - T_i^k)V_{dc} \quad (2.6)$$

$$V_s^k = V_{dc} \sum_{i=1}^{N^k-1} T_i^k. \quad (2.7)$$

Since the result of the summation term is always an integer number, the output voltage is an integer multiple of fixed voltage V_{dc} :

$$V_s^k = v_s^k V_{dc}, \quad \text{where } v_s^k \in \mathbb{N}. \quad (2.8)$$

TABLE 2.1
FIVE-LEVEL DIODE-CLAMPED CONVERTER RELATIONSHIPS.

| T_1^k | T_2^k | T_3^k | T_4^k | V_s^k | V_s^k (if $V_{C_i} = V_{dc}$) | v_s^k |
|---------|---------|---------|---------|---|----------------------------------|---------|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | V_{C_1} | V_{dc} | 1 |
| 1 | 1 | 0 | 0 | $V_{C_1} + V_{C_2}$ | $2V_{dc}$ | 2 |
| 1 | 1 | 1 | 0 | $V_{C_1} + V_{C_2} + V_{C_3}$ | $3V_{dc}$ | 3 |
| 1 | 1 | 1 | 1 | $V_{C_1} + V_{C_2} + V_{C_3} + V_{C_4}$ | $4V_{dc}$ | 4 |

Therefore, V_{dc} and v_s^k are the voltage step and the output level of the converter, respectively. The output level can be calculated from the switching state by means of

$$v_s^k = \sum_{i=1}^{N^k-1} T_i^k, \quad \text{where if } T_i^k = 0 \text{ then } T_{i+1}^k = 0. \quad (2.9)$$

Table 2.1 shows the trigger signals, output voltages and output levels corresponding to the five-level converter in Figure 2.1.

2.2.1 Switching State Redundancy

Since the multilevel modulator provides the converter output level v_s^k , the inverse relationship of (2.9) is required to determine the switching state from v_s^k . The solution of (2.9) is unique and can be written as

$$T_i^k = \begin{cases} 1 & \text{if } i \leq v_s^k \\ 0 & \text{if } i > v_s^k. \end{cases} \quad (2.10)$$

Thus, there is only one switching state for each output level. As a consequence the number of redundant states $R_d(v_s^k)$ corresponding to each level v_s^k is always equal to one:

$$R_d(v_s^k) = 1, \quad \text{for all } v_s^k. \quad (2.11)$$

Because of this zero per-phase redundancy, the joint-redundancy must be used to implement capacitor balancing techniques [11, 124, 125]. This lack of flexibility becomes critical in converters with a high number of levels and it is the main drawback in converters with more than three levels [126].

In this topology, only one switch must be turned-on and its complementary one turned-off to change between two adjacent levels. Consequently, to change the switching state from a level $v_{s_i}^k$ to another $v_{s_j}^k$, the number of devices $S_{w_{i \rightarrow j}}^k$ that must be switched is given by

$$S_{w_{i \rightarrow j}}^k = 2|v_{s_i}^k - v_{s_j}^k|. \quad (2.12)$$

2.2.2 Characteristics

Advantages of the diode-clamped topology compared with its multilevel counterparts are the following [125]:

- It uses a lower number of capacitors. Although this topology requires some additional clamping diodes, a low number of reactive components is usually preferred from the standpoint of cost.
- It can be connected to a single dc-link voltage. The floating-capacitor topology also shares this advantage, but the cascade converter does not, since this converter requires multiple isolated dc power supplies.

Nevertheless, some practical experience with this topology reveals technical difficulties that complicate its application, as follows [125]:

- Clamping diodes are subjected to a high voltage stress in converters with more than three levels. As a result, series connection of the diodes is required. If the inverter runs under pulse-width modulation (PWM), the diode reverse recovery of clamping diodes becomes the major design challenge in high-voltage high-power applications. This issue complicates the design and raises reliability and cost concerns.
- Maintaining the charge balance of the capacitors in topologies with more than three levels has been demonstrated to be impossible for some operating conditions [126]. Such balancing problems appear when dealing with deep modulation indices and active currents. Therefore, high ac output voltages cannot be achieved, which inhibits the most important attribute of multilevel converters. Significant balancing improvements are obtained when two or more converters are connected to the same dc link [124]. These converters are also used for static reactive compensation circuits in which no active power is transmitted and the balancing problem can be handled.
- Although proper control of the three-level topology overcomes the voltage balance concern, a low-frequency voltage ripple appears in the neutral point when dealing with large modulation index and low power factor. The maximum voltage applied to the devices is higher due to this oscillation, and additionally, it produces low-frequency distortion in the ac output voltages. Some solutions for this problem have been proposed in [125].

2.3 Flying-Capacitor Converter

Another fundamental multilevel topology is the flying capacitor converter, which involves series connection of capacitor clamped switching cells [6]. In this topology the voltage clamping is achieved by means of capacitors that float with respect to the ground. Figure 2.2 shows the structure for one phase of a five-level flying capacitor converter.

Neglecting the device voltage drop as well as conduction losses, the blocking voltage of each switch $V_{T_i^k}$ depends on the voltage of the floating capacitor $V_{C_i^k}$ as

$$V_{T_i^k} = (1 - T_i^k)(V_{C_i^k} - V_{C_{i-1}^k}) \quad (2.13)$$

where $V_{C_0^k} = 0$ and $V_{C_{N^k-1}^k} = V_{DC}$ is the voltage of the full dc link. The phase output voltage V_s^k can be calculated from the transistor voltages by means of

$$V_s^k = V_{C_{N^k-1}^k} - \sum_{i=1}^{N^k-1} V_{T_i^k}. \quad (2.14)$$

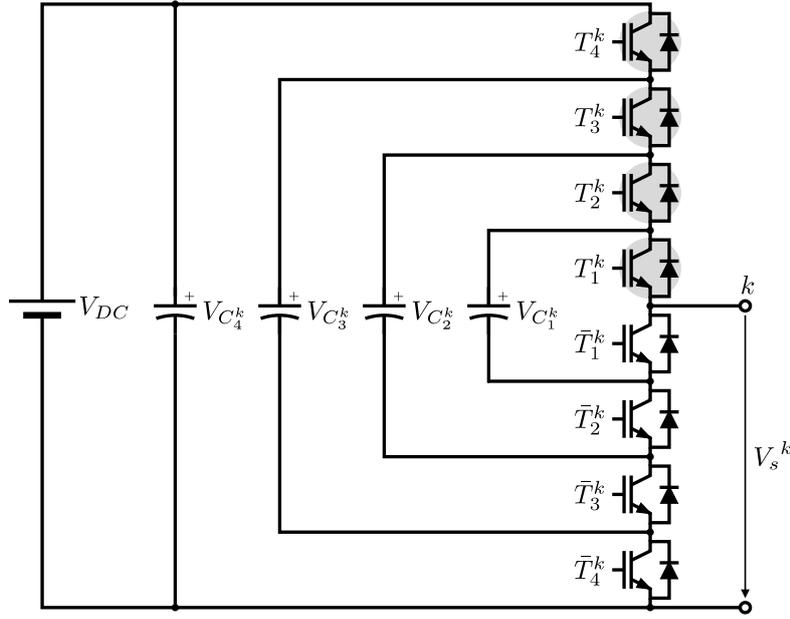


Figure 2.2: Five-level flying capacitor converter.

If (2.13) is substituted in the above expression then the output voltage can be obtained from the trigger signals as

$$V_s^k = \sum_{i=1}^{N^k-1} T_i^k (V_{C_i^k} - V_{C_{i-1}^k}), \quad \text{where } V_{C_0^k} = 0 \text{ and } V_{C_{N^k-1}^k} = V_{DC}. \quad (2.15)$$

Usually, the flying capacitors voltage is an integer multiple of a fixed voltage V_{dc} :

$$V_{C_i^k} = iV_{dc}, \quad i \in \mathbb{N}. \quad (2.16)$$

In this case (2.13) and (2.15) can be simplified as

$$V_{T_i^k} = (1 - T_i^k)V_{dc} \quad (2.17)$$

$$V_s^k = V_{dc} \sum_{i=1}^{N^k-1} T_i^k. \quad (2.18)$$

Since the result of summation term is always an integer number, the output voltage is an integer multiple of the fixed voltage V_{dc} :

$$V_s^k = v_s^k V_{dc}, \quad \text{where } v_s^k \in \mathbb{N}. \quad (2.19)$$

Therefore, V_{dc} and v_s^k are the voltage step and the output level of the converter, respectively. Hence, the relationship between the output level v_s^k and the switching state is

$$v_s^k = \sum_{i=1}^{N^k-1} T_i^k. \quad (2.20)$$

Therefore, there are different combinations of T_i^k that provide the same output level as it is shown in Table 2.2. This redundancy gives great flexibility in the selection of the transistors that must be switched to obtain a particular output level.

TABLE 2.2
FIVE-LEVEL FLYING CAPACITOR CONVERTER RELATIONSHIPS.

| T_1^k | T_2^k | T_3^k | T_4^k | V_s^k | V_s^k (if $V_{C_i^k} = iV_{dc}$) | v_s^k |
|---------|---------|---------|---------|---|-------------------------------------|---------|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | $V_{C_1^k}$ | V_{dc} | 1 |
| 0 | 1 | 0 | 0 | $V_{C_2^k} - V_{C_1^k}$ | V_{dc} | 1 |
| 0 | 0 | 1 | 0 | $V_{C_3^k} - V_{C_2^k}$ | V_{dc} | 1 |
| 0 | 0 | 0 | 1 | $V_{C_4^k} - V_{C_3^k}$ | V_{dc} | 1 |
| 1 | 1 | 0 | 0 | $V_{C_2^k}$ | $2V_{dc}$ | 2 |
| 1 | 0 | 1 | 0 | $V_{C_3^k} - V_{C_2^k} + V_{C_1^k}$ | $2V_{dc}$ | 2 |
| 1 | 0 | 0 | 1 | $V_{C_4^k} - V_{C_3^k} + V_{C_1^k}$ | $2V_{dc}$ | 2 |
| 0 | 1 | 1 | 0 | $V_{C_3^k} - V_{C_1^k}$ | $2V_{dc}$ | 2 |
| 0 | 1 | 0 | 1 | $V_{C_4^k} - V_{C_3^k} + V_{C_2^k} - V_{C_1^k}$ | $2V_{dc}$ | 2 |
| 0 | 0 | 1 | 1 | $V_{C_4^k} - V_{C_2^k}$ | $2V_{dc}$ | 2 |
| 1 | 1 | 1 | 0 | $V_{C_3^k}$ | $3V_{dc}$ | 3 |
| 1 | 1 | 0 | 1 | $V_{C_4^k} - V_{C_3^k} + V_{C_2^k}$ | $3V_{dc}$ | 3 |
| 1 | 0 | 1 | 1 | $V_{C_4^k} - V_{C_2^k} + V_{C_1^k}$ | $3V_{dc}$ | 3 |
| 0 | 1 | 1 | 1 | $V_{C_4^k} - V_{C_1^k}$ | $3V_{dc}$ | 3 |
| 1 | 1 | 1 | 1 | $V_{C_4^k}$ | $4V_{dc}$ | 4 |

2.3.1 Switching State Redundancy

The trigger signals T_i^k required to obtain at the output level v_s^k can be obtained by solving (2.20). This expression does not have a unique solution unlike in the diode clamped topology. It only establishes the following law for switching the semiconductors:

“The number of independent switches that must be turned-on for achieving a particular level is equal to the numeric value of the output level.”

There is no preference in the selection of the switches because the expression in (2.20) does not establish priorities among them. For example, to get the level number two ($v_s^k = 2$) in the converter in Figure 2.2 two switches must be chosen among the highlighted ones. The selected switches must be turned-on and the remaining highlighted switches must be turned-off. The state of the non highlighted switches is determined by the state of their complementary devices. This can be verified by observing Table 2.2.

In an N^k -level converter, there are $N^k - 1$ pairs of active switches. As a consequence, the number of redundant states $R_d(v_s^k)$ corresponding to the level v_s^k can be calculated as the number of combinations without repetition of v_s^k elements chosen in a set of $N^k - 1$ elements:

$$R_d(v_s^k) = \binom{N^k - 1}{v_s^k} = \frac{(N^k - 1)!}{(N^k - v_s^k - 1)! v_s^k!}. \quad (2.21)$$

Table 2.3 shows the number of redundant states for each output level in multilevel flying capacitor converters.

TABLE 2.3

NUMBER OF PER-PHASE REDUNDANT SWITCHING STATES IN MULTILEVEL FLYING CAPACITOR CONVERTERS.

| | $R_d(0)$ | $R_d(1)$ | $R_d(2)$ | $R_d(3)$ | $R_d(4)$ | $R_d(5)$ | $R_d(6)$ | $R_d(7)$ | $R_d(8)$ |
|-----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| $N^k = 3$ | 1 | 2 | 1 | – | – | – | – | – | – |
| $N^k = 4$ | 1 | 3 | 3 | 1 | – | – | – | – | – |
| $N^k = 5$ | 1 | 4 | 6 | 4 | 1 | – | – | – | – |
| $N^k = 6$ | 1 | 5 | 10 | 10 | 5 | 1 | – | – | – |
| $N^k = 7$ | 1 | 6 | 15 | 20 | 15 | 6 | 1 | – | – |
| $N^k = 8$ | 1 | 7 | 21 | 35 | 35 | 21 | 7 | 1 | – |
| $N^k = 9$ | 1 | 8 | 28 | 56 | 70 | 56 | 28 | 8 | 1 |

Finally, the number of switchings $S_{w_i \rightarrow j}^k$ needed to change the inverter output from the level $v_{s_i}^k$ to the level $v_{s_j}^k$ is given by

$$S_{w_i \rightarrow j}^k \geq 2|v_{s_i}^k - v_{s_j}^k|. \quad (2.22)$$

2.3.2 Characteristics

This topology has several unique and attractive features when compared to the diode-clamped inverter:

- Additional clamping diodes are not needed.
- It has per-phase redundancy, which can be used to balance the flying capacitors so that only one dc source is needed. This is an important difference compared to the diode-clamped converter, in which only the joint-phase redundancy can be considered for the balancing issue.

Nevertheless, this topology has the following practical drawbacks:

- It requires a large number of bulk capacitors. Since the current level through all the floating capacitors is the same, they should have the same capacitance to produce similar amplitudes of their voltage ripple. Provided that the voltage rating of each capacitor used is the same as that of the main power switch, an N -level converter will require $(N - 1)(N - 2)/2$ clamping capacitors per phase in addition to $N - 1$ main dc-bus capacitors [8].
- In converters with more than three levels, some transitions between two consecutive voltage levels produce high switching frequencies. If these transitions are avoided, the amplitude of the voltage ripple in the capacitors will increase, and it might not be controllable [125].

2.4 Cascaded Full-Bridge Converter

The cascaded full-bridge converter was the first invented multilevel topology [1]. Early uses of the topology were devoted to single-phase applications [127]. Later, this approach

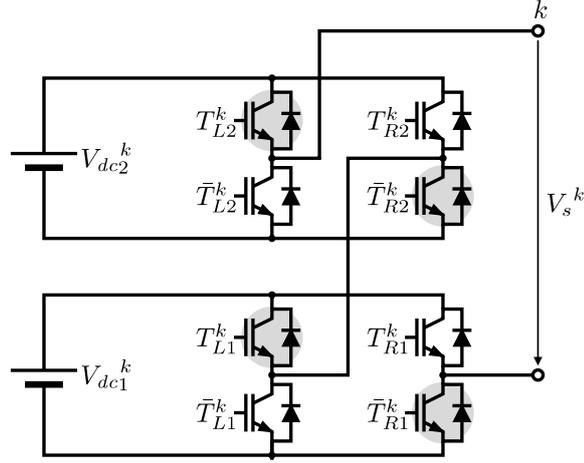


Figure 2.3: Five-level cascaded full-bridge converter.

was extended to include three-phase systems. The cascaded full-bridge converter [128] consists of series full-bridge cells. Since each cell can provide three voltage levels (zero, positive dc voltage, and negative dc voltage), the cells are themselves multilevel converters. A phase of a two-cell series full-bridge inverter is shown in Figure 2.3.

If device voltage drops and conduction losses are neglected then the phase output voltage of one phase is given by

$$V_s^k = \sum_{i=1}^{B^k} (T_{L_i}^k - T_{R_i}^k) V_{dc_i}^k \quad (2.23)$$

where B^k is the number of cascaded cells, $T_{L_i}^k$ and $T_{R_i}^k$ are the trigger signals of the transistor placed left and right, respectively, in cell i , and $V_{dc_i}^k$ is the dc source corresponding to the same cell. The semiconductor blocking voltages can be calculated from the switching states by mean of

$$V_{T_{L_i}^k} = (1 - T_{L_i}^k) V_{dc_i}^k \quad (2.24)$$

$$V_{T_{R_i}^k} = (1 - T_{R_i}^k) V_{dc_i}^k. \quad (2.25)$$

Usually, all independent voltage sources are equal:

$$V_{dc_i}^k = V_{dc}. \quad (2.26)$$

In this case (2.23) can be rewritten as

$$V_s^k = V_{dc} \sum_{i=1}^{B^k} (T_{L_i}^k - T_{R_i}^k). \quad (2.27)$$

Since the result of the summation term is always an integer number, the output voltage is an integer multiple of fixed voltage V_{dc} :

$$V_s^k = v_s^k V_{dc}, \quad \text{where } v_s^k \in \mathbb{Z}. \quad (2.28)$$

TABLE 2.4
FIVE-LEVEL CASCADED FULL-BRIDGE CONVERTER RELATIONSHIPS.

| T_{L1}^k | T_{L2}^k | T_{R1}^k | T_{R2}^k | V_s^k | V_s^k (if $V_{dc_i}^k = V_{dc}$) | v_s^k |
|------------|------------|------------|------------|--------------------------|-------------------------------------|---------|
| 0 | 0 | 1 | 1 | $-V_{dc2}^k - V_{dc1}^k$ | $-2V_{dc}$ | -2 |
| 1 | 0 | 1 | 1 | $-V_{dc2}^k$ | $-V_{dc}$ | -1 |
| 0 | 1 | 1 | 1 | $-V_{dc1}^k$ | $-V_{dc}$ | -1 |
| 0 | 0 | 0 | 1 | $-V_{dc2}^k$ | $-V_{dc}$ | -1 |
| 0 | 0 | 1 | 0 | $-V_{dc1}^k$ | $-V_{dc}$ | -1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | $V_{dc1}^k - V_{dc2}^k$ | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | $V_{dc2}^k - V_{dc1}^k$ | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | V_{dc1}^k | V_{dc} | 1 |
| 1 | 1 | 1 | 0 | V_{dc2}^k | V_{dc} | 1 |
| 1 | 0 | 0 | 0 | V_{dc1}^k | V_{dc} | 1 |
| 0 | 1 | 0 | 0 | V_{dc2}^k | V_{dc} | 1 |
| 1 | 1 | 0 | 0 | $V_{dc2}^k + V_{dc1}^k$ | $2V_{dc}$ | 2 |

Therefore, V_{dc} and v_s^k are the voltage step and the output level of the converter, respectively. Hence, the relationship between the output level v_s^k and the switching state is

$$v_s^k = \sum_{i=1}^{B^k} (T_{Li}^k - T_{Ri}^k). \quad (2.29)$$

Since this topology can generate levels going from $-B^k$ to B^k then the number of different output levels is

$$N^k = 2B^k + 1. \quad (2.30)$$

Thus, if all voltage sources are equal then this kind of converters always provides an odd number of levels. Even so, it is possible to achieve an even number of levels if a half-bridge is added to the chain of full-bridge cells.

Table 2.4 shows the switching signals, output voltages and possible levels for the converter in Figure 2.3.

2.4.1 Switching State Redundancy

The proper switching states to obtain a particular converter output level can be obtained from (2.29). Such equation, like in the flying capacitor topology, has no unique solution. It only establishes the following switching law to select the transistors that must be turned-on:

“The subtraction of the number of independent switches chosen from the left

TABLE 2.5
NUMBER OF PER-PHASE REDUNDANT SWITCHING STATES IN MULTILEVEL CASCADED
FULL-BRIDGE CONVERTERS.

| | $R_d(-4)$ | $R_d(-3)$ | $R_d(-2)$ | $R_d(-1)$ | $R_d(0)$ | $R_d(1)$ | $R_d(2)$ | $R_d(3)$ | $R_d(4)$ |
|-----------|-----------|-----------|-----------|-----------|----------|----------|----------|----------|----------|
| $B^k = 1$ | – | – | – | 1 | 2 | 1 | – | – | – |
| $B^k = 2$ | – | – | 1 | 4 | 6 | 4 | 1 | – | – |
| $B^k = 3$ | – | 1 | 6 | 15 | 20 | 15 | 6 | 1 | – |
| $B^k = 4$ | 1 | 8 | 28 | 56 | 70 | 56 | 28 | 8 | 1 |

side branches of the converter and those that are chosen from the right side has to be equal to the numeric value of the output level.”

There are no restrictions for carrying this selection since (2.29) does not establish priorities among switches.

If (2.1) is substituted in (2.29) then the following equivalent expression is obtained

$$v_s^k + B^k = \sum_{i=1}^{B^k} (T_{Li}^k + \bar{T}_{Ri}^k). \quad (2.31)$$

This new expression provides an alternative law for choosing the switches that must be turned-on:

“Among the switches of the right hand side branches and the complementary ones of the left side branch, a number $B^k + v_s^k$ of independent switches must be selected to obtain a particular output level v_s^k .”

For example, to get the level $v_s^k = 1$ in the converter in Figure 2.3, three switches ($v_s^k + B^k = 1 + 2$) must be turned-on among the highlighted ones. Therefore, $B^k + v_s^k$ switches must be chosen in a set of $2B^k$ elements to obtain the level v_s^k . Hence the number of redundant states corresponding to the level v_s^k is given by

$$R_d(v_s^k) = \binom{2B^k}{B^k + v_s^k} = \frac{(2B^k)!}{(B^k - v_s^k)! (B^k + v_s^k)!}. \quad (2.32)$$

Table 2.5 shows the number of redundant states for each output level in multilevel cascaded full-bridge converters.

The number $S_{w_i \rightarrow j}^k$ of semiconductors that must be switched to change the output from level $v_{s_i}^k$ to level $v_{s_j}^k$ can be obtained by means of

$$S_{w_i \rightarrow j}^k \geq 2|v_{s_i}^k - v_{s_j}^k|. \quad (2.33)$$

2.4.2 Characteristics

Important advantages of this topology are the following:

- The main important feature is its great modularity. Since this topology consists of series-connected power conversion cells, the voltage and power level may be easily scaled adding new cells.

- This topology, like flying capacitor, also presents per-phase redundancy that can be used for dc sources current control and switch loss balancing.
- At expense of reducing per-phase redundancy, the number of output levels can be increased, without adding new components, by using dc sources of different voltages. This approach can be implemented with hybrid device technology where the slower and higher-voltage devices are used to change the output voltage level, while the faster and lower-voltage devices provide the full PWM capability [129–131].

Nevertheless, the fact that the dc-link voltages must be isolated is the major drawback for application of this topology. Several independent dc power supplies are required, which can be provided either by a transformer with multiple isolated secondary windings or by several transformers [129]. Batteries [132], fuel cells [133], capacitors [134], or photovoltaic panels [135, 136] can also be used as independent dc sources.

2.5 Redundant Switching State Selection

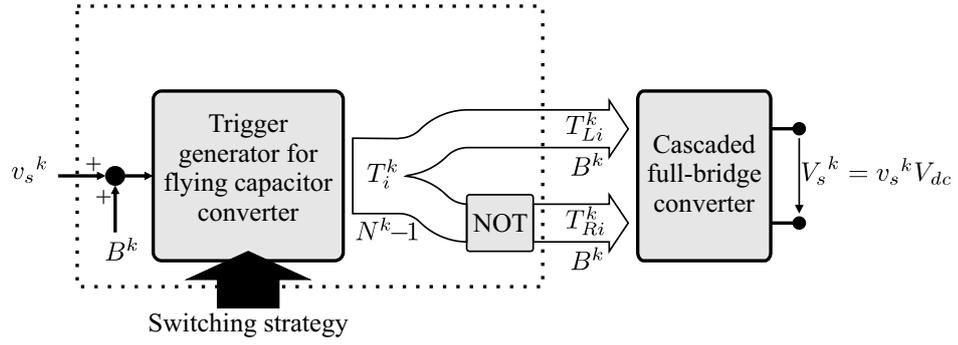
If expressions (2.8), (2.19) and (2.28) are compared then it can be concluded that in the three previous multilevel topologies the output voltage is an integer multiple of a fixed voltage V_{dc} :

$$V_s^k = v_s^k V_{dc}, \quad v_s^k \in \mathbb{Z}. \quad (2.34)$$

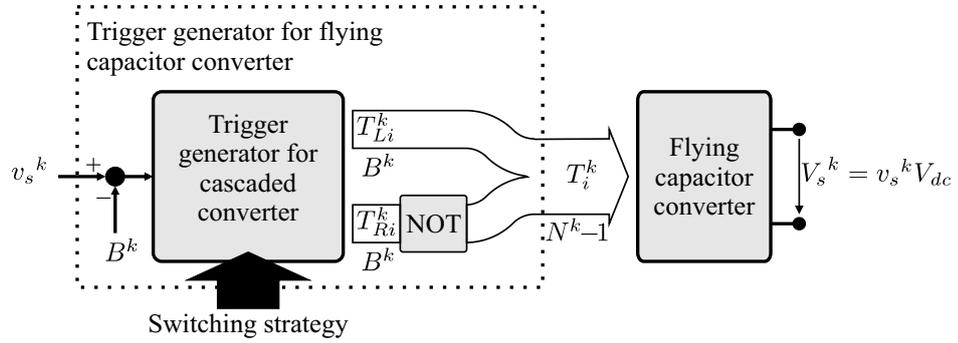
This is a very important property because it makes it possible to use the same SVPWM with the three topologies. In addition, because of this property, the switching vectors $\mathbf{v}_{s1} = [v_s^1, v_s^2, \dots, v_s^P]^T$ and $\mathbf{v}_{s2} = [v_s^1 + n, v_s^2 + n, \dots, v_s^P + n]^T$ provide the same line-to-line voltages; hence, it is also the basis of the joint-phase redundancy in multilevel converters. The new multilevel multiphase SVPWM presented in chapters §4 and §5 are both based on (2.34). Joint-phase redundancy is further investigated in chapter §5 where a new SVPWM algorithm that allows to choose redundant switching states is developed.

Once the converter output level of each phase v_s^k has been obtained from the multilevel modulator, the transistor trigger signals must be determined. As it was shown in sections above, the relationship between output levels and the trigger signal is different for each topology. Therefore, this task must be specifically developed for each topology. In the diode-clamped topology there is no per-phase switching state redundancy. The one-to-one relationship between output levels and trigger signals given by (2.10) can be implemented by means of a lookup table. Consequently, switching strategies in diode-clamped converters can be only implemented by using joint-phase redundancy. In the flying capacitor and cascaded full-bridge topologies, there is no one-to-one correspondence between output levels and trigger signals; therefore, a redundant switching state selection technique is required. Although switching laws of the three topologies are different, they have some similarities that allow establishing some relationship among them.

Flying capacitor converters and cascaded full-bridge converters that have the same number of levels also have the same number of controlled switches, and the number of redundant switching states follows the same profile. In addition, the switching laws, derived from (2.20) and (2.31), have also the same form in both topologies. Consequently, it is possible to adapt the trigger generator of a flying capacitor converter to be used with a cascaded full-bridge converter with the same number of levels, and vice versa.



(a) From a flying capacitor to a cascaded full-bridge converter.



(b) From a cascaded full-bridge to flying capacitor converter.

Figure 2.4: Trigger generator adaptation between flying capacitor and cascaded full-bridge topologies.

If expressions in (2.20) and (2.31) are compared then following corrections are needed to trigger a cascaded full-bridge converter by means of a trigger generator of a flying capacitor converter:

1. The desired output level in the cascaded full-bridge converter v_s^k must be shifted by its number of cells B^k .
2. Next, the trigger signals provided by the generator T_i^k must be divided in two groups:
 - (a) The first group T_{Li}^k will drive the left branch switches of the cascaded full-bridge converter.
 - (b) The inverse signals of the second group T_{Ri}^k will drive the right branch switches of the converter.

The switching laws of both topologies do not make distinctions among switches, therefore it does not matter how the signals are grouped and assigned to the transistors in each cell. Figure 2.4a shows a block diagram of the adaptation technique. Obviously, the switching strategy for capacitor balancing in the flying capacitor topology is useless in a cascaded full-bridge converter; therefore, the switching strategy must be updated taking into account the particular transistor trigger signal assignment. As Figure 2.4b shows, the inverse process is similar:

TABLE 2.6
EQUIVALENCE BETWEEN THE SWITCHING STATES OF THE FLYING CAPACITOR CONVERTER
AND THE CASCADED FULL-BRIDGE CONVERTER.

| Flying capacitor converter | | | | | |
|----------------------------|--------------------------------|------------|------------------|------------------|---------|
| v_s^k | T_1^k | T_2^k | T_3^k | T_4^k | |
| 0 | 0 | 0 | 0 | 0 | -2 |
| 1 | 1 | 0 | 0 | 0 | -1 |
| 1 | 0 | 1 | 0 | 0 | -1 |
| 1 | 0 | 0 | 1 | 0 | -1 |
| 1 | 0 | 0 | 0 | 1 | -1 |
| 2 | 1 | 1 | 0 | 0 | 0 |
| 2 | 1 | 0 | 1 | 0 | 0 |
| 2 | 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 1 | 1 | 0 | 0 |
| 2 | 0 | 1 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 1 | 0 |
| 3 | 1 | 1 | 1 | 0 | 1 |
| 3 | 1 | 1 | 0 | 1 | 1 |
| 3 | 1 | 0 | 1 | 1 | 1 |
| 3 | 0 | 1 | 1 | 1 | 1 |
| 4 | 1 | 1 | 1 | 1 | 2 |
| | T_{L1}^k | T_{L2}^k | \bar{T}_{R1}^k | \bar{T}_{R2}^k | v_s^k |
| | Cascaded full-bridge converter | | | | |

1. The desired output level in the flying capacitor converter v_s^k must be shifted by the negative number of cells $-B^k$.
2. Next, the trigger signals corresponding to the transistors in the right branch of the full-bridge converter T_{Ri}^k must be inverted.
3. Finally, the trigger signals for the flying capacitor converter T_i^k are obtained by grouping all the resulting trigger signals.

It does not matter how the signals are grouped and assigned to the power switches. Table 2.6 verifies the proposed technique with two five-level converters. It compares the trigger signals $\{T_1^k, T_2^k\}$ and $\{T_3^k, T_4^k\}$ corresponding to the flying capacitor converter with trigger signals $\{T_{L1}^k, T_{L2}^k\}$ and $\{\bar{T}_{R1}^k, \bar{T}_{R2}^k\}$ of the cascaded full-bridge converter. Both topologies have the same number of transistors, and the levels that differ by two units ($B^k = 2$) have equivalent switching states. A similar table and the same conclusion would be obtained if other groups of signals were compared.

Expression in (2.9) is the same as (2.20) related to a flying capacitor converter with an additional constraint. Therefore, a trigger generator designed for diode-clamped converters, which implements (2.10), can be used with a flying capacitor converter with the same number of levels. And, consequently, it can also be adapted to be used with a

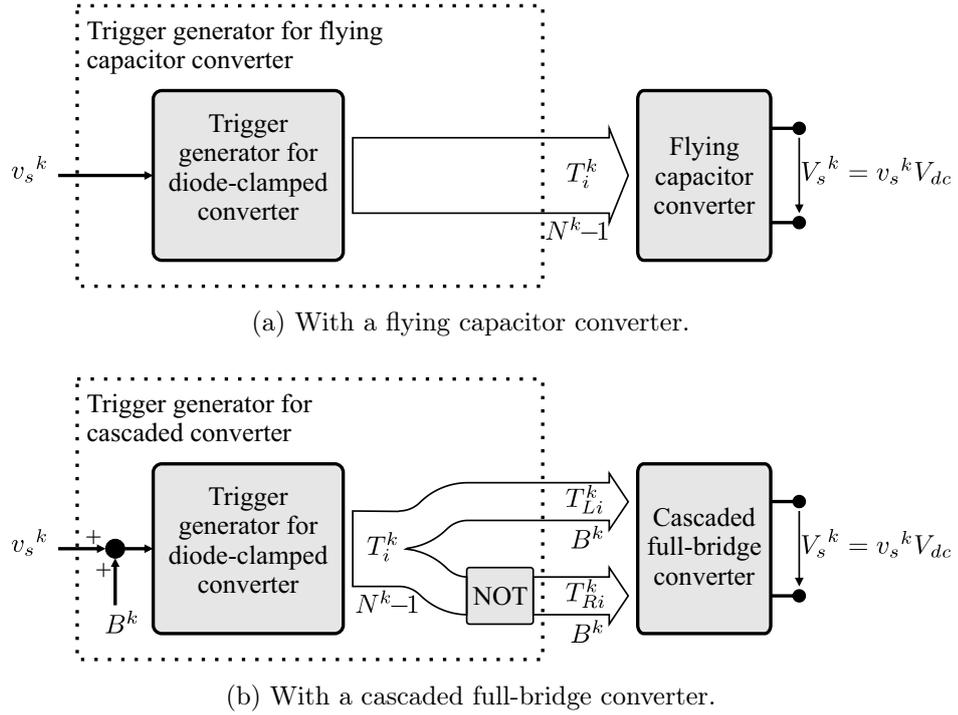


Figure 2.5: Trigger generator for diode-clamped converter used with other topologies.

cascaded full-bridge converter. Nevertheless, since the diode-clamped converter does not have per-phase redundancy, if its trigger generator is used with a flying capacitor or with a cascaded full-bridge converter then it is not possible to take advantage of per-phase redundancy present in those topologies. Hence, capacitor balancing, current control of dc sources and switch loss balancing must be carried out by using alternative methods. As Figure 2.5 shows, a trigger generator for a diode-clamped converter can be directly used with a flying capacitor converter, and it only needs a few adaptations to be used with a cascaded full-bridge converter. The inverse operation is not possible. The trigger signals for a flying capacitor converter or a cascaded full-bridge converter cannot be adapted to a diode-clamped converter because the extra constraint present in the switching law of the latter converter is not observed by the trigger generators designed for the former converters.

Figure 2.6 shows the simulation results of using a trigger generator for a diode-clamped converter with flying capacitor and cascaded full bridge converters. Capacitors have been substituted with dc sources to avoid voltage imbalances. 100 V dc sources have been used. The first plot is the converter output level v_s^k provided by the multilevel modulator. Second plot is the PWM output voltage (thick line) corresponding to a five-level diode-clamped converter. It has been obtained with the trigger signals (thin lines), T_4^k , T_3^k , T_2^k and T_1^k , that have been calculated by means of (2.10). Third plot is the output voltage V_s^k (thick line) of a five-level flying capacitor converter when the trigger generator for the diode-clamped converter is used with it. As Figure 2.5a shows, the same trigger signals (thin lines) are directly used. Last plot is the output voltage V_s^k (thick line) of a five-level cascaded full-bridge converter when the trigger generator for the diode-clamped converter has been adapted to be used with it. The reference output level has been shifted to fit

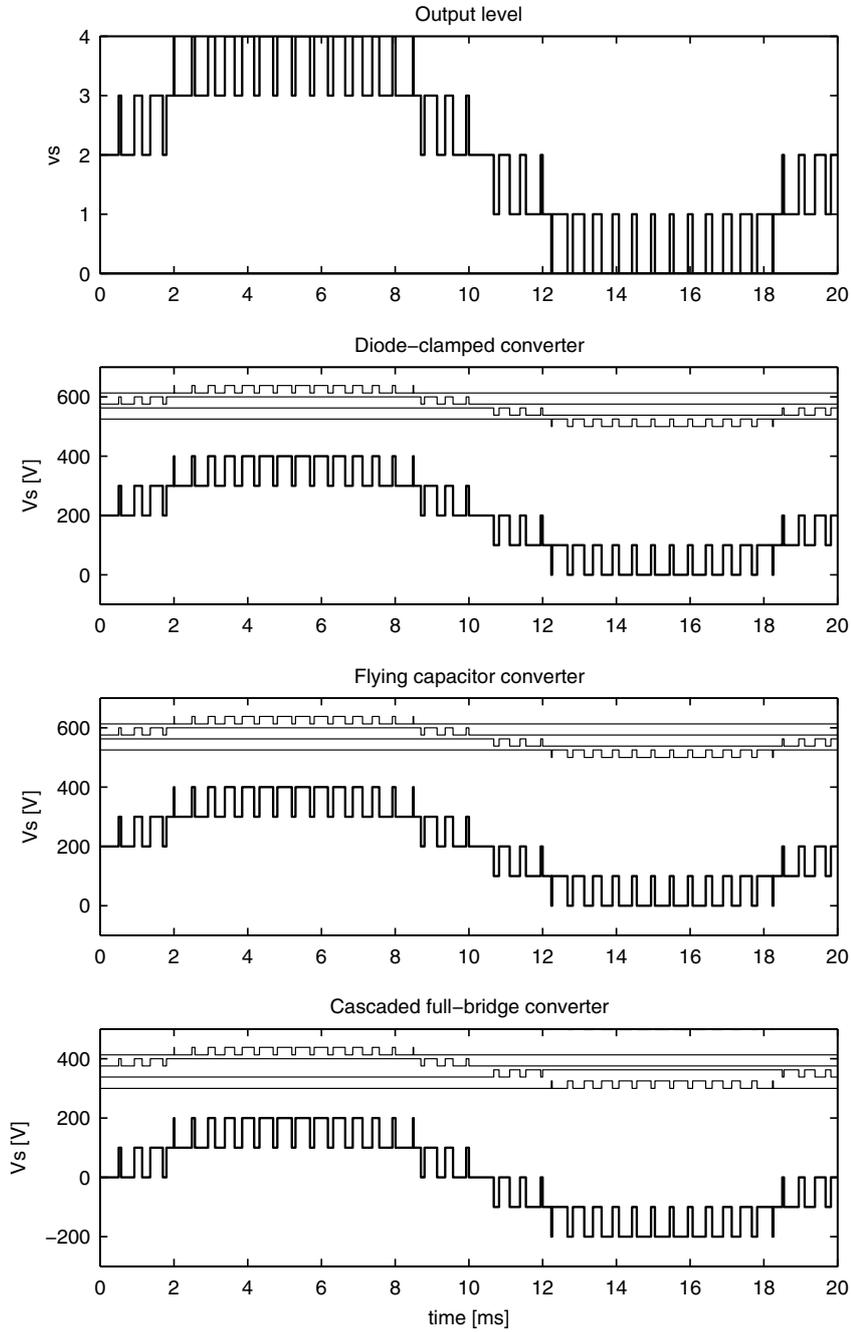


Figure 2.6: Simulation results of using a trigger generator for a diode-clamped converter with a flying capacitor and a cascaded full bridge converter.

the output voltage in the range $[-200\text{ V}, 200\text{ V}]$. The four trigger signals (thin lines), T_{L2}^k , T_{L1}^k , T_{R2}^k and T_{R1}^k , have been adapted using the scheme in Figure 2.5b. The output voltages of all multilevel converters are equal, which validates the proposed adaptation techniques.

2.6 Conclusion

In this chapter the per-phase switching state redundancy in the three main multilevel topologies has been studied. Contributions of the chapter include new expressions to calculate the number of redundant switching states of each output level in the diode-clamped, flying capacitor and cascade full-bridge topologies, and the switching laws for triggering them. Those laws have been used to study the relationships among the trigger generators of the multilevel topologies concluding that:

- The trigger generator of a diode-clamped converter can be used with a flying capacitor or a cascaded full-bridge converter sacrificing their per-phase redundancy. The inverse task is not possible.
- The trigger generator for a flying capacitor converter can be adapted to be used with a cascaded full-bridge converter and vice versa.

Contributions of the chapter have been published in a journal [117].

Chapter 3

Implementation of Multilevel SVPWM Algorithms

***Abstract** — The modulation process in multilevel converters requires controlling a high number of active power switches. Concurrent processing capability of FPGAs makes them suitable for hardware implementation of modern SVPWM algorithms. The chapter addresses the implementation in an FPGA of two very efficient three-phase multilevel SVPWM algorithms. Both algorithms have been first analyzed and later described in VHDL, partly hand-coded and partly automatically generated. Both implementations are compared in terms of complexity and logic resources required. Finally, they are tested with an NPC inverter feeding an induction motor.*

3.1 Introduction

Classical two-level pulse-width modulation (PWM) algorithms can be readily implemented by using on-board PWM channels available in many commercial digital signal processors (DSPs). At present, for more than two-levels, DSPs having appropriate built-in PWM units enough to control the large amount of switches used by multilevel converters are not available in the market. A software implementation of these PWM units is very time consuming and it requires a fast and expensive DSP to carry out the modulation and control of the converter. An architecture where a programmable logic device (PLD) carries out the modulation task and the DSP implements the control strategy is better suited for multilevel converters [9]. Thus, cheap DSP and PLD are needed. Usually, the switching vectors and the switching times are calculated in the DSP. Next, this information is then transferred to a PLD which operates at a higher clock frequency and can make the transitions of the switching state on a nanosecond time scale. In a practical implementation, the DSP and PLD clocks are tied together by a PLD circuit which divides its clock and sends a clock signal to the DSP on the microsecond time scale. Exact details of this process can be found in the literature [80, 138]. Besides the zero-order hold effect, there is a one-sample delay effect which is caused by the fact that the DSP will take some time to determine the levels and switching times. The computed duty cycle is used in the following switching period causing a lag between the duty cycle and the switching state.

Research work included in this chapter has been published in the journal *IEEE Transactions on Industrial Electronics* [137].

This effect is negligible for high switching frequencies.

Among PLDs, the field-programmable gate arrays (FPGAs) reached a level of maturity that made them the choice for implementation in many fields [139]. Recent applications of FPGAs in industrial electronics include mobile robot path planning and intelligent transportation [140, 141], current controlling applied to power converters [142–144], real-time hardware-in-the-loop testing for control design [145], controller implementation [146–149], separating and recovering independent source signals [150], and neural computation [151]. Digital controllers using a microprocessor together with an FPGA have been demonstrated in previous works [152–157]. The implementation in an FPGA of a space vector pulse-width modulation (SVPWM) control for a two-level inverter has been described in detail in [158] and [159]. An FPGA was also proposed in [160] to control a three-level neutral-point clamped (NPC) inverter in wind turbine applications.

The contribution of the chapter is the study of the hardware implementation of two well-known multilevel three-phase SVPWM techniques. One is the three-dimensional (3D) modulation algorithm proposed by Prats et al. in [81], which does not take into account per-phase switching state redundancy; and the other is the two-dimensional (2D) modulation algorithm presented by Celanovic and Boroyevich in [72], which takes it into account. Both are very efficient modulation techniques for multilevel three-phase converters. They calculate switching vectors and dwell times by means of simple addition and comparison operations, without using angles, trigonometric functions or precomputed tables. Such very low complexity and computational cost make them very suitable for real-time implementation in low cost devices. Both SVPWM algorithms are analyzed and described in very-high-speed integrated circuit hardware description language (VHDL) to carry the implemented into an FPGA. Matlab Simulink and System Generator for Simulink tools [161, 162], as well as Foundation ISE tools [163], are used to develop all the tasks involved. Both implementations are compared in terms of complexity and FPGA resources needed. An NPC voltage-source inverter (VSI) is used to test them.

This chapter is organized as follows. Section §3.2 analyzes both SVPWM algorithms. In section §3.3 the algorithms are organized in functional blocks. Section §3.4 describes the hardware implementation of each functional block. Section §3.5 shows the architecture of the experimental setup and gives some testing results. Section §3.6 is the conclusion of the chapter.

3.2 Space Vector PWM Algorithms

Algorithms proposed by Prats et al. [81] and Celanovic and Boroyevich [72] are two SVPWM algorithms which approximate the reference vector $\mathbf{V}_r = [V_r^a, V_r^b, V_r^c]^T$ from a sequence of vectors selected by the discrete set of switching vectors that can be generated by the converter. Both algorithms can be applied to converters with any number of levels in which the output voltage vector $\mathbf{V}_{sj} = [V_{sj}^a, V_{sj}^b, V_{sj}^c]^T$ can be written as:

$$\mathbf{V}_{sj} = \mathbf{v}_{sj} V_{dc}, \quad \mathbf{v}_{sj} \in \mathbb{Z}^3 \quad (3.1)$$

where V_{dc} is the output voltage step of the converter and $\mathbf{v}_{sj} = [v_{sj}^a, v_{sj}^b, v_{sj}^c]^T$ is the switching vector corresponding to the output vector \mathbf{V}_{sj} . This expression is fulfilled for the three multilevel topologies studied in chapter §2; therefore, both modulation algorithms

can be used with the diode-clamped, the flying capacitor and the cascaded full-bridge converters.

The 2D SVPWM algorithm in [72] guarantees that the line-to-line voltage reference is equal to the averaged line-to-line converter voltage output:

$$\begin{bmatrix} V_r^a - V_r^b \\ V_r^b - V_r^c \\ V_r^c - V_r^a \end{bmatrix} = \frac{1}{T} \sum_j \begin{bmatrix} V_{sj}^a - V_{sj}^b \\ V_{sj}^b - V_{sj}^c \\ V_{sj}^c - V_{sj}^a \end{bmatrix} T_j \quad (3.2)$$

where T is the switching period and T_j is the switching time corresponding to \mathbf{V}_{sj} . The homopolar component of the reference is not taken into account; therefore, the line-to-neutral voltage of the reference and the output can be different. Hence, this algorithm is only useful for inverters feeding a load in delta connection or with floating neutral.

The 3D SVPWM algorithm in [81] assures that the line-to-neutral voltage reference and the averaged line-to-neutral converter output voltage are equal:

$$\begin{bmatrix} V_r^a \\ V_r^b \\ V_r^c \end{bmatrix} = \frac{1}{T} \sum_j \begin{bmatrix} V_{sj}^a \\ V_{sj}^b \\ V_{sj}^c \end{bmatrix} T_j. \quad (3.3)$$

This algorithm takes into account the homopolar component of the reference; therefore it is well suited for inverters connected to a load with connected neutral. However, it can be used with the inverter feeding the same loads as the 2D algorithm, but underutilizing the capabilities of the dc bus, because the algorithm does not inject the proper homopolar component which allows extending the modulation index range.

The 2D algorithm takes its name from the fact that the phase vectors can be represented in a 2D space if their homopolar component is obviated. The 2D algorithm by Celanovic and Boroyevich uses the following transformation:

$$\begin{bmatrix} V_r^g \\ V_r^h \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \end{bmatrix} \begin{bmatrix} V_r^a - V_r^b \\ V_r^b - V_r^c \\ V_r^c - V_r^a \end{bmatrix}. \quad (3.4)$$

The set of output vectors that the converter is able to generate inside the control space determines the region where the reference vector can be synthesized. When the reference vector lies outside this control region, it is in the overmodulation zone and it cannot be approximated accurately. Figure 3.1 shows these control regions of both algorithms: a hexagon in the 2D case and a hexahedron in the 3D case.

Since both SVPWM algorithms try to approximate a reference vector by using a set of switching vectors inside the control region, they follow the same main steps:

1. Placing the reference vector in the control space.
2. Searching the nearest space vectors to the reference vector; three vectors in the 2D algorithm and four ones in the 3D algorithm.

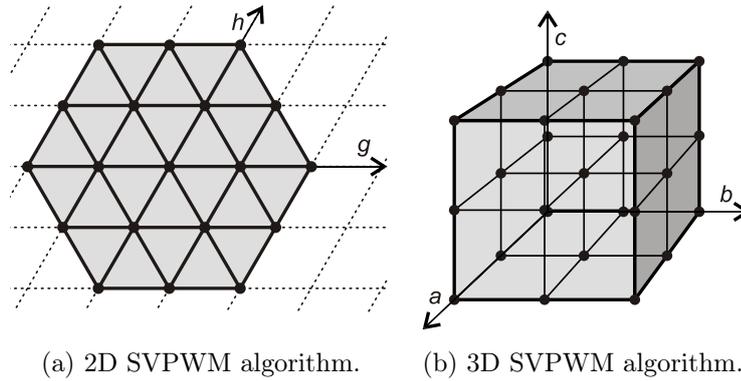


Figure 3.1: Control regions.

3. Calculating the switching time of each switching vector.
4. Translating the switching vectors in trigger signals.
5. Inserting dead times into the trigger signals.

All these steps must be repeated in each switching cycle.

The 3D SVPWM modulation problem has a unique solution, and the 3D algorithm provides this solution with sorted vectors so that the number of switchings between two consecutive elements is minimized. The 2D SVPWM algorithm does not have this characteristic because there are several line-to-neutral voltage combinations that provide the same line-to-line voltage output. That is, each 2D vector in the gh coordinate system corresponds to several switching vectors in the abc coordinate system. Relationship between both frames is given by [72]

$$\begin{bmatrix} v_{s_j}^a \\ v_{s_j}^b \\ v_{s_j}^c \end{bmatrix} = \begin{bmatrix} n \\ n - v_{s_j}^g \\ n - v_{s_j}^g - v_{s_j}^h \end{bmatrix}, \quad n \in \mathbb{Z}. \quad (3.5)$$

Such property allows the 2D SVPWM algorithm to manage joint-phase redundancy by means of the index n . Nevertheless, the redundant switching state selection, i.e., the selection of n , is not addressed in [72]. In addition, the algorithm does not provide the switching vectors in any particular order. Therefore, an algorithm to select and put in order the switching vector sequence is needed.

3.3 Implementation Design

3.3.1 Specifications

Both SVPWM algorithms have been described in VHDL for the NPC inverter in Figure 3.2. The switching frequency of the inverter can be selected in both implementations, as 2.5, 5.0, 7.5 and 10 kHz, by means of the input signal f_s . Vector reference is provided in the dq frame; hence the input signals are the vector components v_r^d and v_r^q and the angle

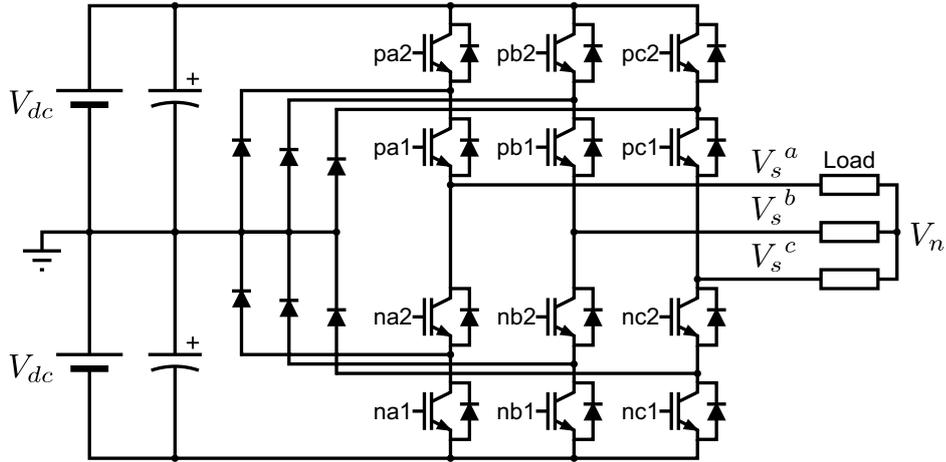


Figure 3.2: Three-phase NPC inverter.

of the rotating frame θ . Finally, the modulator output signals are twelve trigger signals corresponding to the power switches of the NPC inverter. A symmetrical arrangement of the pulses in the switching period is carried out to get a good spectral output.

After analyzing similarities and differences between both algorithms, they have been organized as shown in the block diagrams in Figure 3.3. The functional description of the blocks is detailed below. Bit number of signals and hardware implementation concerning each block will be discussed in the next section.

3.3.2 Description of the 2D SVPWM Algorithm Components

Figure 3.3a shows the block diagram designed for the 2D algorithm, which includes the following components: `freq`, `dq2gh`, `2DSVM`, `gh2abc`, `PWM3`, and `dead_time`.

Component `freq`

The user can select one of four modulator switching frequencies available by means of the input signal f_s . The selected frequency defines the needed values for the switching period T and the dead time T_d . A dead time of 5% of the switching period was used.

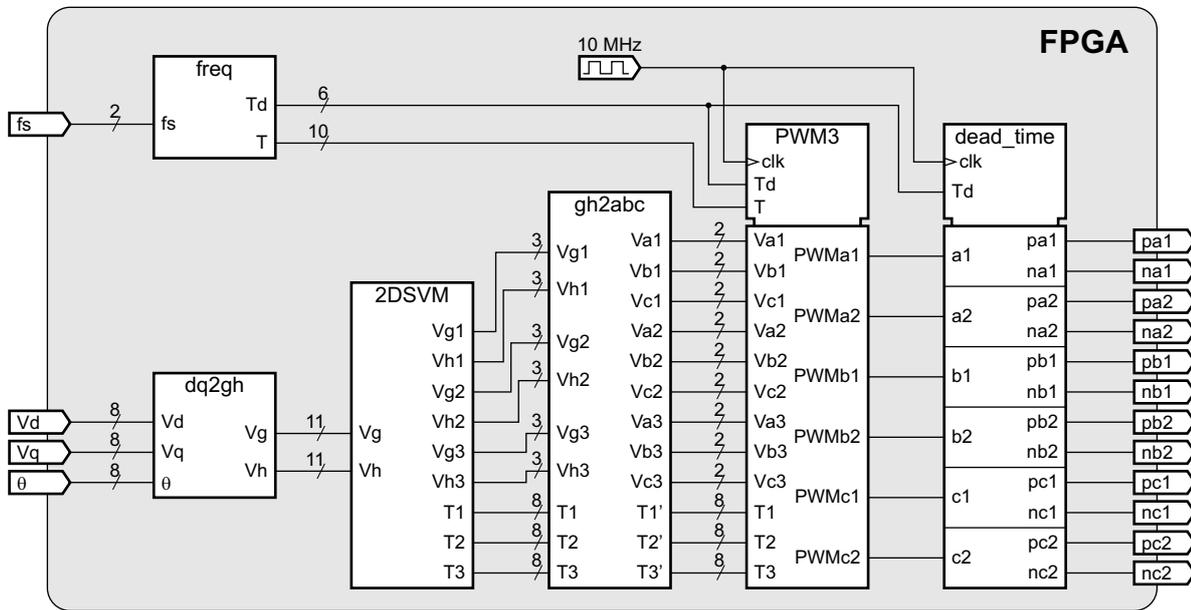
Component `dq2gh`

This element carries out the following dq/gh transformation of the reference vector $[v_r^d, v_r^q]^T$ from the rotational frame to the stationary frame

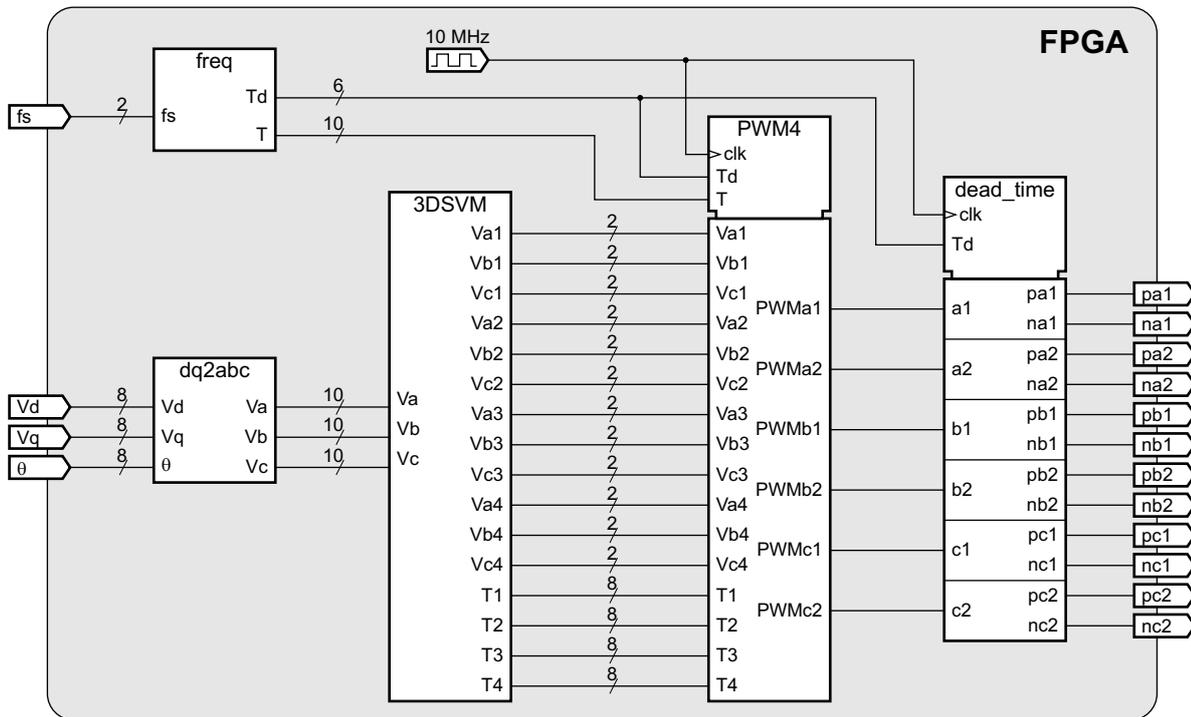
$$\begin{bmatrix} v_r^g \\ v_r^h \end{bmatrix} = \begin{bmatrix} \sqrt{3} & -1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} v_r^d \\ v_r^q \end{bmatrix}. \quad (3.6)$$

Component `2DSVM`

This component is the heart of the system. It determines the nearest three vectors $[v_{s1}^g, v_{s1}^h]^T$, $[v_{s2}^g, v_{s2}^h]^T$ and $[v_{s3}^g, v_{s3}^h]^T$ to the reference vector $[v_r^g, v_r^h]^T$ in the gh frame and calculates their corresponding switching times [72].



(a) 2D SVPWM algorithm.



(b) 3D SVPWM algorithm.

Figure 3.3: Implementation diagrams of the SVPWM algorithms.

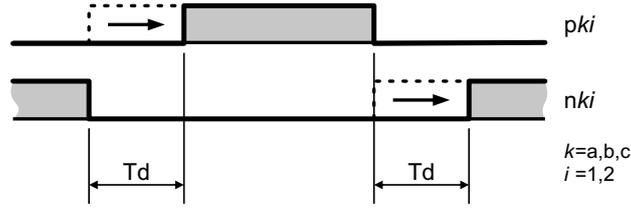


Figure 3.5: Dead time implementation.

$$\begin{aligned} 6\text{th: } & [v_{s2}^a, v_{s2}^b, v_{s2}^c]^T && \rightarrow T_2/2 \\ 7\text{th: } & [v_{s1}^a, v_{s1}^b, v_{s1}^c]^T && \rightarrow T_1/4. \end{aligned}$$

After that, it generates the six PWM signals corresponding to each complementary pair of switches.

Component dead_time

Finally, this circuit generates the trigger signal for each power switch, introducing the proper dead time to give each insulated-gate bipolar transistor (IGBT) enough time to switch off, before its complementary one is switched on. This is done, as shown in Figure 3.5, by delaying the rising edges of the trigger signals by the time T_d .

3.3.3 Description of the 3D SVPWM Algorithm Components

Figure 3.3b shows the simplified block diagram of the 3D SVPWM algorithm. It is composed of the following main components: `freq`, `dq2abc`, `3DSVM`, `PWM4`, and `dead_time`.

Component freq

It is the same block used in the 2D SVPWM algorithm for the selection of the switching frequency that calculates the switching period T and the dead time T_d .

Component dq2abc

This component implements the following dq/abc transformation:

$$\begin{bmatrix} v_r^a \\ v_r^b \\ v_r^c \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} v_r^d \\ v_r^q \end{bmatrix}. \quad (3.7)$$

Component 3DSVM

This component finds the four nearest switching vectors, $[v_{s1}^a, v_{s1}^b, v_{s1}^c]^T$, $[v_{s2}^a, v_{s2}^b, v_{s2}^c]^T$, $[v_{s3}^a, v_{s3}^b, v_{s3}^c]^T$ and $[v_{s4}^a, v_{s4}^b, v_{s4}^c]^T$, to the reference vector and calculates their corresponding switching times. This problem has a unique solution, and the resulting sequence minimizes the number of switchings. Thus, no extra circuits like `gh2abc` are needed.

Component PWM4

This block is very similar to the PWM3 but working with four input vectors instead of three. It arranges the vectors in a symmetrical way into the switching period and generates the six PWM signals corresponding to each complementary pair of IGBTs. The sequence generated is the following:

$$\begin{aligned}
 \text{1st:} & \quad [v_{s1}^a, v_{s1}^b, v_{s1}^c]^T \rightarrow T_1/2 \\
 \text{2nd:} & \quad [v_{s2}^a, v_{s2}^b, v_{s2}^c]^T \rightarrow T_2/2 \\
 \text{3rd:} & \quad [v_{s3}^a, v_{s3}^b, v_{s3}^c]^T \rightarrow T_3/2 \\
 \text{4th:} & \quad [v_{s4}^a, v_{s4}^b, v_{s4}^c]^T \rightarrow T_4 \\
 \text{5th:} & \quad [v_{s3}^a, v_{s3}^b, v_{s3}^c]^T \rightarrow T_3/2 \\
 \text{6th:} & \quad [v_{s2}^a, v_{s2}^b, v_{s2}^c]^T \rightarrow T_2/2 \\
 \text{7th:} & \quad [v_{s1}^a, v_{s1}^b, v_{s1}^c]^T \rightarrow T_1/2.
 \end{aligned}$$

Component dead_time

This is the same component used in the 2D algorithm to generate the twelve trigger signals inserting the corresponding dead times.

3.4 Hardware Implementation

The Digilent S3 board was used to implement both algorithms. This board hosts a XC3S200 FPGA from Xilinx, which has 4.320 logic cells, each constituted by two 16×1 lookup tables (LUTs) and two flip-flops. This FPGA also has twelve 18×18 hardware multipliers, as well as twelve 18 kb block random access memories (BRAMs).

Although FPGAs allow a great degree of parallelism in the implementation of such algorithms, it is necessary to reduce the total amount of hardware needed, keeping the final cost at a reasonable level. Operations are done with a 10 MHz clock, which is sufficiently high frequency to achieve a real-time operation of this algorithm. Area of arithmetic circuits grows exponentially with the number of bits. Trade-off between precision and area led to choose eight bits for the input variables: v_r^d , v_r^q and θ . In the worst case, when the amplitude of the reference is equal to unity, the total error because of the bit truncation is 2.52%.

3.4.1 Component Implementation

The hardware description of the components of the algorithms shown in Figure 3.3 has been hand-coded in VHDL. An exception was made with the components dq2gh and dq2abc. These components have been developed in Simulink with the System Generator libraries provided by Xilinx. The VHDL code that describes these components has been automatically generated using the System Generator tool, in order to produce a correct system implementation in a short design time.

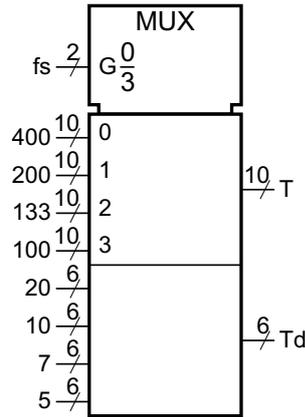


Figure 3.6: Component freq.

Component freq

One of the four frequency and dead time pairs can be selected by means of the two-bit signal f_s . Signals T and T_d are integer numbers that express the switching period and the dead time in microseconds. Signal T is ten bits wide, which allows periods of up to $1023 \mu s$, i.e., 977.5 Hz . T_d is a six-bit signal, which allows defining a dead time up to $63 \mu s$. The modulator switching period T and the corresponding dead time T_d are selected by means of a four-channel multiplexer, as shown in Figure 3.6.

Components dq2gh and dq2abc

They both implement transformations (3.6) and (3.7) from the rotating to the stationary frames. Figure 3.7 shows the Simulink models used to describe the components and to generate the corresponding VHDL description files.

Internal arithmetic operations are done with adequate precision using fixed-point number representation and the two's complement format. The fractional part of the final result was rounded to eight bits in order to calculate the switching times with adequate accuracy in the components 2DSVM and 3DSVM. For an N -level inverter, in the 2D SVPWM algorithm, the integer part of the results takes values from $-(N - 1)$ to $(N - 1)$ while in the 3D SVPWM algorithm it is in the range from 0 to $(N - 1)$. Therefore, three and two bits are necessary to represent the integer part, respectively. Consequently, the output signals in the 2D case are eleven bits wide and the output signals in the 3D case are ten bits wide. Both blocks could be used without any modification for an NPC inverter of up to five levels. If the number of levels increases then more bits would be needed to represent properly these signals.

The sine and cosine operations have been implemented by means of a table stored in a memory. 256 points of the sinusoidal waveform have been stored with eight bits of resolution. In order to store these data a 256×8 memory is needed; therefore, a BRAM was used to take advantage of the FPGA hardware resources.

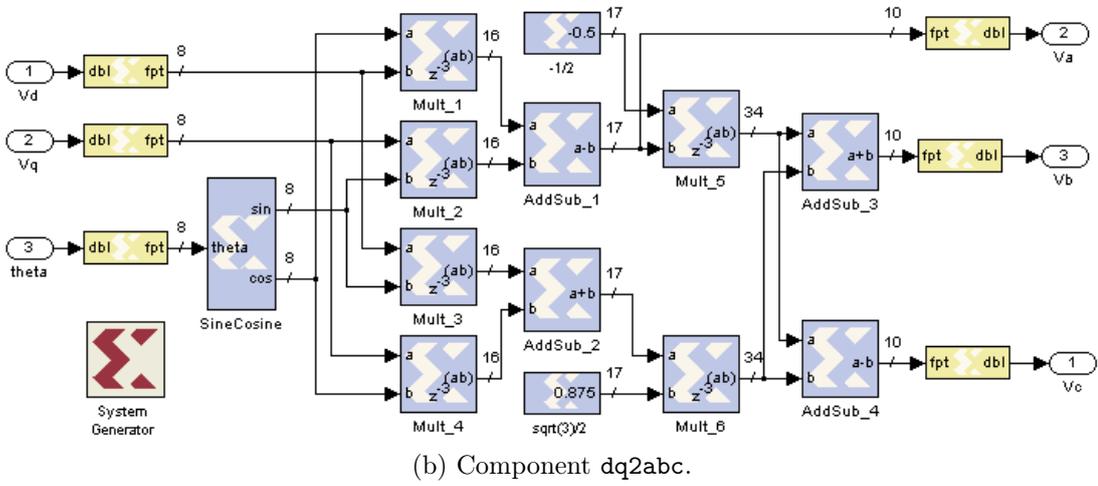
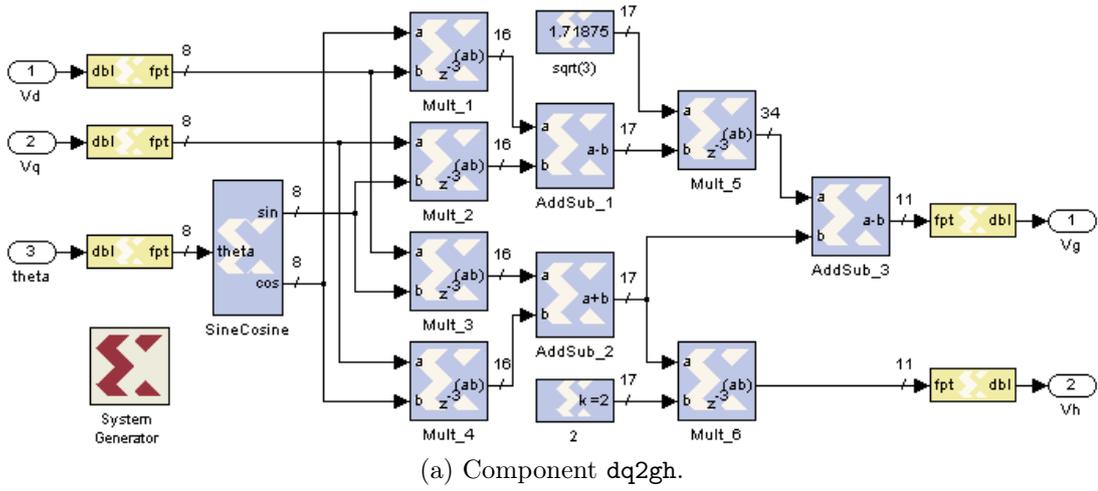


Figure 3.7: Simulink implementation of the transformation components.

Components 2DSVM and 3DSVM

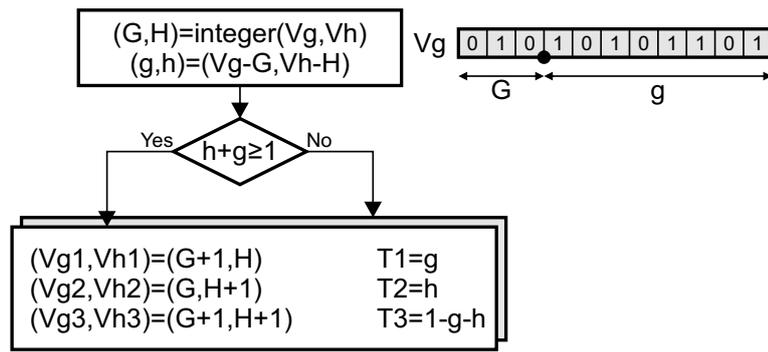
Both elements have been implemented strictly following the algorithm descriptions given in [72] and [81] by means of simple arithmetic and comparison operations.

The nearest vectors and unit switching times are calculated from the integer and the fractional part of the reference, respectively. Therefore, the signals corresponding to the vector components have three bits in the 2D algorithm and two bits in the 3D algorithm; and the unit switching times are eight bits wide.

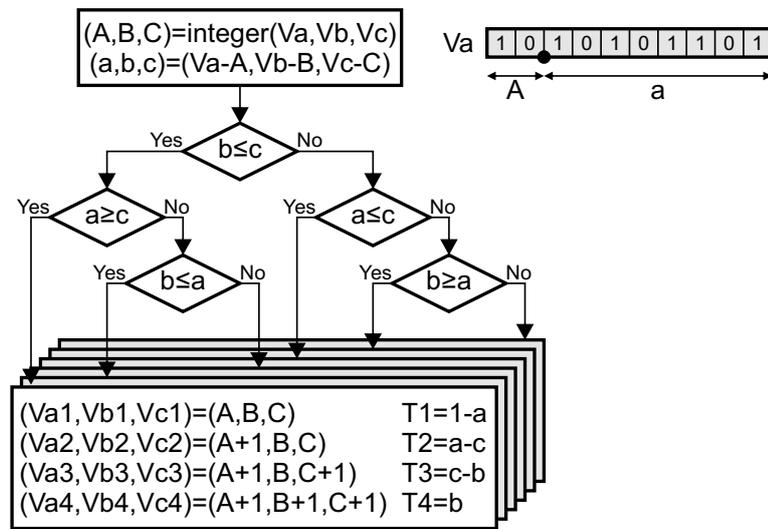
The integer and fractional parts of the references can be done by the proper bit extraction from the fixed point number. Corrections have been done in the case of negative numbers in the 2D algorithm. Figure 3.8 shows the flow diagrams of the VHDL description and the bit extraction operations.

Component gh2abc

Figure 3.9 shows the flow diagram utilized for the VHDL description of this block in accordance with Figure 3.4. As shown previously, the output signals of the component have two bits which are enough to represent the three levels of the inverter.



(a) Component 2DSVM.



(b) Component 3DSVM.

Figure 3.8: Flow diagram of the VHDL description of the modulation components.

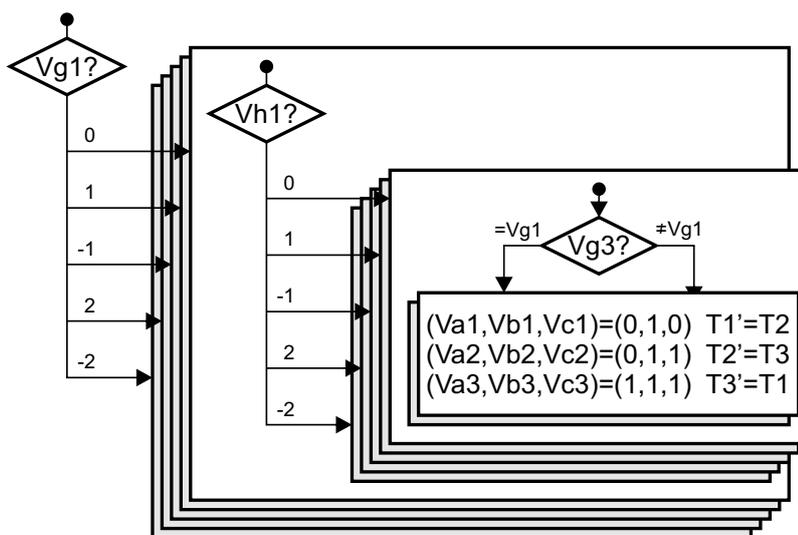


Figure 3.9: Flow diagram of the VHDL description of the gh2abc component.

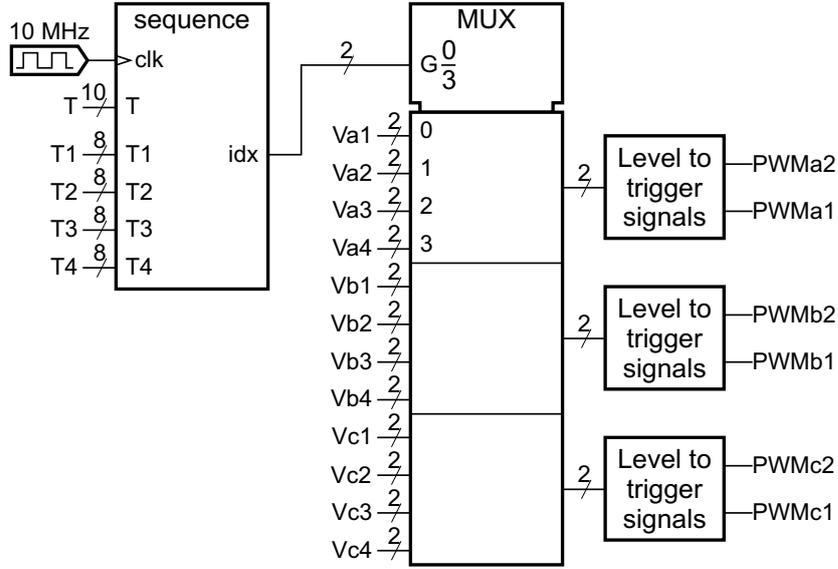


Figure 3.10: Implementation diagram of the component PWM4.

Components PWM3 and PWM4

Figure 3.10 shows the component PWM4 implementation. The implementation of the component PWM3 is very similar. But, in this case unit time T_4 is not needed and the fourth vector is internally generated as

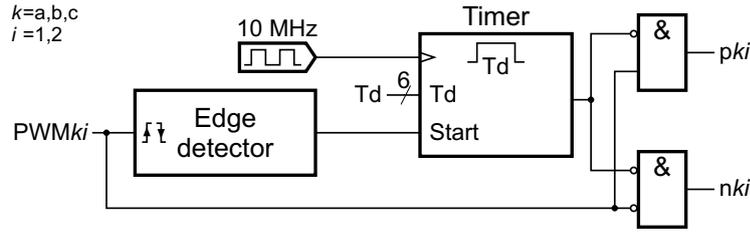
$$[v_{s4}^a, v_{s4}^b, v_{s4}^c]^T = [v_{s1}^a + 1, v_{s1}^b + 1, v_{s1}^c + 1]^T. \quad (3.8)$$

The block **sequence** compares the switching time corresponding to each vector with the value of a counter to generate the vector index (**idx**) corresponding to each time interval. The 50 MHz master clock of the S3 board was divided using the digital clock manager of the FPGA in order to obtain a 10 kHz clock. That clock allows generating the PWM signals with a time precision of $0.1 \mu\text{s}$. Due to the fact that vector times are eight bits wide, then the maximum switching frequency available in this system is 39.2 kHz. Without the frequency division of the master clock, the maximum switching frequency could be increased five times, up to 196 kHz. The output of the multiplexer is the space vector that must be generated by the inverter at a particular time.

The level of each phase in the corresponding trigger signals is translated by the component **Level to trigger signal**, which implements (2.10). As it was shown in the previous chapter, this translation depends on the converter topology; therefore, this block must be redesigned if an inverter different from the NPC (with the same number of levels) is used. It is not necessary to modify the rest of components because both modulation algorithms do not depend on the topology.

Component `dead_time`

This block generates the inverse of the trigger signals adding the corresponding dead time. It was implemented using an edge detector together with a counter working as a timer in order to delay the rising edges of the trigger signals as shown in Figure 3.11.

Figure 3.11: Component `dead_time` description.

3.4.2 Resources

All circuits were combined in a top file according to Figure 3.3. Finally, the whole system was synthesized and implemented in the XC3S200 FPGA through the use of the Xilinx Foundation ISE tools, which are specific for these tasks.

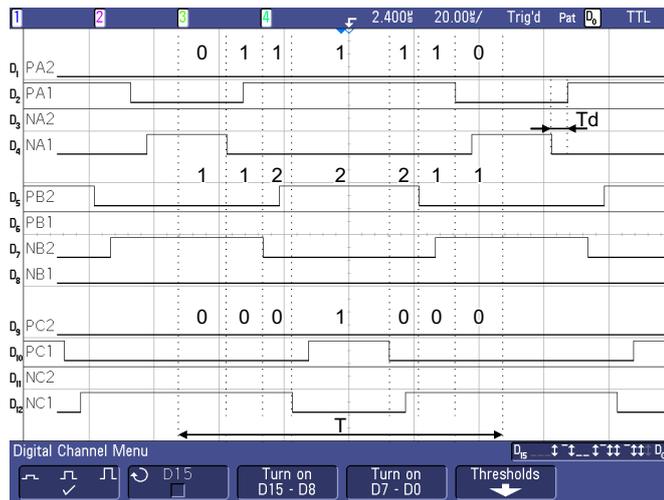
Table 3.1 compares the FPGA logic resources used to implement both algorithms. In our implementation, both algorithms use only one BRAM, to generate the sine and cosine functions, but the 2D algorithm uses half flip-flops and two hardware multipliers less than the 3D algorithm. This is because the first algorithm works with only three 2D vectors, instead of the four 3D vectors of the second algorithm. Although the 2D algorithm works with fewer vectors represented with fewer bits, both implementations use similar number of slices (logic blocks) and LUTs. This is due to the need for an extra task to select the generating vectors in the 2D algorithm (component `gh2abc`). Consequently, the 3D algorithm is easier to implement but uses more logical resources of the FPGA.

In order to extend the proposed implementation to a higher level inverter, many considerations have to be taken into account. The implementation of the component `freq` is independent of the inverter topology; therefore, it can be used without any modification. Components `dq2gh`, `dq2abc`, `2DSVM` and `3DSVM` only need slight modifications in order to adapt the bit number of the signals to the level number of the inverter. Resources used by those components grow in a stepwise manner with the number of levels. In fact, they could be directly used for a five-level topology. Components `PWM3`, `PWM4` and `dead_time` have to be extended in order to generate additional PWM signals for extra IGBTs of the inverter. Therefore, resources used by the additional components grow almost linearly with the number of levels of the NPC inverter. Nevertheless, the component `gh2abc` was specifically designed for a three-level inverter, and therefore it needs a major revision.

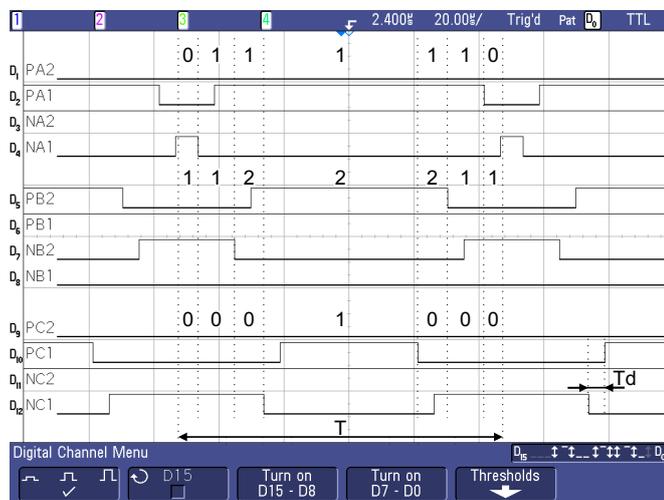
3.5 Experimental Results

Figure 3.12 shows the FPGA outputs, corresponding to the trigger signals for the 2D and 3D algorithms when $v_r^d = 0.6$, $v_r^q = 0$, and $\theta = 100^\circ$. Vector sequence corresponding to this case and the rising edge delay because of the dead time can be seen in the figure.

Both FPGA implementations were tested with a three-level NPC inverter driving a star connected induction motor with isolated neutral. A 220/380 V, 1.420 r/min, 1.35 kW rated motor was used. According with the motor characteristics a dc bus of 300 V was selected. Figure 3.13a shows a diagram of the experimental setup. The inverter is controlled in open loop in the rotating dq frame. The controller was implemented in a DSPACE



(a) 2D SVPWM algorithm.



(b) 3D SVPWM algorithm.

Figure 3.12: Trigger signals.

TABLE 3.1
RESOURCES SUMMARY.

| | | | |
|---|--------------------|------|--|
| Target Device | : xc3s200 | | |
| Number of Slice Flip Flops: | 744 out of 3,840 | 19% | |
| Number of occupied Slices: | 1,547 out of 1,920 | 80% | |
| Total Number 4 input LUTs: | 2,702 out of 3,840 | 70% | |
| Number of bonded IOBs: | 41 out of 173 | 23% | |
| IOB Flip Flops: | 37 | | |
| Number of Block RAMs: | 1 out of 12 | 8% | |
| Number of MULT18X18s: | 10 out of 12 | 83% | |
| Number of GCLKs: | 4 out of 8 | 50% | |
| Number of Startups: | 1 out of 1 | 100% | |
| Total equivalent gate count for design: | 133,126 | | |

(a) 2D SVPWM implementation.

| | | | |
|---|--------------------|------|--|
| Target Device | : xc3s200 | | |
| Number of Slice Flip Flops: | 1,519 out of 3,840 | 39% | |
| Number of occupied Slices: | 1,438 out of 1,920 | 74% | |
| Total Number 4 input LUTs: | 2,120 out of 3,840 | 55% | |
| Number of bonded IOBs: | 41 out of 173 | 23% | |
| IOB Flip Flops: | 37 | | |
| Number of Block RAMs: | 1 out of 12 | 8% | |
| Number of MULT18X18s: | 12 out of 12 | 100% | |
| Number of GCLKs: | 4 out of 8 | 50% | |
| Number of Startups: | 1 out of 1 | 100% | |
| Total equivalent gate count for design: | 145,551 | | |

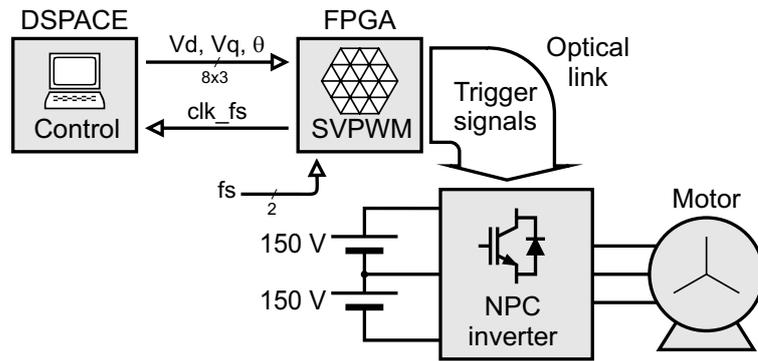
(b) 3D SVPWM implementation.

DS1103 PPC Controller Board. The communication link between the DSPACE board and the FPGA uses a data bus of 3×8 bits. Synchronization is made using the switching frequency clock. Every falling edge, the controller calculates a new set of references and writes them on the bus. New data is read by the FPGA on the following rising edge of the clock. Figure 3.13b shows a photograph of the whole system used in the test, including the power converter, the FPGA board, and a personal computer.

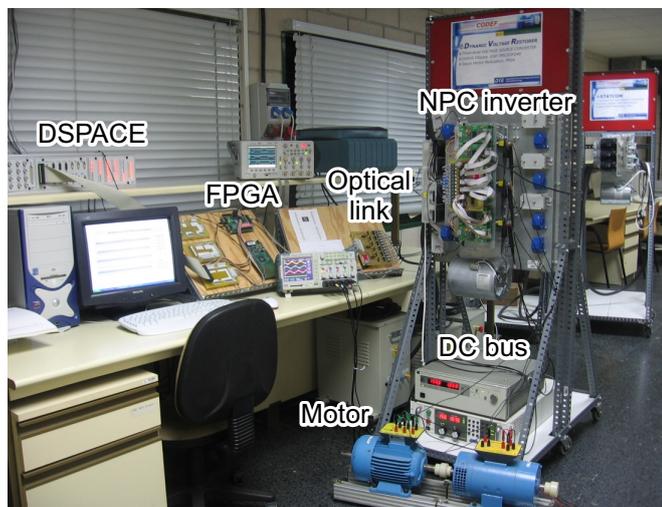
Figure 3.14 depicts line-to-line voltages and load neutral voltages measured with both implementations. The filtered neutral voltage is shown in Figure 3.15. Voltages are measured with respect to the midpoint of the dc bus in the points displayed in Figure 3.2. As expected, the line-to-line voltages are very similar in both cases and differences are in the neutral voltage V_n . The filtered V_n in the 3D SVPWM algorithm is nearly zero because this algorithm does not inject an homopolar component in the phase voltages. On the contrary, the homopolar component injected by the 2D SVPWM algorithm produces voltage fluctuations in the neutral point voltage of the load.

3.6 Conclusion

The contributions of the chapter are the analysis and implementation of a 2D and a 3D multilevel SVPWM algorithms. The 3D technique provides a sorted sequence with four space vectors, which minimizes the number of switchings. Nevertheless, it does not take

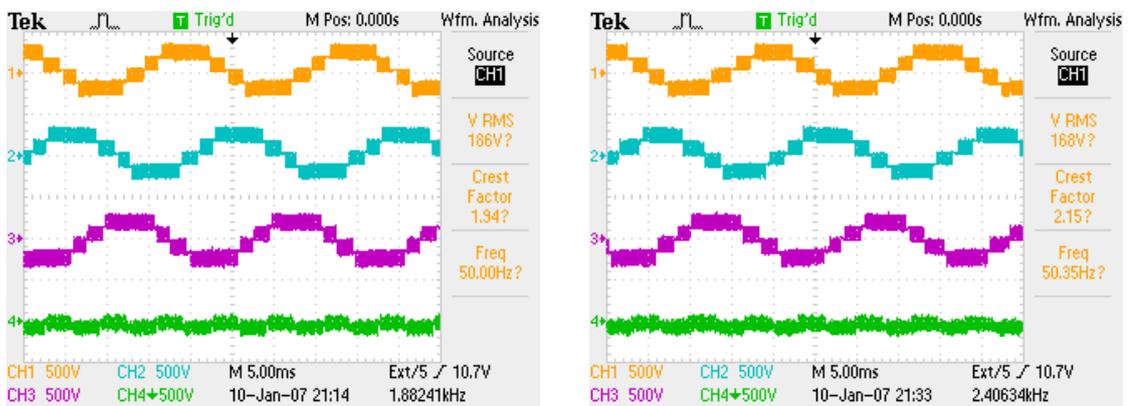


(a) Diagram.



(b) Photograph.

Figure 3.13: Experimental test setup.



(a) 2D SVPWM algorithm.

(b) 3D SVPWM algorithm.

Figure 3.14: Experimental results. Ch1: $V_s^a - V_s^b$; Ch2: $V_s^b - V_s^c$; Ch3: $V_s^c - V_s^a$; Ch4: V_n .

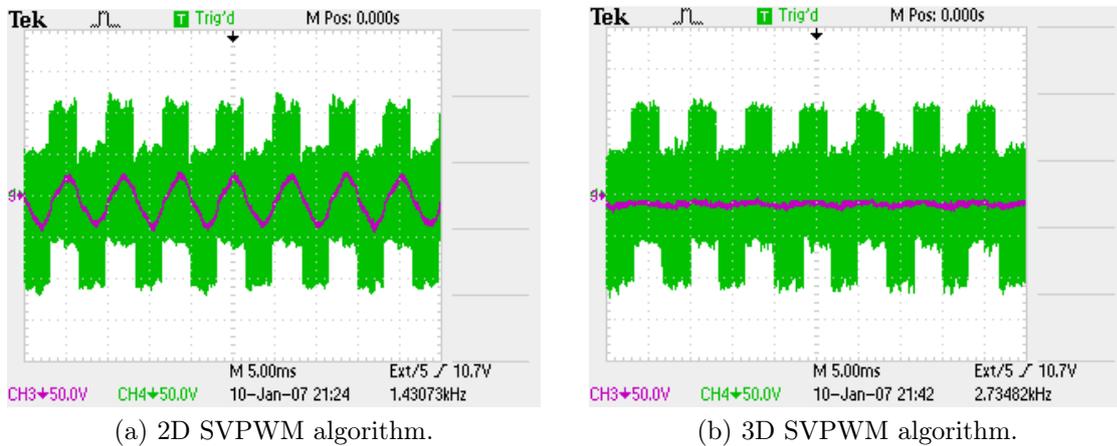


Figure 3.15: Neutral voltage. Ch3: filtered V_n ; Ch4: V_n .

into account per-phase switching state redundancy present in converters supplying loads with isolated neutral point. The 2D technique can manage such redundancy. It provides a sequence with three space vectors, but unlike its counterpart, this sequence does not have a particular order. Consequently, this modulation technique requires selecting the switching state from space vectors, and sorting the switching vector sequence. A new method to achieve this in three-level converters was developed.

Both SVPWM algorithms have been implemented in a FPGA for a three-phase NPC inverter. Since both algorithms follow similar steps, their implementations have been structured using similar functional blocks. The main difference between them is due to the fact that the 2D algorithm requires an extra block to carry out the redundant switching state selection. Hence, the 3D algorithm needs fewer components and it is easier to implement than its counterpart.

Research work included in this chapter has been published in a journal [137].

Chapter 4

Multilevel Multiphase SVPWM Algorithm

Abstract — *The chapter presents a SVPWM algorithm for multilevel multiphase voltage source converters. This algorithm is formulated in a multidimensional space and it is based on a two-level multiphase SVPWM algorithm. This two-level algorithm is also developed in the chapter. The new SVPWM algorithm can be applied to the classical multilevel topologies; it has low computational complexity and it is suitable for online implementation. It was implemented in an FPGA and it was tested in laboratory with a five-level five-phase VSI. Since the algorithm is formulated for a generic number of phases, it was also particularized for the usual three-phase converters with three and four legs.*

4.1 Introduction

In three-phase systems, the space vector pulse-width modulation (SVPWM) technique has a great acceptance and has been deeply studied. In fact, there are available in literature very efficient multilevel algorithms that can be easily implemented in hardware devices [72, 81] as it was shown in chapter §3. The situation is very different in multiphase systems where the SVPWM has not been studied much. In fact there is not available any modulation algorithm for multilevel converters suitable for online implementation.

In this chapter a novel SVPWM algorithm that can be used with any number of levels or phases is developed. This algorithm is the result of the two main contributions of the chapter:

- The demonstration that a multilevel multiphase modulator can be realized from a two-level multiphase modulator.
- The development of a new two-level multiphase SVPWM algorithm.

The two-level concept is not new, it has been successfully used in multilevel three-phase systems [91–95]. The new two-level multiphase SVPWM algorithm has been developed in

The algorithm presented in the chapter has been published in the journal *IEEE Transactions on Industrial Electronics* [164]. Its particularization for three phase converters has been presented at the *IEEE Industrial Electronics Society Conference 2008 (IECON'08)* [165, 166]. A demo of the new modulation technique will be included in the *SimPowerSystems* library [167] of *Simulink* in *Matlab R2009a*.

a multidimensional space [115] to avoid the change of reference carried in algorithms that use the decomposition of voltage vectors in multiple dq planes [96–100]. The combination of both ideas provides a generic algorithm with a low computational cost that can be implemented in real-time.

The new multilevel multiphase SVPWM algorithm was implemented in a low-cost field-programmable gate array (FPGA) and it was verified with a five-level five-phase cascaded full-bridge inverter. Since the algorithm can be used with any number of phases, it can be also applied to the classical three-phase converters. Hence, it was particularized for three-phase converters with three and four legs.

This chapter is organized as follows. Section §4.2 describes the mathematical justification of the SVPWM algorithm in depth. This includes the problem formulation, the demonstration of multilevel modulation in multiphase systems by using the two-level concept and the development of the new two-level multiphase SVPWM algorithm. Additionally, from mathematical treatment, the new multilevel multiphase SVPWM algorithm is obtained. Section §4.3 addresses the practical implementation of the algorithm in a FPGA. In section §4.4, the implementation is verified by comparing experimental measurements with simulation results. Some experimental results are also given so as to evaluate the real performance of the implemented modulation algorithm. In section §4.5, the multilevel multiphase SVPWM algorithm is particularized for three-phase converters and compared with an existing SVPWM technique. In section §4.6, a new SVPWM algorithm for multilevel four-leg converters is presented which is the result of applying the multilevel multiphase technique to a four-phase converter. Section §4.7 includes the conclusions of this chapter.

4.2 Algorithm Development

4.2.1 Algorithm Formulation

Since the switching states of any power converter topology stay at discrete values, the SVPWM is used to approximate a particular output vector \mathbf{V}_s by means of a sequence of switching vectors $\{\mathbf{V}_{s1}, \mathbf{V}_{s2}, \dots, \mathbf{V}_{sl}\}$ during each modulation cycle. In order to achieve a proper synthesis of the output vector \mathbf{V}_s , each switching vector \mathbf{V}_{sj} must be applied during an interval T_j in accordance with the following modulation law:

$$\mathbf{V}_s = \frac{1}{T} \sum_{j=1}^l \mathbf{V}_{sj} T_j \quad (4.1)$$

where the sum of the intervals T_j must be equal to the modulation period T :

$$\sum_{j=1}^l T_j = T. \quad (4.2)$$

The reference vector gathers the reference voltages for all phases of the system, the output vector includes the output voltage of each phase and the switching vectors summarize the

switching states of each phase of the converter

$$\mathbf{V}_r = [V_r^1, V_r^2, \dots, V_r^P]^T \in \mathbb{R}^P \quad (4.3)$$

$$\mathbf{V}_s = [V_s^1, V_s^2, \dots, V_s^P]^T \in \mathbb{R}^P \quad (4.4)$$

$$\mathbf{V}_{sj} = [V_{sj}^1, V_{sj}^2, \dots, V_{sj}^P]^T \in \mathbb{R}^P. \quad (4.5)$$

Therefore, the reference vector, the output vector and the switching vectors belong to the multidimensional space \mathbb{R}^P , where P is the number of phases of the converter. The aim of the modulation algorithm is that the output voltage of the converter follows the reference voltage:

$$\mathbf{V}_r = \mathbf{V}_s. \quad (4.6)$$

As it was shown in §2.5, in the typical multilevel topologies the phase output voltage is an integer multiple of a fixed voltage step V_{dc} . This voltage step together with the switching period can be used to normalize all vectors and switching times:

$$\mathbf{v}_r = \frac{\mathbf{V}_r}{V_{dc}} \in \mathbb{R}^P \quad (4.7)$$

$$\mathbf{v}_s = \frac{\mathbf{V}_s}{V_{dc}} \in \mathbb{R}^P \quad (4.8)$$

$$\mathbf{v}_{sj} = \frac{\mathbf{V}_{sj}}{V_{dc}} \in \mathbb{Z}^P \quad (4.9)$$

$$t_j = \frac{T_j}{T}. \quad (4.10)$$

It is important to remark that the new normalized switching vectors \mathbf{v}_{sj} now belong to the multidimensional space of integer numbers \mathbb{Z}^P . If the above expressions are substituted in (4.1) and (4.2), the modulation law can be rewritten in terms of the new normalized variables as

$$\mathbf{v}_s = \sum_{j=1}^l \mathbf{v}_{sj} t_j, \quad \sum_{j=1}^l t_j = 1. \quad (4.11)$$

Expression in (4.6) can be normalized as well as

$$\mathbf{v}_r = \mathbf{v}_s. \quad (4.12)$$

If (4.11) is substituted in the above equation then the following normalized modulation law is obtained:

$$\mathbf{v}_r = \sum_{j=1}^l \mathbf{v}_{sj} t_j, \quad \sum_{j=1}^l t_j = 1. \quad (4.13)$$

If the reference and the switching normalized vectors are both expressed as $\mathbf{v}_r = [v_r^1, v_r^2, \dots, v_r^P]^T$ and $\mathbf{v}_{sj} = [v_{sj}^1, v_{sj}^2, \dots, v_{sj}^P]^T$ then (4.13) can be rewritten in matrix format as

$$\begin{bmatrix} 1 \\ v_r^1 \\ v_r^2 \\ \vdots \\ v_r^P \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ v_{s1}^1 & v_{s2}^1 & \dots & v_{sl}^1 \\ v_{s1}^2 & v_{s2}^2 & \dots & v_{sl}^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_{s1}^P & v_{s2}^P & \dots & v_{sl}^P \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_l \end{bmatrix}. \quad (4.14)$$

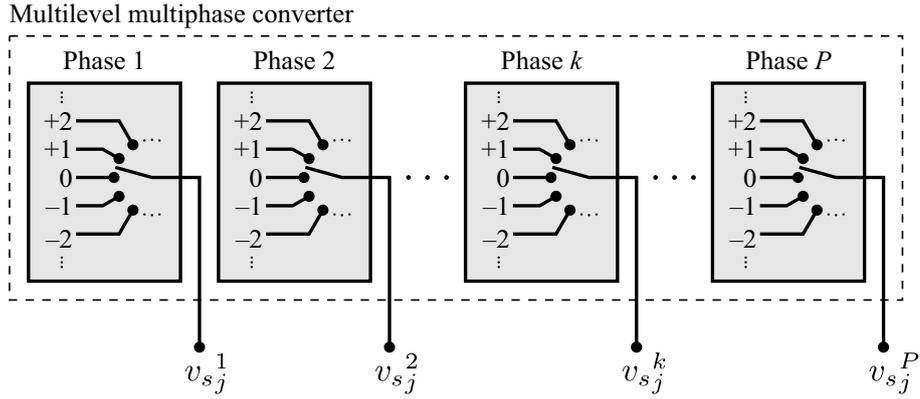


Figure 4.1: Functional diagram of a multilevel multiphase converter.

The above system of linear equations constitutes the modulation law, which must be solved by the multilevel multiphase SVPWM algorithm. The problem solving includes three main steps:

1. Searching a set of integer coefficients for the matrix that permits to solve the linear system.
2. Solving the system of linear equations in order to calculate the switching times.
3. Extracting the switching vector sequence from the coefficient matrix.

The multilevel multiphase SVPWM problem can be simplified if it is decomposed into the sum of a displacement plus a two-level SVPWM problem with the same number of phases.

4.2.2 Algorithm Decomposition

The reference vector can be decomposed into the sum of its integer and fractional parts

$$\mathbf{v}_r = \mathbf{v}_i + \mathbf{v}_f, \quad \mathbf{v}_i = \text{integ}(\mathbf{v}_r) \in \mathbb{Z}^P. \quad (4.15)$$

Components of the new vector \mathbf{v}_i are integer numbers, therefore it belongs to the same space \mathbb{Z}^P of the switching vectors and it could be directly synthesized with one of them. The fractional part \mathbf{v}_f still belongs to the space \mathbb{R}^P and it cannot be directly synthesized by means of a single switching vector. It has to be approximated with a sequence of switching vectors.

Besides, a new set of switching vectors is obtained by displacing all switching vectors the distance given by \mathbf{v}_i

$$\mathbf{v}_{dj} = \mathbf{v}_{sj} - \mathbf{v}_i. \quad (4.16)$$

If those vectors are expressed as

$$\mathbf{v}_i = [v_i^1, v_i^2, \dots, v_i^P]^T \quad (4.17)$$

$$\mathbf{v}_f = [v_f^1, v_f^2, \dots, v_f^P]^T \quad (4.18)$$

$$\mathbf{v}_{dj} = [v_{dj}^1, v_{dj}^2, \dots, v_{dj}^P]^T \quad (4.19)$$

and if (4.16) is substituted in (4.14) the new expression for the modulation law is obtained

$$\begin{bmatrix} 1 \\ v_r^1 \\ v_r^2 \\ \vdots \\ v_r^P \end{bmatrix} = \begin{bmatrix} 0 \\ v_i^1 \\ v_i^2 \\ \vdots \\ v_i^P \end{bmatrix} + \begin{bmatrix} 1 & 1 & \dots & 1 \\ v_{d1}^1 & v_{d2}^1 & \dots & v_{dl}^1 \\ v_{d1}^2 & v_{d2}^2 & \dots & v_{dl}^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_{d1}^P & v_{d2}^P & \dots & v_{dl}^P \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_l \end{bmatrix}. \quad (4.20)$$

Finally, if (4.15) is written as

$$\begin{bmatrix} 1 \\ v_r^1 \\ v_r^2 \\ \vdots \\ v_r^P \end{bmatrix} = \begin{bmatrix} 0 \\ v_i^1 \\ v_i^2 \\ \vdots \\ v_i^P \end{bmatrix} + \begin{bmatrix} 1 \\ v_f^1 \\ v_f^2 \\ \vdots \\ v_f^P \end{bmatrix} \quad (4.21)$$

and if (4.20) and (4.21) are compared, the following relationship between the fractional part of the reference and the displaced switching vectors is obtained:

$$\begin{bmatrix} 1 \\ v_f^1 \\ v_f^2 \\ \vdots \\ v_f^P \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ v_{d1}^1 & v_{d2}^1 & \dots & v_{dl}^1 \\ v_{d1}^2 & v_{d2}^2 & \dots & v_{dl}^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_{d1}^P & v_{d2}^P & \dots & v_{dl}^P \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_l \end{bmatrix}. \quad (4.22)$$

This new system of linear equations presents the same form as the general modulation law in (4.14). However, in this case, the components of vector \mathbf{v}_f are bounded in the interval $[0, 1)$. Therefore, only the subset of displaced vectors with components zero or one is enough to carry out the reference approximation. Consequently, this new equation represents a two-level modulator where the reference vector is \mathbf{v}_f and the array of switching vectors are the displaced set of switching vectors \mathbf{v}_{dj} . Switching times are the same in the multilevel and the two-level modulators. Figure 4.2 shows a two-dimensional (2D) example of the decomposition where vector \mathbf{v}_i coincides with a switching vector and the subset $\{[0, 0], [1, 0], [0, 1], [1, 1]\}$ of displaced vectors is enough to synthesize the fractional part of the reference \mathbf{v}_f .

In summary, (4.20) demonstrates that a multilevel multiphase modulator can be realized from a displacement plus a two-level modulator with the same number of phases. Figure 4.3 shows a block diagram of the proposed technique.

4.2.3 Two-level Multiphase SVPWM Algorithm

Once the multilevel problem has been decomposed, the two-level modulation law (4.22) has to be solved. To obtain an exactly-determined system of linear equations, the coefficient matrix of that modulation law must be a square matrix. Hence, the length l of the

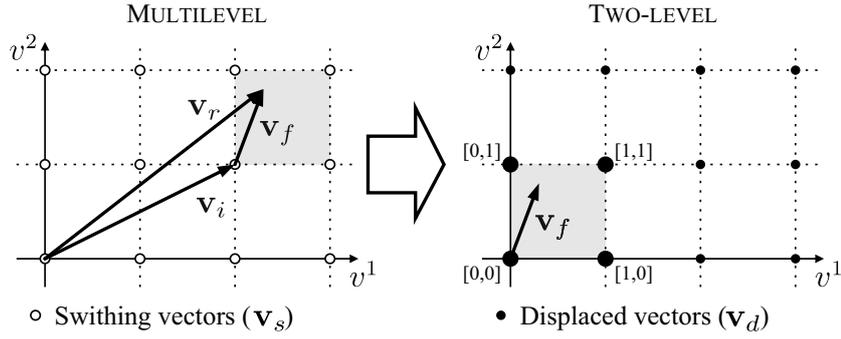


Figure 4.2: Example of decomposition in the 2D problem.

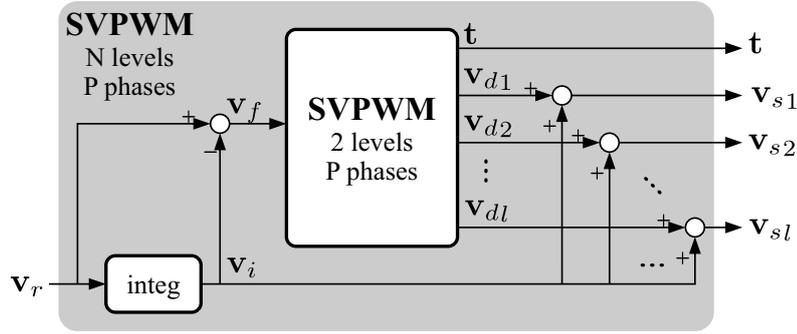


Figure 4.3: Block diagram of the multilevel multiphase SVPWM.

switching vector sequence must be

$$l = P + 1 \quad (4.23)$$

and the particular linear system, which has to be solved, is

$$\begin{bmatrix} 1 \\ v_f^1 \\ v_f^2 \\ \vdots \\ v_f^P \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ v_{d1}^1 & v_{d2}^1 & \dots & v_{dP+1}^1 \\ v_{d1}^2 & v_{d2}^2 & \dots & v_{dP+1}^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_{d1}^P & v_{d2}^P & \dots & v_{dP+1}^P \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_{P+1} \end{bmatrix}. \quad (4.24)$$

The objective of the two-level modulation algorithm is to find out a switching vector sequence; that is, the coefficient matrix of the system (4.24) should be filled with zeros and ones, thus allowing a subsequent system solution. Moreover, the coefficient selection must be carried out taking into account that the switching times must be always positive after the system solution.

There are many different possibilities to fill the coefficient matrix. Nevertheless, the whole power system performance depends on the method employed for calculating the coefficient matrix. In this way, switching losses are minimized if coefficients are selected in such a way that consecutive switching vectors of the switching sequence are adjacent. In other words, only one coefficient is different in two consecutive matrix columns. One possible method for calculating such matrix is detailed below.

Equation (4.24) can be written in a shorter form as

$$\begin{bmatrix} 1 \\ \mathbf{v}_f \end{bmatrix} = \mathbf{D}\mathbf{t}. \quad (4.25)$$

Finding a permutation matrix \mathbf{P} that puts the elements of the reference vector \mathbf{v}_f in descending order

$$\mathbf{P} \begin{bmatrix} 1 \\ \mathbf{v}_f \end{bmatrix} = \begin{bmatrix} 1 \\ \hat{\mathbf{v}}_f \end{bmatrix} \quad (4.26)$$

where

$$1 > \hat{v}_f^1 \geq \dots \geq \hat{v}_f^{k-1} \geq \hat{v}_f^k \geq \dots \geq \hat{v}_f^P \geq 0 \quad (4.27)$$

and multiplying both sides of (4.25) by this permutation matrix \mathbf{P} , we obtain the following equation:

$$\begin{bmatrix} 1 \\ \hat{\mathbf{v}}_f \end{bmatrix} = \hat{\mathbf{D}}\mathbf{t} \quad (4.28)$$

where

$$\hat{\mathbf{D}} = \mathbf{P}\mathbf{D}. \quad (4.29)$$

One coefficient matrix $\hat{\mathbf{D}}$ with adjacent consecutive columns that makes this new system of linear equations exactly determined is the following upper triangular matrix:

$$\hat{\mathbf{D}} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ & 1 & 1 & \dots & 1 \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & 1 \\ 0 & & & & 1 \end{bmatrix}. \quad (4.30)$$

As it will be shown below, the switching times obtained with this coefficient matrix are always positive.

A permutation matrix is an orthogonal matrix so \mathbf{P} is invertible and

$$\mathbf{P}^{-1} = \mathbf{P}^T. \quad (4.31)$$

Therefore, the coefficient matrix \mathbf{D} of the two-level modulation law can be obtained by solving (4.29) as

$$\mathbf{D} = \mathbf{P}^T \hat{\mathbf{D}}. \quad (4.32)$$

The permutation matrix \mathbf{P} applies a set of elementary row-switching transformations to the column vector \mathbf{v}_f . In the same manner, the inverse set of elementary row-switching transformations is applied to the matrix $\hat{\mathbf{D}}$ by the matrix \mathbf{P}^T and, consequently, the number of ones and zeros in each column does not change. Hence, the switching number is minimized because consecutive vectors of the sequence are still adjacent after the transformation.

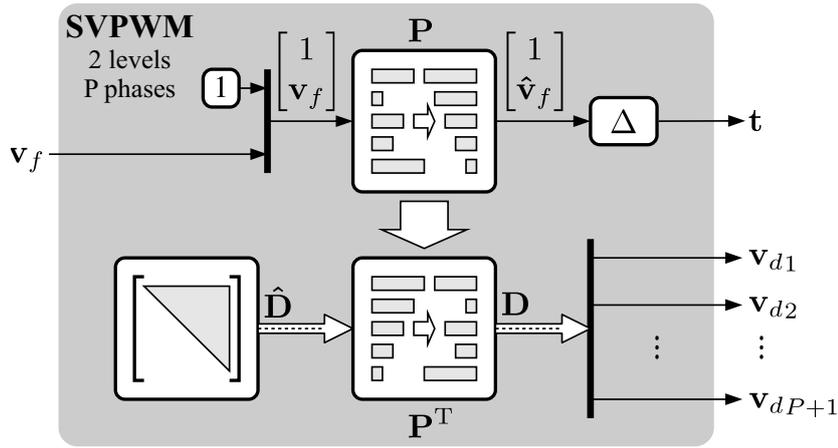


Figure 4.4: Block diagram of the two-level multiphase SVPWM.

Due to the fact that the solution \mathbf{t} is the same for both linear systems, (4.25) and (4.28), it can be calculated by using either of them. The second option seems the best choice because, in this case, the solution is trivial as shown below

$$t_j = \begin{cases} 1 - \hat{v}_f^1, & \text{if } j = 1 \\ \hat{v}_f^{j-1} - \hat{v}_f^j, & \text{if } 2 \leq j \leq P \\ \hat{v}_f^P, & \text{if } j = P + 1. \end{cases} \quad (4.33)$$

All intervals calculated by means of the above expression will always be positive numbers because the coordinates of the vector \hat{v}_f obey (4.27).

In summary, matrix \mathbf{D} permits to solve the two-level modulation law getting positive switching times and minimizing the number of switchings. The two-level switching sequence can be directly extracted from columns of that matrix. Figure 4.4 shows the block diagram of the proposed two-level multiphase SVPWM.

4.2.4 Multilevel Multiphase SVPWM Algorithm

The proposed multilevel multiphase SVPWM algorithm is derived from the previous mathematical treatment. The steps of this algorithm, summarized in the flow chart in Figure 4.5, are:

1. Calculate the normalized reference \mathbf{v}_r from the reference voltage vector \mathbf{V}_r by using the expression in (4.7).
2. Decompose the normalized reference into the sum of its integer part \mathbf{v}_i and its fractional part \mathbf{v}_f by means of (4.15).
3. Sort in descending order the components of vector \mathbf{v}_f to obtain vector $\hat{\mathbf{v}}_f$ and calculate the permutation matrix \mathbf{P} .
4. Rearrange the rows of matrix $\hat{\mathbf{D}}$ to obtain the new matrix \mathbf{D} by means of (4.32).
5. Extract the displaced switching vectors \mathbf{v}_{dj} from the matrix \mathbf{D} by taking into account the expression in (4.24).

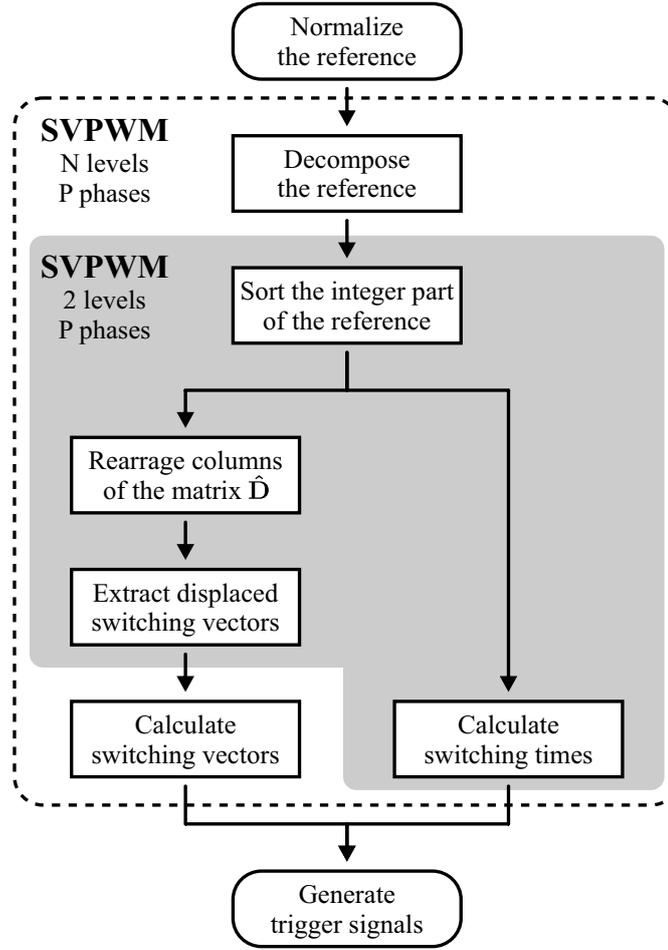


Figure 4.5: Algorithm flow chart.

6. Switching vectors \mathbf{v}_{sj} must be calculated by adding the integer part of the reference \mathbf{v}_i to the displaced switching vectors \mathbf{v}_{dj} according to expression in (4.16).
7. Calculate the switching time t_j corresponding to each switching vector, from the components of $\hat{\mathbf{v}}_f$, by using (4.33).

Finally, trigger signals have to be generated from the switching vectors and the switching times. The relationship between switching states and the particular trigger signals of transistors depends on the multilevel topology as it was shown in §2.

4.2.5 Example

The simplicity of the algorithm is here shown by means of an example in which the steps of the previous flow chart are followed. Let us consider a multiphase drive where the voltage reference for each phase k is purely sinusoidal

$$V_r^k = A \sin \left(\omega t + 2\pi \frac{k-1}{P} \right), \quad k = 1, \dots, P. \quad (4.34)$$

If a voltage amplitude $A = 32$ V and an angular frequency $\omega = 2\pi 50$ rd/s is considered, the instantaneous reference for a five-phase drive when $t = 3.51$ ms is

$$\mathbf{V}_r = [28.6, 22.6, -14.6, -31.6, -5.0]^T \text{ V.} \quad (4.35)$$

If (4.23) is taken into account, the switching sequence will have six switching vectors: \mathbf{v}_{s1} , \mathbf{v}_{s2} , \mathbf{v}_{s3} , \mathbf{v}_{s4} , \mathbf{v}_{s5} and \mathbf{v}_{s6} . From (4.7), if the voltage step of the converter is $V_{dc} = 20$ V, then normalized voltage reference is

$$\mathbf{v}_r = \frac{\mathbf{V}_r}{V_{dc}} = [1.43, 1.13, -0.73, -1.58, -0.25]^T. \quad (4.36)$$

By means of (4.15), this vector is decomposed into an integer and a fractional part

$$\mathbf{v}_i = \text{integ}(\mathbf{v}_r) = [1, 1, -1, -2, -1]^T \quad (4.37)$$

$$\mathbf{v}_f = \mathbf{v}_r - \mathbf{v}_i = [0.43, 0.13, 0.27, 0.42, 0.75]^T. \quad (4.38)$$

If the elements of the vector \mathbf{v}_f are sorted out in descending order the following vector is obtained:

$$\hat{\mathbf{v}}_f = [0.75, 0.43, 0.42, 0.27, 0.13]^T. \quad (4.39)$$

In accordance with (4.26), the permutation matrix that carries out the above sorting operation is

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \quad (4.40)$$

The coefficient matrix \mathbf{D} is calculated using the expression in (4.32) as

$$\mathbf{D} = \mathbf{P}^T \hat{\mathbf{D}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}. \quad (4.41)$$

The displaced switching vectors can be extracted from this matrix by means of the expression in (4.24):

$$\begin{aligned} \mathbf{v}_{d1} &= [0, 0, 0, 0, 0]^T \\ \mathbf{v}_{d2} &= [0, 0, 0, 0, 1]^T \\ \mathbf{v}_{d3} &= [1, 0, 0, 0, 1]^T \\ \mathbf{v}_{d4} &= [1, 0, 0, 1, 1]^T \\ \mathbf{v}_{d5} &= [1, 0, 1, 1, 1]^T \\ \mathbf{v}_{d6} &= [1, 1, 1, 1, 1]^T. \end{aligned} \quad (4.42)$$

From (4.16), the final switching sequence can be calculated by adding the shifting vector \mathbf{v}_i to those vectors

$$\begin{aligned}
\mathbf{v}_{s1} &= \mathbf{v}_i + \mathbf{v}_{d1} = [1, 1, -1, -2, -1]^T \\
\mathbf{v}_{s2} &= \mathbf{v}_i + \mathbf{v}_{d2} = [1, 1, -1, -2, 0]^T \\
\mathbf{v}_{s3} &= \mathbf{v}_i + \mathbf{v}_{d3} = [2, 1, -1, -2, 0]^T \\
\mathbf{v}_{s4} &= \mathbf{v}_i + \mathbf{v}_{d4} = [2, 1, -1, -1, 0]^T \\
\mathbf{v}_{s5} &= \mathbf{v}_i + \mathbf{v}_{d5} = [2, 1, 0, -1, 0]^T \\
\mathbf{v}_{s6} &= \mathbf{v}_i + \mathbf{v}_{d6} = [2, 2, 0, -1, 0]^T.
\end{aligned} \tag{4.43}$$

As expected, consecutive vectors of the sequence are adjacent. Therefore, the number of switchings is minimized.

Finally, the switching times are calculated from the ordered reference vector $\hat{\mathbf{v}}_f$ by means of the expression in (4.33):

$$\begin{aligned}
t_1 &= 1 - \hat{v}_f^a = 0.25 \\
t_2 &= \hat{v}_f^a - \hat{v}_f^b = 0.32 \\
t_3 &= \hat{v}_f^b - \hat{v}_f^c = 0.01 \\
t_4 &= \hat{v}_f^c - \hat{v}_f^d = 0.15 \\
t_5 &= \hat{v}_f^d - \hat{v}_f^e = 0.14 \\
t_6 &= \hat{v}_f^e = 0.13.
\end{aligned} \tag{4.44}$$

4.2.6 Algorithm Features

The computational cost of the presented SVPWM algorithm is low and it is independent of the number of levels. However, it grows slightly with the number of phases because the vector \mathbf{v}_f includes more components which must be sorted out. Besides, lookup tables, trigonometric functions or memories to store predefined switching sequences are not needed. Hence, the algorithm is well suited for real-time implementation in low-cost devices.

Several previous two-level multiphase SVPWM algorithms only use a subset of the switching vectors [168] for practical reasons because of the high number (2^P) of available switching vectors in multiphase systems. Nevertheless, in the proposed modulation technique, even with the higher number (N^P) of available switching vectors in multilevel multiphase converters, all space vectors are handled by the algorithm without discarding any of them. In addition, the provided switching vector sequence is such that it minimizes the number of switchings. Hence no extra effort is needed to achieve this significant goal.

If the modulation index m is defined as the ratio of the peak fundamental of the output voltage to the dc voltage step

$$m = \frac{V_{\text{fund}}}{V_{dc}} \tag{4.45}$$

and if harmonic injection is not considered then the modulation index has a range of

$$0 \leq m \leq \frac{N-1}{2}. \tag{4.46}$$

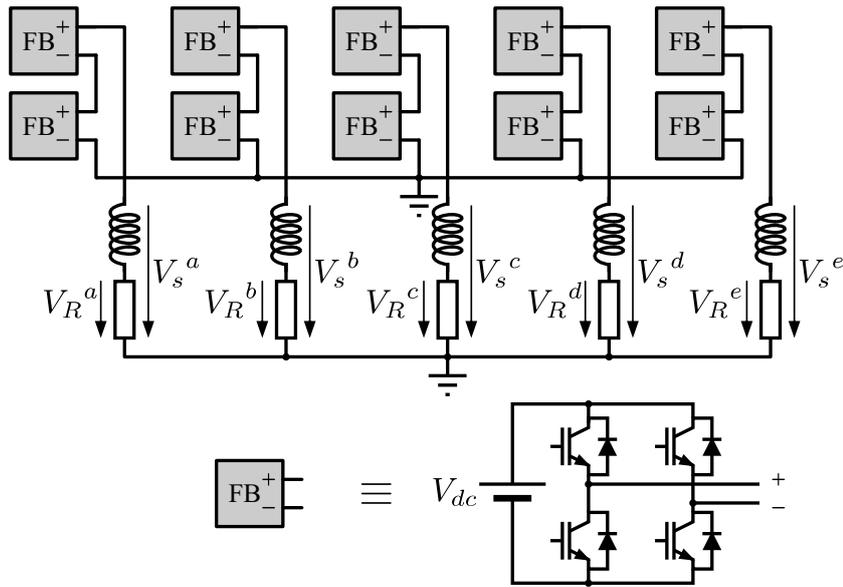


Figure 4.6: Five-level five-phase cascaded full-bridge inverter.

4.3 Algorithm Hardware Implementation

The Digilent S3 board, which hosts a XC3S200 FPGA from Xilinx, was used to implement the new SVPWM algorithm for the five-level five-phase cascaded full bridge inverter in Figure 4.6. Before the very-high-speed integrated circuit hardware description language (VHDL) description was carried out, the model of the hardware implementation was first tested by simulation, using the Simulink model in Figure 4.7. The algorithm implementation follows the flow chart in Figure 4.5. The integer part of the reference \mathbf{v}_i is calculated by the block `Data Type Conversion`. The fractional part of the reference \mathbf{v}_f feeds the block `SVPWM 2L 5P`. This block is a two-level five-phase modulator that provides the displaced switching vectors \mathbf{v}_{dj} and the switching times \mathbf{t} . Switching vectors \mathbf{v}_{sj} that form the final switching sequence are calculated by adding the integer part of the reference to each displaced switching vector. Switching times are simply those ones provided by the two-level modulator block.

Although the matrix approximation of the two-level problem is useful for the algorithm demonstration, it is inefficient in the hardware implementation to calculate the permutation matrix and the matrix operations. Therefore, the \mathbf{P} calculation and the operations made with it were replaced by sorting algorithms that provide the same result. In addition, the first row of matrices $\hat{\mathbf{D}}$ and \mathbf{D} , which are also useful in the algorithm demonstration, was not taken into account in the implementation because these rows are always constant and they are not needed for extracting the switching vectors \mathbf{v}_d . The implementation of the block `SVPWM 2L 5P` is detailed at the bottom of Figure 4.7 where the block `Sort` calculates the ordered vector $\hat{\mathbf{v}}_f$ and the vector of indices `Idx` that summarizes the permutations carried out in the sorting process of \mathbf{v}_f . The vector $\hat{\mathbf{v}}_f$ is required to calculate the switching times. The indices `Idx` are used by the block `Sort (inverse)` to select the rows of the matrix $\hat{\mathbf{D}}$ (without the first row) provided by the block `Triangular matrix`. The output of the `Selector` block is the two-level switching vector sequence.

Finally, the algorithm was described in VHDL following the block diagram of Simulink.

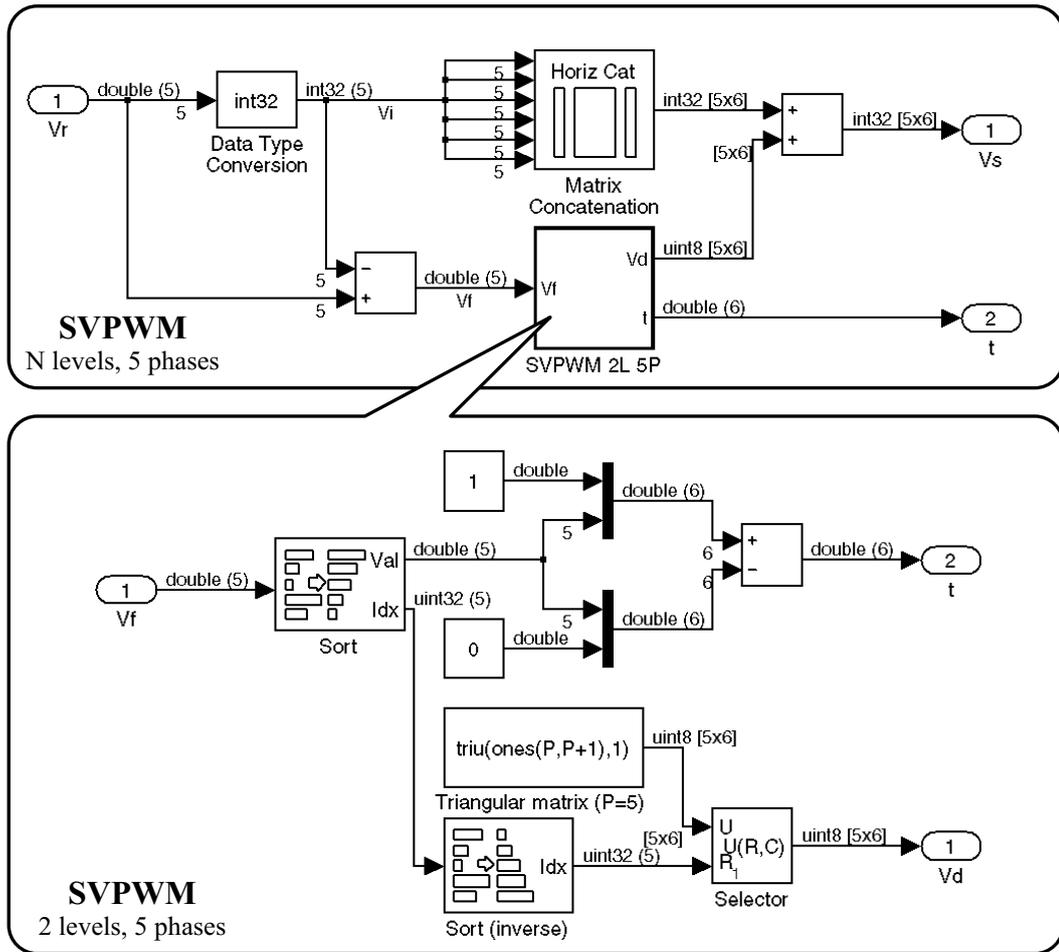


Figure 4.7: Simulink model of the hardware implementation of the SVPWM algorithm.

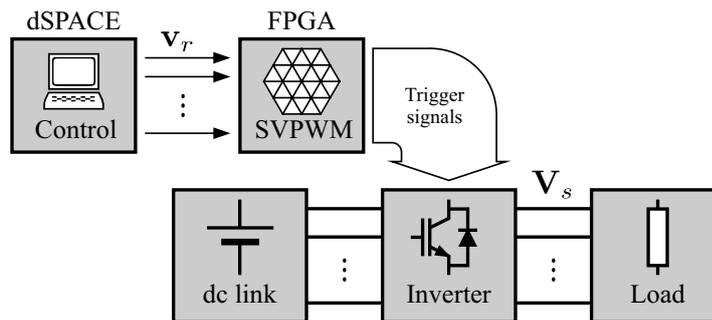
An extra block was added to translate the switching vectors into the trigger signals for controlling the five-level cascaded full-bridge inverter. Table 4.1 shows the summary of resources used by the implementation. It is important to remark that block random access memories (BRAMs) and multipliers available in the FPGA were not used because algorithm does not need data storage or multiplication operations.

4.4 Experimental Results

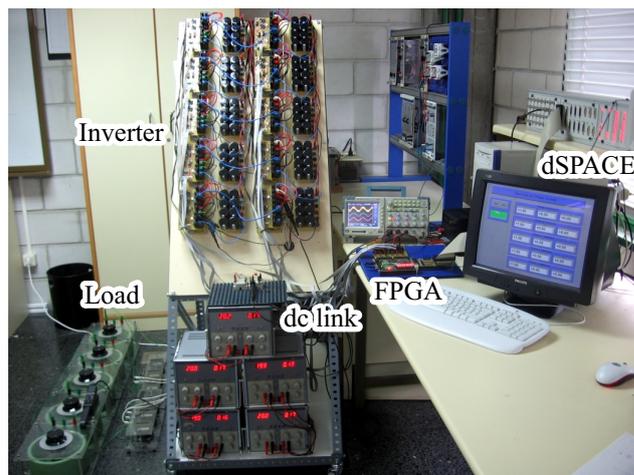
The SVPWM algorithm was tested by using a low-power laboratory prototype. Figure 4.8 shows the block diagram and a photograph of the experimental setup that includes the FPGA, a dSPACE platform, the inverter and the load. The DSPACE DS1103 PPC Controller Board provides the reference vectors to the FPGA. The trigger signals generated by the FPGA control the transistors of the multilevel inverter. The five-level five-phase cascaded full-bridge inverter shown in Figure 4.6, with 3125 different switching states, was used in the experiments. The dc source voltage of all full-bridge cells is 20 V; therefore the inverter voltage step V_{dc} is 20 V as well. A 100 Ω resistive load with a series connected 15 mH inductance was used in tests. The five-phase load is star-connected with

TABLE 4.1
RESOURCES SUMMARY.

| | | | |
|---|--------------------|------|--|
| Target Device | : xc3s200 | | |
| Number of Slice Flip Flops: | 2,523 out of 3,840 | 65% | |
| Number of 4 input LUTs: | 2,718 out of 3,840 | 70% | |
| Number of occupied Slices: | 1,918 out of 1,920 | 99% | |
| Total Number 4 input LUTs: | 2,762 out of 3,840 | 71% | |
| Number of bonded IOBs: | 99 out of 173 | 57% | |
| IOB Flip Flops: | 75 | | |
| Number of Block RAMs: | 0 out of 12 | 0% | |
| Number of MULT18X18s: | 0 out of 12 | 0% | |
| Number of GCLKs: | 8 out of 8 | 100% | |
| Number of Startups: | 1 out of 1 | 100% | |
| Total equivalent gate count for design: | 38,765 | | |



(a) Diagram



(b) Photograph

Figure 4.8: Experimental test setup.

its neutral point connected to the neutral point of the inverter.

The simulation model of the experimental setup was done in Simulink and it includes the algorithm implementation previously shown in Figure 4.7. The inverter and the load were modeled using the SimPowerSystems toolbox [167]. Figure 4.9 compares simulation results with experimental measurements. A low switching frequency (2 kHz) was selected to make the comparison easier. To observe the behavior of the modulator with a generic input, a 50 Hz unbalanced reference with a fifth harmonic was considered. Figure 4.9 shows a good agreement between the simulation model and the experimental setup. Measurements of voltage across load resistances and inverter output are very similar to simulation results, except for some lost switching pulses due to the dead time included in trigger signals which was not considered in the simulation model. To test the performance of the proposed SVPWM algorithm, the case of sinusoidal output voltage with harmonic injection, typically used in concentrated winding ac machines for torque enhancement, was considered. In all the tests made, the voltage reference had a 50 Hz fundamental frequency and the output switching frequency was 10 kHz. Four cases were considered in tests: two cases with purely sinusoidal output voltage with normalized amplitudes $m_1 = 1.8$ and $m_1 = 0.8$ and two additional cases where a third harmonic with magnitude $m_3 = m_1/6$ has been injected.

Figure 4.10 shows the inverter voltage and phase current waveforms, besides the low-order voltage harmonics of the inverter output. Voltages have been measured in the points displayed in Figure 4.6 taking the neutral voltage as reference. The first channel of the oscilloscope shows the inverter output waveform, the second channel shows the filtered inverter output waveform and the third one shows the phase current. In cases 4.10a and 4.10b, the modulation index is high and the modulation algorithm takes advantage of all five levels of the inverter. Nevertheless, in cases 4.10c and 4.10d the modulation index is low and the output voltage is a three-level waveform. In Figure 4.10a and 4.10c, with purely sinusoidal output when the modulation index is high, the low-order harmonics are negligible and the total harmonic distortion (THD) is 3.8%. If the modulation index is low, then the low-order harmonics grow because of the three-level output and the THD increases up to 6.4%. In the cases of Figure 4.10b and 4.10d, the amplitude of the third harmonic is nearly the sixth part of the fundamental, and the high THD obtained corresponds to the injected third harmonic.

Figure 4.11 shows the trajectories of the inverter output voltage and the load current vectors in stationary dq frames [98] with a balanced sinusoidal reference ($m_1 = 1.8$ and $m_3 = 0$). Both vectors move, at constant speed, along a circular trajectory in the $dq1$ plane. No third harmonic was injected; hence, as expected, vectors in the $dq3$ plane stay close to the origin.

4.5 Application to Three-Phase Three-Leg Converters

4.5.1 3D SVPWM Algorithm

The modulation algorithm for three-phase converters is obtained by making $P = 3$ in the multiphase algorithm. Therefore, the modulation problem is formulated in a three-

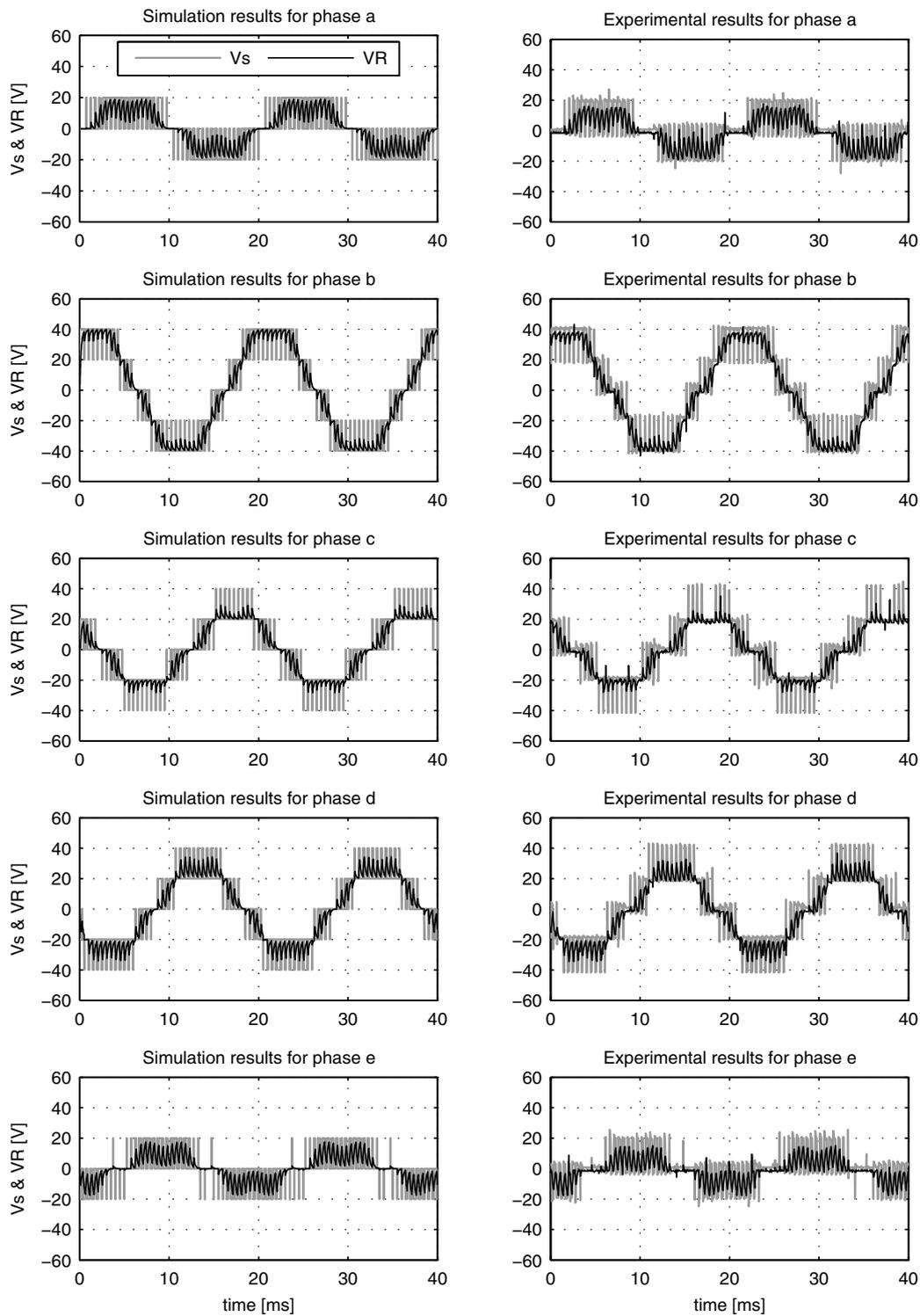
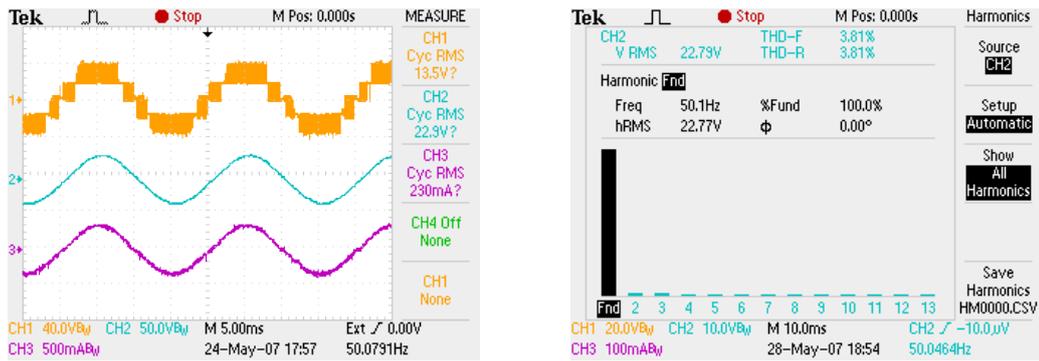
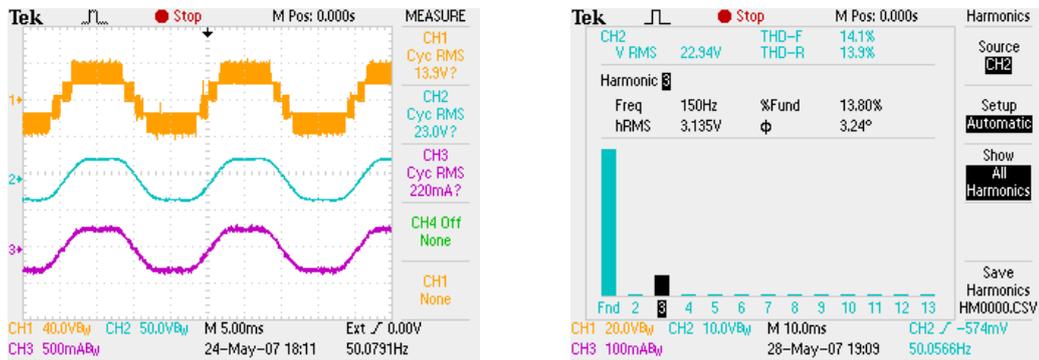


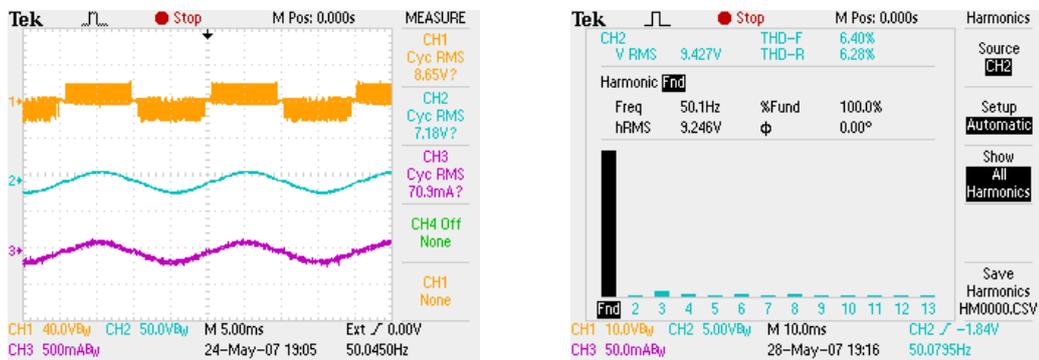
Figure 4.9: Comparison of experimental measurements with simulation results (neutrals of the load and the inverter connected).



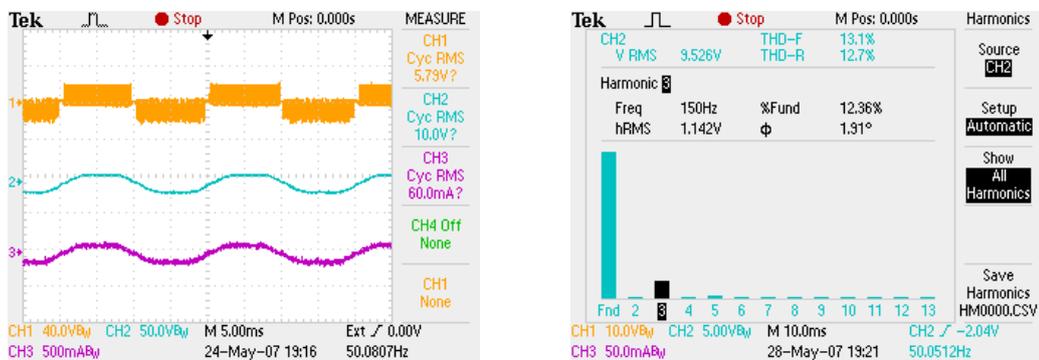
(a) $m_1 = 1.80, m_3 = 0.00$



(b) $m_1 = 1.80, m_3 = 0.30$



(c) $m_1 = 0.80, m_3 = 0.00$



(d) $m_1 = 0.80, m_3 = 0.13$

Figure 4.10: Experimental results. Ch1: inverter output voltage V_s^a ; Ch2: filtered inverter output voltage; Ch3: phase current I_s^a (neutrals of the load and the inverter connected).

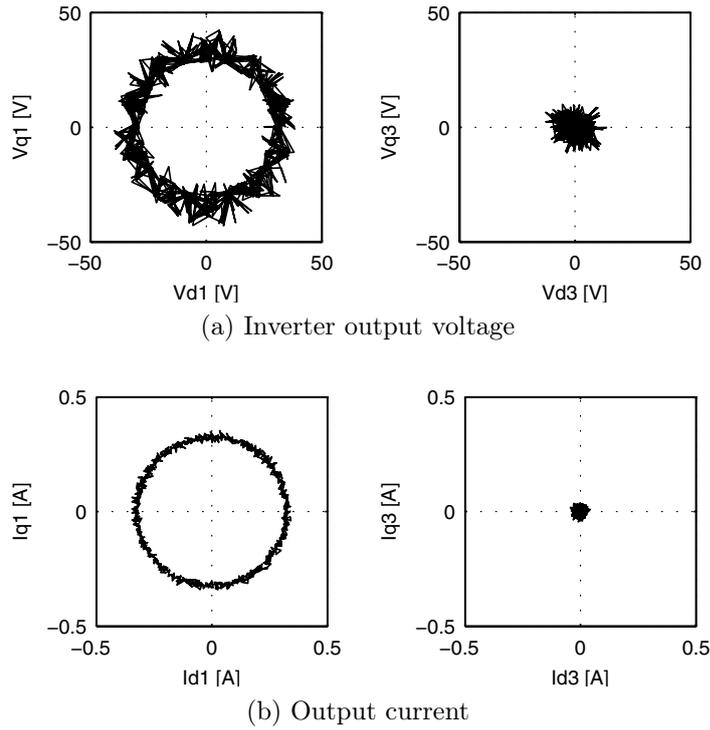


Figure 4.11: Trajectories of the output voltage and the current vectors in stationary dq axes.

dimensional (3D) space and the switching states are 3D integer vectors:

$$\mathbf{v}_{sj} = [v_{sj}^a, v_{sj}^b, v_{sj}^c]^T \in \mathbb{Z}^3. \quad (4.47)$$

Taking into account (4.7), the normalized voltage reference \mathbf{v}_r is the 3D real vector:

$$\mathbf{v}_r = \frac{\mathbf{V}_r}{V_{dc}} = [v_r^a, v_r^b, v_r^c]^T \in \mathbb{R}^3. \quad (4.48)$$

In accordance with (4.15) and (4.16), the integer and fractional parts of the reference vector are

$$\mathbf{v}_i = \text{integ}(\mathbf{v}_r) = [v_i^a, v_i^b, v_i^c]^T \in \mathbb{Z}^3 \quad (4.49)$$

$$\mathbf{v}_f = \mathbf{v}_r - \mathbf{v}_i = [v_f^a, v_f^b, v_f^c]^T \in \mathbb{R}^3. \quad (4.50)$$

From \mathbf{v}_f , the permutation matrix \mathbf{P} can be determined by testing the three following logical conditions:

$$\begin{aligned} C_{ab} &= [v_f^a \geq v_f^b] \\ C_{bc} &= [v_f^b \geq v_f^c] \\ C_{ca} &= [v_f^c \geq v_f^a]. \end{aligned} \quad (4.51)$$

Table 4.2 shows the relationship between the results of these three conditions and the permutation matrix \mathbf{P} . Since cases 000 and 111 are not coherent, there are only six

TABLE 4.2
PERMUTATION MATRIX IN THE 3D SVPWM ALGORITHM.

| C_{ab} | C_{bc} | C_{ca} | Ordered vector $\hat{\mathbf{v}}_f$ | Matrix \mathbf{P} |
|----------|----------|----------|---|--|
| 0 | 0 | 1 | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$ |
| 0 | 1 | 0 | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ |
| 0 | 1 | 1 | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$ |
| 1 | 0 | 0 | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ |
| 1 | 0 | 1 | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ |
| 1 | 1 | 0 | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ |

different situations that must be taken into account. Next, the matrix \mathbf{D} is calculated from the upper triangular matrix $\hat{\mathbf{D}}$ in (4.30) by means of (4.32) as

$$\mathbf{D} = \mathbf{P}^T \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.52)$$

The sequence of displaced vectors $\{\mathbf{v}_{d1}, \mathbf{v}_{d2}, \mathbf{v}_{d3}, \mathbf{v}_{d4}\}$ is extracted from the matrix \mathbf{D}

taking into account the expression in (4.24):

$$\mathbf{D} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ v_{d1}^a & v_{d2}^a & v_{d3}^a & v_{d4}^a \\ v_{d1}^b & v_{d2}^b & v_{d3}^b & v_{d4}^b \\ v_{d1}^c & v_{d2}^c & v_{d3}^c & v_{d4}^c \end{bmatrix}. \quad (4.53)$$

From (4.16), the final switching sequence is calculated adding the integer part of the reference to the vectors of the displaced switching sequence:

$$\begin{aligned} \mathbf{v}_{s1} &= \mathbf{v}_i + \mathbf{v}_{d1} \\ \mathbf{v}_{s2} &= \mathbf{v}_i + \mathbf{v}_{d2} \\ \mathbf{v}_{s3} &= \mathbf{v}_i + \mathbf{v}_{d3} \\ \mathbf{v}_{s4} &= \mathbf{v}_i + \mathbf{v}_{d4}. \end{aligned} \quad (4.54)$$

Finally, the time corresponding to each switching vector is calculated directly from the components of $\hat{\mathbf{v}}_f$ by means of (4.33) as

$$\begin{aligned} t_1 &= 1 - \hat{v}_f^a \\ t_2 &= \hat{v}_f^a - \hat{v}_f^b \\ t_3 &= \hat{v}_f^b - \hat{v}_f^c \\ t_4 &= \hat{v}_f^c. \end{aligned} \quad (4.55)$$

If the modulation problem is solved in all the six possible cases in Table 4.2 then the results shown in Table 4.3 are obtained. This table proves that the application of the multiphase algorithm to a three-phase system is very simple. After normalizing the reference vector, it only requires three comparison operations and a few simple additions to evaluate the switching vectors and their corresponding switching times.

4.5.2 Example

Let us consider a three-phase drive where the voltage reference for each phase is purely sinusoidal:

$$\mathbf{V}_r = A \begin{bmatrix} \sin(\omega t) \\ \sin(\omega t + 2\pi/3) \\ \sin(\omega t - 2\pi/3) \end{bmatrix}. \quad (4.56)$$

If a voltage amplitude $A = 285$ V and an angular frequency $\omega = 2\pi \cdot 50$ rad/s are considered, the instantaneous reference when $t = 9$ ms is

$$\mathbf{V}_r = [88, -279, 191]^T \text{ V}. \quad (4.57)$$

From (4.48), if the voltage step of the converter is $V_{dc} = 150$ V then the normalized voltage reference is

$$\mathbf{v}_r = \frac{\mathbf{V}_r}{V_{dc}} = [0.59, -1.86, 1.27]^T. \quad (4.58)$$

TABLE 4.3
VECTOR SEQUENCE AND SWITCHING TIMES OF THE 3D SVPWM ALGORITHM.

| C_{ab} | C_{bc} | C_{ca} | Vector sequence \mathbf{v}_{sj} | Switching times t_j |
|----------|----------|----------|--|-----------------------|
| 0 | 0 | 1 | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | | | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1]^T$ | $t_2 = v_f^c - v_f^b$ |
| | | | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 1]^T$ | $t_3 = v_f^b - v_f^a$ |
| | | | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1]^T$ | $t_4 = v_f^a$ |
| 0 | 1 | 0 | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | | | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0]^T$ | $t_2 = v_f^b - v_f^a$ |
| | | | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 1, 0]^T$ | $t_3 = v_f^a - v_f^c$ |
| | | | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1]^T$ | $t_4 = v_f^c$ |
| 0 | 1 | 1 | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | | | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0]^T$ | $t_2 = v_f^b - v_f^c$ |
| | | | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 1]^T$ | $t_3 = v_f^c - v_f^a$ |
| | | | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1]^T$ | $t_4 = v_f^a$ |
| 1 | 0 | 0 | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | | | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0]^T$ | $t_2 = v_f^a - v_f^c$ |
| | | | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 1]^T$ | $t_3 = v_f^c - v_f^b$ |
| | | | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1]^T$ | $t_4 = v_f^b$ |
| 1 | 1 | 0 | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | | | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0]^T$ | $t_2 = v_f^a - v_f^b$ |
| | | | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 1, 0]^T$ | $t_3 = v_f^b - v_f^c$ |
| | | | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1]^T$ | $t_4 = v_f^c$ |
| 1 | 0 | 1 | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | | | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1]^T$ | $t_2 = v_f^c - v_f^a$ |
| | | | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 1]^T$ | $t_3 = v_f^a - v_f^b$ |
| | | | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1]^T$ | $t_4 = v_f^b$ |

By means of (4.49) and (4.50), this vector is decomposed into an integer and a fractional part:

$$\mathbf{v}_i = \text{integ}(\mathbf{v}_r) = [0, -2, 1]^T \quad (4.59)$$

$$\mathbf{v}_f = \mathbf{v}_r - \mathbf{v}_i = [0.59, 0.14, 0.27]^T. \quad (4.60)$$

If the conditions in (4.51) are calculated the results are

$$\begin{aligned} C_{ab} &= [v_f^a \geq v_f^b] = 1 \\ C_{bc} &= [v_f^b \geq v_f^c] = 0 \\ C_{ca} &= [v_f^c \geq v_f^a] = 0. \end{aligned} \quad (4.61)$$

The case *100* of Table 4.3 provides the following switching sequence:

$$\begin{aligned} \mathbf{v}_{s1} &= \mathbf{v}_i + [0, 0, 0]^T = [0, -2, 1]^T \\ \mathbf{v}_{s2} &= \mathbf{v}_i + [1, 0, 0]^T = [1, -2, 1]^T \\ \mathbf{v}_{s3} &= \mathbf{v}_i + [1, 0, 1]^T = [1, -2, 2]^T \\ \mathbf{v}_{s4} &= \mathbf{v}_i + [1, 1, 1]^T = [1, -1, 2]^T. \end{aligned} \quad (4.62)$$

together with the corresponding switching times:

$$\begin{aligned} t_1 &= 1 - v_f^a = 0.41 \\ t_2 &= v_f^a - v_f^c = 0.32 \\ t_3 &= v_f^c - v_f^b = 0.13 \\ t_4 &= v_f^b = 0.14. \end{aligned} \quad (4.63)$$

It is important to remark that consecutive vectors of the sequence are adjacent. Therefore, the number of switchings is minimized.

4.5.3 Comparison With Existing 3D Algorithm

Although the nomenclature is different¹, Table 4.3 is similar to table that details the state's sequences and the switching times in [81]. In fact, the six cases corresponding to the six tetrahedrons in each subcube of the paper are equivalent to the six cases derived from the three comparison operations of this work. In particular, the cases *1.1*, *1.2*, *1.3*, *2.1*, *2.2* and *2.3* are respectively equivalent to the cases *100*, *101*, *001*, *011*, *010* and *110* in Table 4.3.

Even though the comparison operations in both algorithms are not identical, the differences are in the boundaries of the tetrahedrons where two or more components of the fractional part of the reference are equal. In such cases, both algorithms can provide different switching sequences, but the switching vectors that are different in those sequences have a zero switching time so they are not executed and the output trigger signals are identical in both algorithms.

¹The terms $[S_a^j, S_b^j, S_c^j]$, $[a, b, c]$, $[u_a, u_b, u_c]$ and d_j in [81] are respectively the same as the terms $[v_{s_j}^a, v_{s_j}^b, v_{s_j}^c]$, $[v_i^a, v_i^b, v_i^c]$, $[v_r^a, v_r^b, v_r^c]$ and t_j here.

As a consequence, the multiphase algorithm particularized to three-phase systems obtained in this section is nearly the same as the 3D SVPWM generalized algorithm presented in [81]. Figure 4.12 shows the modulation output of both modulation techniques when a balanced three-phase reference with a normalized amplitude of 1.9 is considered. A low switching frequency (twenty times the fundamental) has been selected to make the comparison easier. In both cases the output is identical, even in the sixth sampling period, when $\mathbf{v}_r = [1.9, -0.95, -0.95]^T$, in which the fractional parts of the phases b and c are equal. In this particular case the new multilevel three-phase algorithm provides the following switching vectors and switching times (case 110):

$$\begin{aligned} \mathbf{v}_{s1} &= [1, -1, -1]^T & \rightarrow & t_1 = 0.10 \\ \mathbf{v}_{s2} &= [2, -1, -1]^T & \rightarrow & t_2 = 0.85 \\ \mathbf{v}_{s3} &= [2, 0, -1]^T & \rightarrow & t_3 = 0.00 \\ \mathbf{v}_{s4} &= [2, 0, 0]^T & \rightarrow & t_4 = 0.05 \end{aligned} \tag{4.64}$$

whereas the 3D SVPWM generalized algorithm provides the following results (case 1.1):

$$\begin{aligned} \mathbf{v}_{s1} &= [1, -1, -1]^T & \rightarrow & t_1 = 0.10 \\ \mathbf{v}_{s2} &= [2, -1, -1]^T & \rightarrow & t_2 = 0.85 \\ \mathbf{v}_{s3} &= [2, -1, 0]^T & \rightarrow & t_3 = 0.00 \\ \mathbf{v}_{s4} &= [2, 0, 0]^T & \rightarrow & t_4 = 0.05. \end{aligned} \tag{4.65}$$

Two switching sequences differ in the third output vector. Nevertheless, those vectors have a zero switching time so finally none of them must be generated by the inverter and the trigger signals are identical in both cases.

Since both algorithms are nearly the same, all considerations made in chapter §3 and in other works about the 3D SVPWM generalized algorithm also apply to the new algorithm [81]. Hence, it can be considered as an extension of 2D algorithms [72]. It is also equivalent to the 3D direct generalized algorithm proposed in [169]. It can be used as a modulation algorithm in all applications needing a 3D control where the conventional 2D SVPWM cannot be used, such as in active filters with four-wire and single-phase distorting loads which generate large neutral currents.

Since the algorithm is a particularization of the multilevel multiphase SVPWM developed in above section, it inherits some features such as it minimizes the number of switchings, it does not need trigonometric functions, and it has low computational cost. Hence, the algorithm is well suited for real-time implementation in low cost devices.

4.5.4 Experimental Results

The SVPWM algorithm was described for a three-level inverter by using VHDL and it was implemented in a Digilent S3 board. Table 4.4 shows a summary of the resources used by the implementation. It is important to remark that the BRAMs and the multipliers available in the FPGA were not used because the algorithm does not need data storage or multiplication operations.

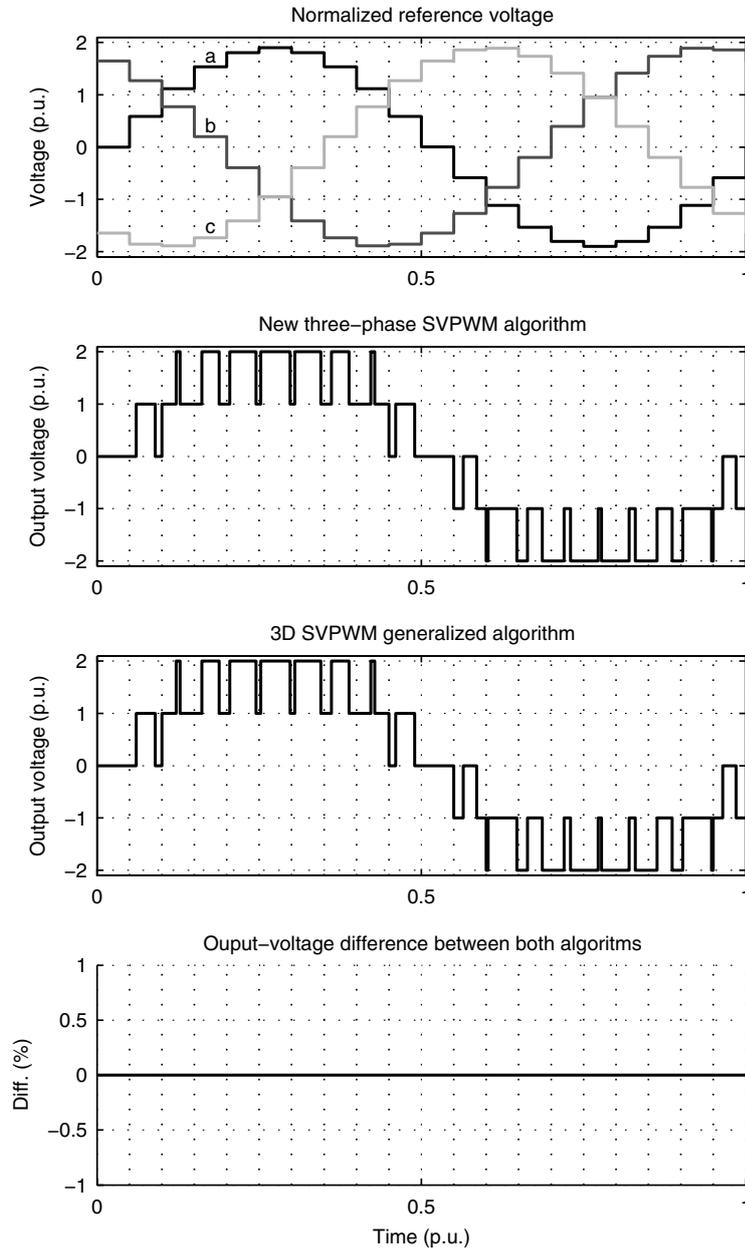


Figure 4.12: 3D algorithms comparison.

TABLE 4.4
RESOURCES SUMMARY.

| | | |
|---|--------------------|------|
| Target Device | : xc3s200 | |
| Number of Slice Flip Flops: | 2,179 out of 3,840 | 56% |
| Number of 4 input LUTs: | 2,282 out of 3,840 | 59% |
| Number of occupied Slices: | 1,669 out of 1,920 | 86% |
| Total Number 4 input LUTs: | 2,450 out of 3,840 | 63% |
| Number of bonded IOBs: | 63 out of 173 | 36% |
| IOB Flip Flops: | 39 | |
| Number of Block RAMs: | 0 out of 12 | 0% |
| Number of MULT18X18s: | 0 out of 12 | 0% |
| Number of GCLKs: | 8 out of 8 | 100% |
| Number of Startups: | 1 out of 1 | 100% |
| Total equivalent gate count for design: | 32,174 | |

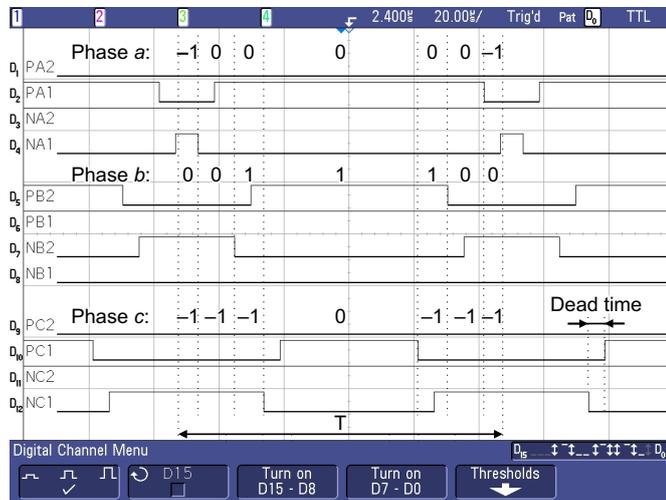
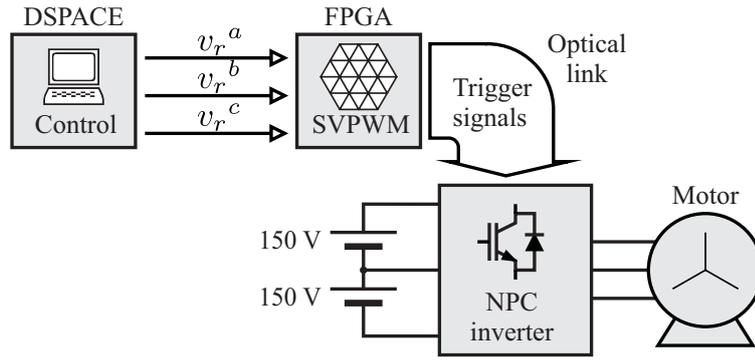


Figure 4.13: Trigger signals.

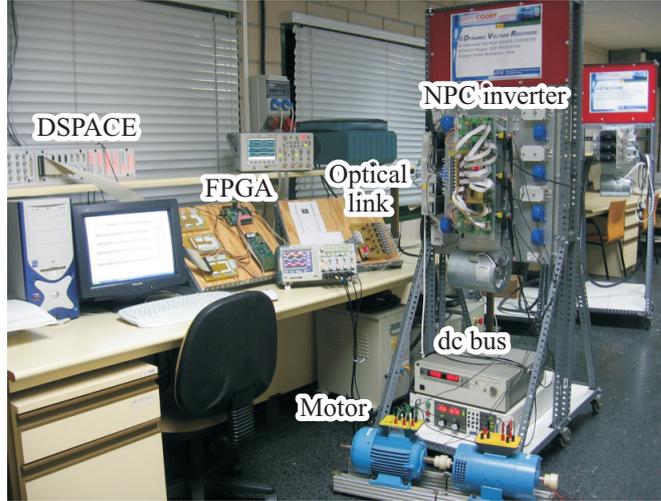
Figure 4.13 shows the FPGA output waveforms that correspond to the trigger signals in the case $\mathbf{v}_r = [-0.16, 0.52, -0.77]^T$. The needed dead time in complementary trigger signals was implemented by delaying the rising edges.

The new algorithm was tested with the three-level neutral-point clamped (NPC) voltage-source inverter (VSI) shown in Figure 3.2 driving a star connected induction motor with isolated neutral. A 220/380 V, 1.420 r/min, 1.35 kW rated motor was used. In accordance with the motor characteristics a dc bus of 300 V was selected. Figure 4.14 shows a diagram and a photograph of the experimental setup used in tests. It includes the power converter, the FPGA board and a personal computer with a DSPACE DS1103 PPC Controller Board.

A reference voltage with 50 Hz fundamental frequency and a 10 kHz output switching frequency were considered. Figure 4.15 shows the inverter output voltage waveform and the low-order voltage harmonics, in the cases of sinusoidal reference with and without third harmonic injection. Channel one shows the phase output voltage V_s^a measured with respect to the midpoint of the dc bus (see Figure 3.2). Channel two is the same signal after been filtered. Figure 4.15a shows the case of purely sinusoidal reference with



(a) Diagram



(b) Photograph

Figure 4.14: Experimental test setup.

a normalized amplitude of 0.9 p.u. In this case the low-order harmonics are low and the THD is 4.8%. Figure 4.15b shows the case where a third harmonic of amplitude 0.4 p.u. was added to the reference. In this case the high THD obtained corresponds mainly to the injected harmonic.

4.6 Application to Three-Phase Four-Leg Converters

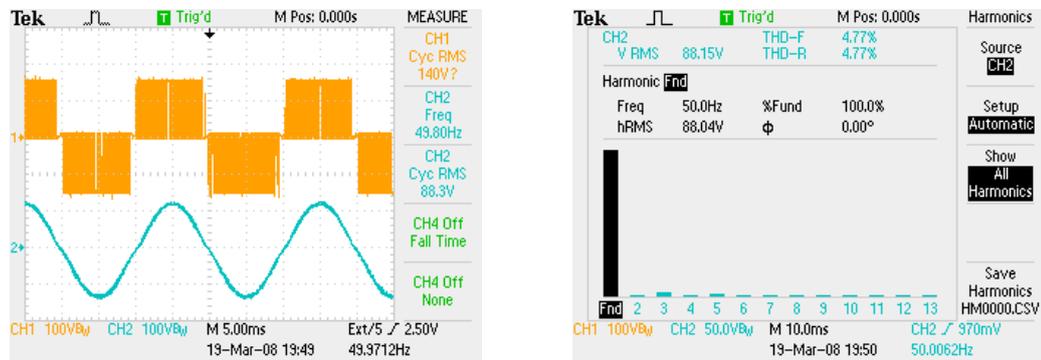
4.6.1 4D SVPWM Algorithm

Prior art of SVPWM for four-leg converters is formulated in a 3D space [89, 170–172]. Nevertheless, the switching states of multilevel four-leg converters are four-dimensional (4D) integer vectors:

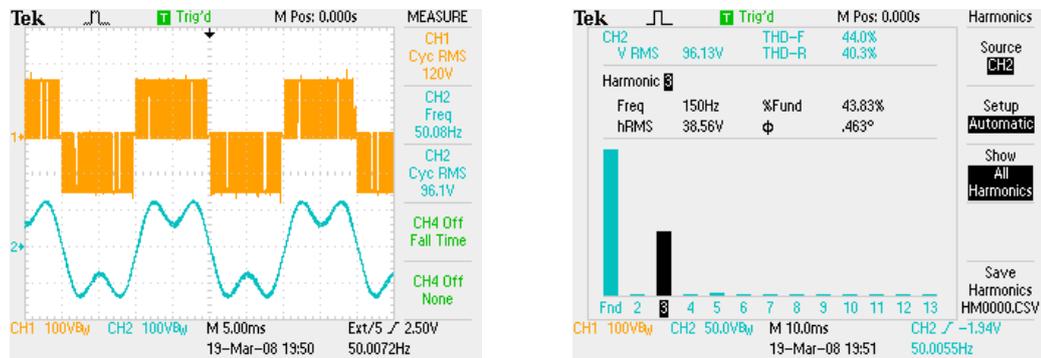
$$\mathbf{v}_{sj} = [v_{sj}^a, v_{sj}^b, v_{sj}^c, v_{sj}^n]^T \in \mathbb{Z}^4. \quad (4.66)$$

Therefore, the modulation problem can be naturally formulated in a 4D vector space.

The functional diagram of a multilevel four-leg converter, shown in Figure 4.16, is identical to the functional diagram of a multilevel four-phase converter where the neutral



(a) Without harmonic injection



(b) With third harmonic injection

Figure 4.15: Inverter output voltage V_s^a . Ch1: switched voltage; Ch2: filtered voltage.

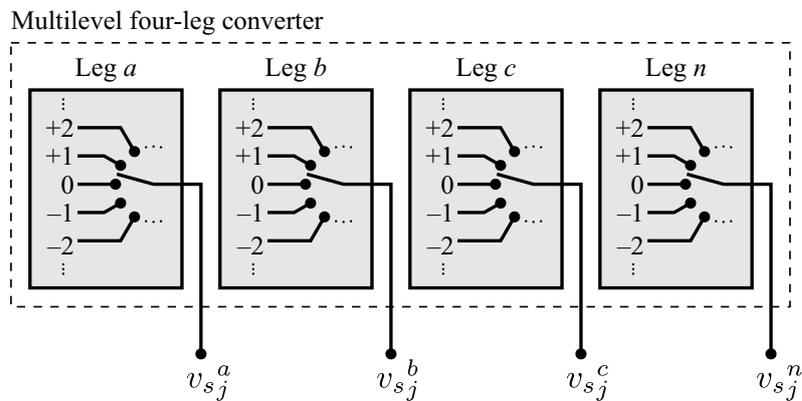


Figure 4.16: Functional diagram of multilevel four-leg converter.

leg corresponds to the fourth phase. Hence, the modulation algorithm for four-leg converters can be obtained from the multiphase SVPWM algorithm by making $P = 4$. The voltage reference includes the reference for each phase together with the reference for the neutral leg. Taking into account (4.7), the normalized voltage reference \mathbf{v}_r is the 4D real vector:

$$\mathbf{v}_r = \frac{\mathbf{V}_r}{V_{dc}} = [v_r^a, v_r^b, v_r^c, v_r^n]^T \in \mathbb{R}^4. \quad (4.67)$$

In accordance with (4.15), the integer and fractional parts of the reference vector are

$$\mathbf{v}_i = \text{integ}(\mathbf{v}_r) = [v_i^a, v_i^b, v_i^c, v_i^n]^T \in \mathbb{Z}^4 \quad (4.68)$$

$$\mathbf{v}_f = \mathbf{v}_r - \mathbf{v}_i = [v_f^a, v_f^b, v_f^c, v_f^n]^T \in \mathbb{R}^4. \quad (4.69)$$

The permutation matrix \mathbf{P} defined in (4.26) that sorts the components of the vector \mathbf{v}_f can be determined by testing the six following logical conditions:

$$\begin{aligned} C_{ab} &= [v_f^a \geq v_f^b] & C_{bc} &= [v_f^b \geq v_f^c] \\ C_{ac} &= [v_f^a \geq v_f^c] & C_{bn} &= [v_f^b \geq v_f^n] \\ C_{an} &= [v_f^a \geq v_f^n] & C_{cn} &= [v_f^c \geq v_f^n]. \end{aligned} \quad (4.70)$$

Let us define the following binary word joining the bits resulting from those six comparison operations

$$C_w = [C_{cn} C_{bn} C_{bc} C_{an} C_{ac} C_{ab}]. \quad (4.71)$$

Table 4.5 shows permutation matrix \mathbf{P} corresponding to each C_w . Although, with six bits there are sixty four possible combinations, just the twenty four cases listed in the table are coherent. The coefficient matrix \mathbf{D} is calculated from the upper triangular matrix $\hat{\mathbf{D}}$ in (4.30) by means of (4.32) as

$$\mathbf{D} = \mathbf{P}^T \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.72)$$

Taking into account (4.24), the sequence of displaced vectors $\{\mathbf{v}_{d1}, \mathbf{v}_{d2}, \mathbf{v}_{d3}, \mathbf{v}_{d4}, \mathbf{v}_{d5}\}$ must be extracted from the matrix:

$$\mathbf{D} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ v_{d1}^a & v_{d2}^a & v_{d3}^a & v_{d4}^a & v_{d5}^a \\ v_{d1}^b & v_{d2}^b & v_{d3}^b & v_{d4}^b & v_{d5}^b \\ v_{d1}^c & v_{d2}^c & v_{d3}^c & v_{d4}^c & v_{d5}^c \\ v_{d1}^n & v_{d2}^n & v_{d3}^n & v_{d4}^n & v_{d5}^n \end{bmatrix}. \quad (4.73)$$

TABLE 4.5
PERMUTATION MATRIX IN THE 4D SVPWM ALGORITHM.

| C_w | Vector \hat{v}_f | Matrix \mathbf{P} | C_w | vector \hat{v}_f | Matrix \mathbf{P} |
|----------|--|---|----------|--|---|
| [000000] | $\hat{v}_f^a = v_f^n$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^b$ $\hat{v}_f^n = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ | [100000] | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^n$ $\hat{v}_f^c = v_f^b$ $\hat{v}_f^n = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ |
| [000001] | $\hat{v}_f^a = v_f^n$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^a$ $\hat{v}_f^n = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$ | [100001] | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^n$ $\hat{v}_f^c = v_f^a$ $\hat{v}_f^n = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$ |
| [000011] | $\hat{v}_f^a = v_f^n$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^c$ $\hat{v}_f^n = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$ | [100101] | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^n$ $\hat{v}_f^n = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$ |
| [000111] | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^n$ $\hat{v}_f^c = v_f^c$ $\hat{v}_f^n = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$ | [100111] | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^n$ $\hat{v}_f^n = v_f^b$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$ |
| [001000] | $\hat{v}_f^a = v_f^n$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^c$ $\hat{v}_f^n = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ | [110000] | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^n$ $\hat{v}_f^n = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ |
| [001010] | $\hat{v}_f^a = v_f^n$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^a$ $\hat{v}_f^n = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$ | [110100] | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^a$ $\hat{v}_f^n = v_f^n$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ |

Continued on next page

TABLE 4.5
PERMUTATION MATRIX IN THE 4D SVPWM ALGORITHM. (CONTINUED)

| C_w | Vector \hat{v}_f | Matrix \mathbf{P} | C_w | vector \hat{v}_f | Matrix \mathbf{P} |
|----------|--|---|----------|--|---|
| [001011] | $\hat{v}_f^a = v_f^n$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^b$ $\hat{v}_f^n = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$ | [110101] | $\hat{v}_f^a = v_f^c$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^b$ $\hat{v}_f^n = v_f^n$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ |
| [001111] | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^n$ $\hat{v}_f^c = v_f^b$ $\hat{v}_f^n = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$ | [110111] | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^b$ $\hat{v}_f^n = v_f^n$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ |
| [011000] | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^n$ $\hat{v}_f^c = v_f^c$ $\hat{v}_f^n = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ | [111000] | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^n$ $\hat{v}_f^n = v_f^a$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ |
| [011010] | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^n$ $\hat{v}_f^c = v_f^a$ $\hat{v}_f^n = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$ | [111100] | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^c$ $\hat{v}_f^c = v_f^a$ $\hat{v}_f^n = v_f^n$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ |
| [011110] | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^n$ $\hat{v}_f^n = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$ | [111110] | $\hat{v}_f^a = v_f^b$ $\hat{v}_f^b = v_f^a$ $\hat{v}_f^c = v_f^c$ $\hat{v}_f^n = v_f^n$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ |
| [011111] | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^n$ $\hat{v}_f^n = v_f^c$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$ | [111111] | $\hat{v}_f^a = v_f^a$ $\hat{v}_f^b = v_f^b$ $\hat{v}_f^c = v_f^c$ $\hat{v}_f^n = v_f^n$ | $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ |

From (4.16), the vectors of the final switching sequence are calculated by adding the integer part of the reference to the displaced vectors:

$$\begin{aligned}
 \mathbf{v}_{s1} &= \mathbf{v}_i + \mathbf{v}_{d1} \\
 \mathbf{v}_{s2} &= \mathbf{v}_i + \mathbf{v}_{d2} \\
 \mathbf{v}_{s3} &= \mathbf{v}_i + \mathbf{v}_{d3} \\
 \mathbf{v}_{s4} &= \mathbf{v}_i + \mathbf{v}_{d4} \\
 \mathbf{v}_{s5} &= \mathbf{v}_i + \mathbf{v}_{d5}.
 \end{aligned} \tag{4.74}$$

The time corresponding to each switching vector is calculated directly from the components of $\hat{\mathbf{v}}_f$ by means of (4.33) as

$$\begin{aligned}
 t_1 &= 1 - \hat{v}_f^a \\
 t_2 &= \hat{v}_f^a - \hat{v}_f^b \\
 t_3 &= \hat{v}_f^b - \hat{v}_f^c \\
 t_4 &= \hat{v}_f^c - \hat{v}_f^n \\
 t_5 &= \hat{v}_f^n.
 \end{aligned} \tag{4.75}$$

If the modulation problem is solved in all the possible cases, listed in Table 4.5, then the results shown in Table 4.6 are obtained. This table proves that the application of the multiphase algorithm to four-leg converters is very simple. After normalizing the reference vector, it only requires six comparison operations and a few additions in order to evaluate the switching vectors and their corresponding switching times.

4.6.2 Example

Let us consider a three-phase four-leg system with an unbalanced voltage reference with a zero sequence:

$$\mathbf{V}_r = \begin{bmatrix} 42 \sin(\omega t) \\ 57 \sin(\omega t + 2\pi/3) \\ 24 \sin(\omega t - 2\pi/3) \\ 36 \sin(3\omega t + \pi) \end{bmatrix} \text{ V}. \tag{4.76}$$

If $\omega = 2\pi \cdot 50$ rad/s is considered then the instantaneous reference when $t = 5.4$ ms is

$$\mathbf{V}_r = [41.7, -34.5, -9.3, 33.5]^T \text{ V}. \tag{4.77}$$

From (4.67), if the voltage step of the converter is $V_{dc} = 30$ V then normalized voltage reference is

$$\mathbf{v}_r = \frac{\mathbf{V}_r}{V_{dc}} = [1.39, -1.15, -0.31, 1.12]^T. \tag{4.78}$$

By means of (4.68) and (4.69), this vector is decomposed into an integer and a fractional part as

$$\mathbf{v}_i = \text{integ}(\mathbf{v}_r) = [1, -2, -1, 1]^T \tag{4.79}$$

$$\mathbf{v}_f = \mathbf{v}_r - \mathbf{v}_i = [0.39, 0.85, 0.69, 0.12]^T. \tag{4.80}$$

TABLE 4.6
VECTOR SEQUENCE AND SWITCHING TIMES OF THE 4D SVPWM ALGORITHM.

| C_w | Vector sequence \mathbf{v}_{sj} | Switching times t_j |
|----------|---|-----------------------|
| [000000] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^n$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 0, 1]^T$ | $t_2 = v_f^n - v_f^c$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 0, 1, 1]^T$ | $t_3 = v_f^c - v_f^b$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [0, 1, 1, 1]^T$ | $t_4 = v_f^b - v_f^a$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^a$ |
| [000001] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^n$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 0, 1]^T$ | $t_2 = v_f^n - v_f^c$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 0, 1, 1]^T$ | $t_3 = v_f^c - v_f^a$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 0, 1, 1]^T$ | $t_4 = v_f^a - v_f^b$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^b$ |
| [000011] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^n$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 0, 1]^T$ | $t_2 = v_f^n - v_f^a$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 0, 1]^T$ | $t_3 = v_f^a - v_f^c$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 0, 1, 1]^T$ | $t_4 = v_f^c - v_f^b$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^b$ |
| [000111] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0, 0]^T$ | $t_2 = v_f^a - v_f^n$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 0, 1]^T$ | $t_3 = v_f^n - v_f^c$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 0, 1, 1]^T$ | $t_4 = v_f^c - v_f^b$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^b$ |
| [001000] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^n$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 0, 1]^T$ | $t_2 = v_f^n - v_f^b$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 0, 1]^T$ | $t_3 = v_f^b - v_f^c$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [0, 1, 1, 1]^T$ | $t_4 = v_f^c - v_f^a$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^a$ |
| [001010] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^n$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 0, 1]^T$ | $t_2 = v_f^n - v_f^b$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 0, 1]^T$ | $t_3 = v_f^b - v_f^a$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 0, 1]^T$ | $t_4 = v_f^a - v_f^c$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^c$ |
| [001011] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^n$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 0, 1]^T$ | $t_2 = v_f^n - v_f^a$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 0, 1]^T$ | $t_3 = v_f^a - v_f^b$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 0, 1]^T$ | $t_4 = v_f^b - v_f^c$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^c$ |

Continued on next page

TABLE 4.6
VECTOR SEQUENCE AND SWITCHING TIMES OF THE 4D SVPWM ALGORITHM.
(CONTINUED)

| C_w | Vector sequence \mathbf{v}_{sj} | Switching times t_j |
|----------|---|-----------------------|
| [001111] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0, 0]^T$ | $t_2 = v_f^a - v_f^n$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 0, 1]^T$ | $t_3 = v_f^n - v_f^b$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 0, 1]^T$ | $t_4 = v_f^b - v_f^c$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^c$ |
| [011000] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0, 0]^T$ | $t_2 = v_f^b - v_f^n$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 0, 1]^T$ | $t_3 = v_f^n - v_f^c$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [0, 1, 1, 1]^T$ | $t_4 = v_f^c - v_f^a$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^a$ |
| [011010] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0, 0]^T$ | $t_2 = v_f^b - v_f^n$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 0, 1]^T$ | $t_3 = v_f^n - v_f^a$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 0, 1]^T$ | $t_4 = v_f^a - v_f^c$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^c$ |
| [011110] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0, 0]^T$ | $t_2 = v_f^b - v_f^a$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 1, 0, 0]^T$ | $t_3 = v_f^a - v_f^n$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 0, 1]^T$ | $t_4 = v_f^n - v_f^c$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^c$ |
| [011111] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0, 0]^T$ | $t_2 = v_f^a - v_f^b$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 1, 0, 0]^T$ | $t_3 = v_f^b - v_f^n$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 0, 1]^T$ | $t_4 = v_f^n - v_f^c$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^c$ |
| [100000] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1, 0]^T$ | $t_2 = v_f^c - v_f^n$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 0, 1, 1]^T$ | $t_3 = v_f^n - v_f^b$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [0, 1, 1, 1]^T$ | $t_4 = v_f^b - v_f^a$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^a$ |
| [100001] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1, 0]^T$ | $t_2 = v_f^c - v_f^n$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 0, 1, 1]^T$ | $t_3 = v_f^n - v_f^a$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 0, 1, 1]^T$ | $t_4 = v_f^a - v_f^b$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^b$ |

Continued on next page

TABLE 4.6
VECTOR SEQUENCE AND SWITCHING TIMES OF THE 4D SVPWM ALGORITHM.
(CONTINUED)

| C_w | Vector sequence \mathbf{v}_{sj} | Switching times t_j |
|----------|---|-----------------------|
| [100101] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1, 0]^T$ | $t_2 = v_f^c - v_f^a$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 1, 0]^T$ | $t_3 = v_f^a - v_f^n$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 0, 1, 1]^T$ | $t_4 = v_f^n - v_f^b$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^b$ |
| [100111] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0, 0]^T$ | $t_2 = v_f^a - v_f^c$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 1, 0]^T$ | $t_3 = v_f^c - v_f^n$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 0, 1, 1]^T$ | $t_4 = v_f^n - v_f^b$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^b$ |
| [110000] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1, 0]^T$ | $t_2 = v_f^c - v_f^b$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 1, 0]^T$ | $t_3 = v_f^b - v_f^n$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [0, 1, 1, 1]^T$ | $t_4 = v_f^n - v_f^a$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^a$ |
| [110100] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1, 0]^T$ | $t_2 = v_f^c - v_f^b$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 1, 0]^T$ | $t_3 = v_f^b - v_f^a$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1, 0]^T$ | $t_4 = v_f^a - v_f^n$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^n$ |
| [110101] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^c$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 0, 1, 0]^T$ | $t_2 = v_f^c - v_f^a$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 1, 0]^T$ | $t_3 = v_f^a - v_f^b$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1, 0]^T$ | $t_4 = v_f^b - v_f^n$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^n$ |
| [110111] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0, 0]^T$ | $t_2 = v_f^a - v_f^c$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 0, 1, 0]^T$ | $t_3 = v_f^c - v_f^b$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1, 0]^T$ | $t_4 = v_f^b - v_f^n$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^n$ |
| [111000] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0, 0]^T$ | $t_2 = v_f^b - v_f^c$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 1, 0]^T$ | $t_3 = v_f^c - v_f^n$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [0, 1, 1, 1]^T$ | $t_4 = v_f^n - v_f^a$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^a$ |

Continued on next page

TABLE 4.6
VECTOR SEQUENCE AND SWITCHING TIMES OF THE 4D SVPWM ALGORITHM.
(CONTINUED)

| C_w | Vector sequence \mathbf{v}_{sj} | Switching times t_j |
|----------|---|-----------------------|
| [111100] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0, 0]^T$ | $t_2 = v_f^b - v_f^c$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [0, 1, 1, 0]^T$ | $t_3 = v_f^c - v_f^a$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1, 0]^T$ | $t_4 = v_f^a - v_f^n$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^n$ |
| [111110] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^b$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [0, 1, 0, 0]^T$ | $t_2 = v_f^b - v_f^a$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 1, 0, 0]^T$ | $t_3 = v_f^a - v_f^c$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1, 0]^T$ | $t_4 = v_f^c - v_f^n$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^n$ |
| [111111] | $\mathbf{v}_{s1} = \mathbf{v}_i + [0, 0, 0, 0]^T$ | $t_1 = 1 - v_f^a$ |
| | $\mathbf{v}_{s2} = \mathbf{v}_i + [1, 0, 0, 0]^T$ | $t_2 = v_f^a - v_f^b$ |
| | $\mathbf{v}_{s3} = \mathbf{v}_i + [1, 1, 0, 0]^T$ | $t_3 = v_f^b - v_f^c$ |
| | $\mathbf{v}_{s4} = \mathbf{v}_i + [1, 1, 1, 0]^T$ | $t_4 = v_f^c - v_f^n$ |
| | $\mathbf{v}_{s5} = \mathbf{v}_i + [1, 1, 1, 1]^T$ | $t_5 = v_f^n$ |

If the six conditions in (4.70) are tested the following six-bit word is obtained:

$$C_w = [111100]. \quad (4.81)$$

In this case, Table 4.6 provides the following switching sequence:

$$\begin{aligned}
\mathbf{v}_{s1} &= \mathbf{v}_i + [0, 0, 0, 0]^T = [1, -2, -1, 1]^T \\
\mathbf{v}_{s2} &= \mathbf{v}_i + [0, 1, 0, 0]^T = [1, -1, -1, 1]^T \\
\mathbf{v}_{s3} &= \mathbf{v}_i + [0, 1, 1, 0]^T = [1, -1, 0, 1]^T \\
\mathbf{v}_{s4} &= \mathbf{v}_i + [1, 1, 1, 0]^T = [2, -1, 0, 1]^T \\
\mathbf{v}_{s5} &= \mathbf{v}_i + [1, 1, 1, 1]^T = [2, -1, 0, 2]^T
\end{aligned} \quad (4.82)$$

together with the following switching times:

$$\begin{aligned}
t_1 &= 1 - v_f^b = 0.15 \\
t_2 &= v_f^b - v_f^c = 0.16 \\
t_3 &= v_f^c - v_f^a = 0.30 \\
t_4 &= v_f^a - v_f^n = 0.27 \\
t_5 &= v_f^n = 0.12.
\end{aligned} \quad (4.83)$$

It is important to remark that consecutive vectors of the sequence are adjacent. Therefore, the number of switchings is minimized.

TABLE 4.7
RESOURCES SUMMARY.

| | | | |
|---|-----------|--------------|------|
| Target Device | : xc3s200 | | |
| Number of Slice Flip Flops: | 2,278 | out of 3,840 | 59% |
| Number of 4 input LUTs: | 2,682 | out of 3,840 | 69% |
| Number of occupied Slices: | 1,850 | out of 1,920 | 96% |
| Total Number 4 input LUTs: | 2,597 | out of 3,840 | 67% |
| Number of bonded IOBs: | 91 | out of 173 | 36% |
| IOB Flip Flops: | 66 | | |
| Number of Block RAMs: | 0 | out of 12 | 0% |
| Number of MULT18X18s: | 0 | out of 12 | 0% |
| Number of GCLKs: | 8 | out of 8 | 100% |
| Number of Startups: | 1 | out of 1 | 100% |
| Total equivalent gate count for design: | 35,861 | | |

4.6.3 Experimental Results

The algorithm was tested by simulation and in laboratory. In both cases, the considered conditions are an unbalanced voltage reference, where $v_r^a = 1.4 \sin(\omega t)$, $v_r^b = 1.9 \sin(\omega t + 2\pi/3)$ and $v_r^c = 0.8 \sin(\omega t - 2\pi/3)$ with a third-harmonic zero sequence, $v_r^n = 1.2 \sin(3\omega t + \pi)$. The fundamental frequency is 50 Hz and the switching frequency is 10 kHz. Figure 4.17 shows the simulation results obtained with Simulink. The four traces in the first plot are the reference voltages for each leg. The subsequent plots are the output voltage (in black) and the filtered output voltage (in gray) obtained in each leg. The filtered output voltages clearly follow the reference signals.

The SVPWM algorithm was described for a five-level inverter by using VHDL and it was implemented by using a Digilent S3 board. Table 4.7 shows a summary of the resources used by the implementation. It is important to remark that the BRAMs and the multipliers available in the FPGA were not used because the algorithm does not need data storage or multiplication operations. Figure 4.18 shows the FPGA output waveforms, which correspond to the trigger signals, in the same case shown in the example where $\mathbf{v}_r = [1.39, -1.10, -0.33, 1.15]^T$. The experimental results are in accordance with the theoretical results calculated in previous section.

The new algorithm was tested with a five-level cascaded full-bridge VSI shown in Figure 4.19. The voltage of all dc sources is $V_{dc} = 30$ V. Figure 4.20 shows a diagram and a photograph of the experimental setup used in tests. It includes the power converter, the FPGA board and a personal computer with a DSPACE DS1103 PPC Controller Board. Figure 4.21 shows the measured output voltage of the inverter in the simulated case. The output voltages V_s^a , V_s^b , V_s^c and V_s^n have been measured with respect to the neutral of the inverter in the points displayed in Figure 4.19. The experimental results agree with the simulation results shown in Figure 4.17.

Since the low-order harmonics of the phase voltage shown in Figure 4.22 are low, the output voltage has low THD, which is lower than 5.0%. The spectrum of the same signal, shown in Figure 4.23, reveals that the main high-order harmonics are integer multiples of the switching frequency.

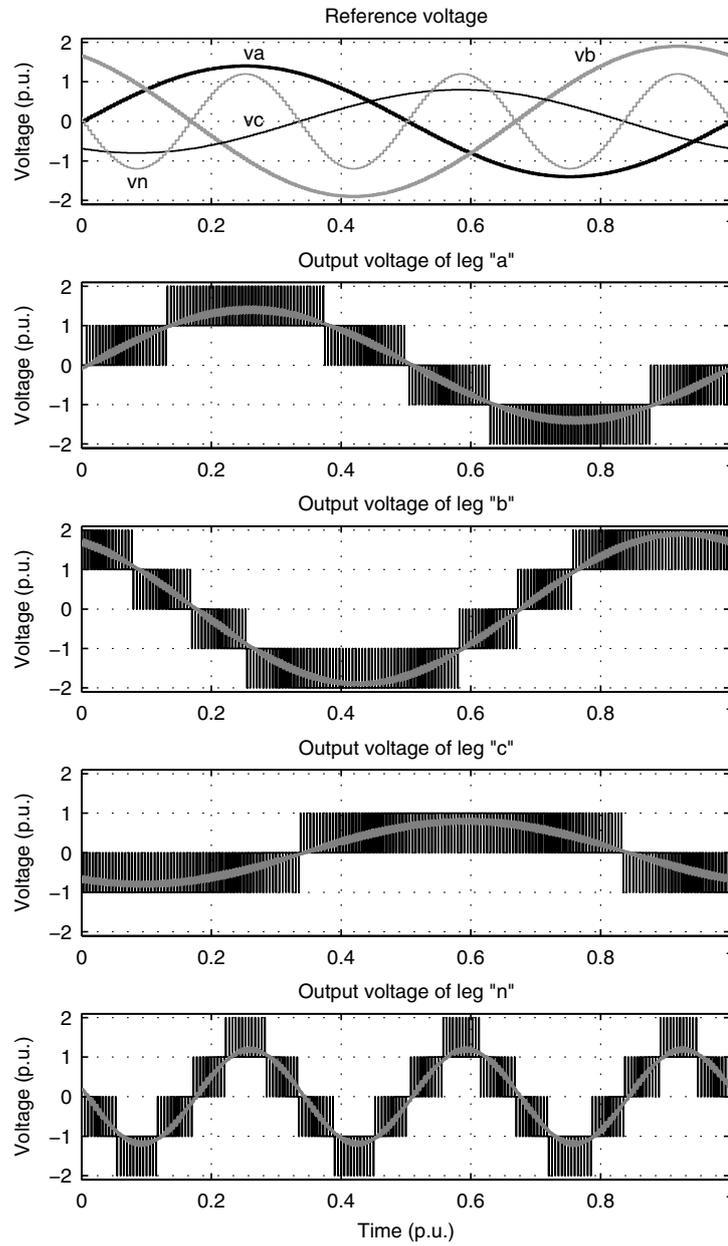


Figure 4.17: Simulation results.

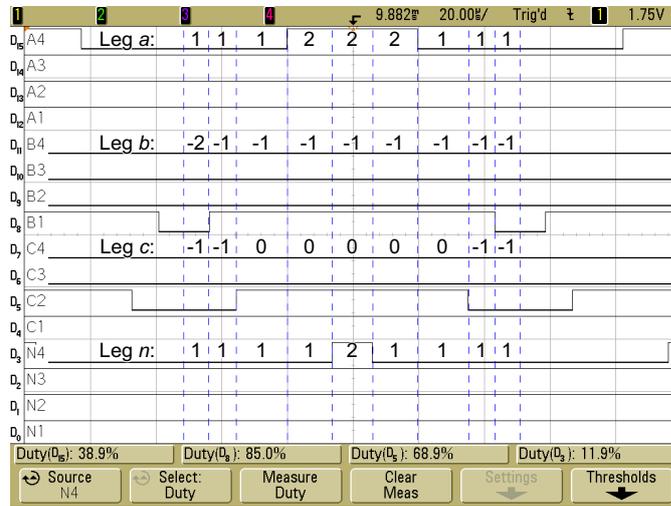


Figure 4.18: Trigger signals.

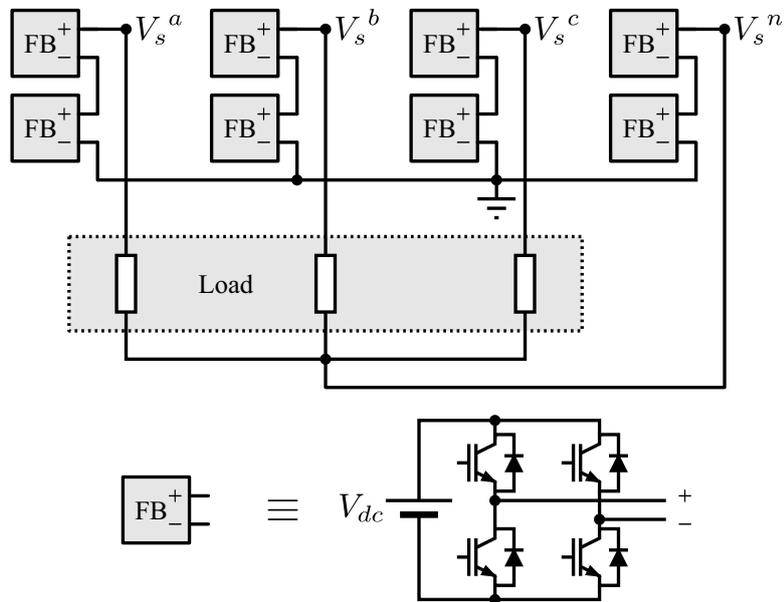
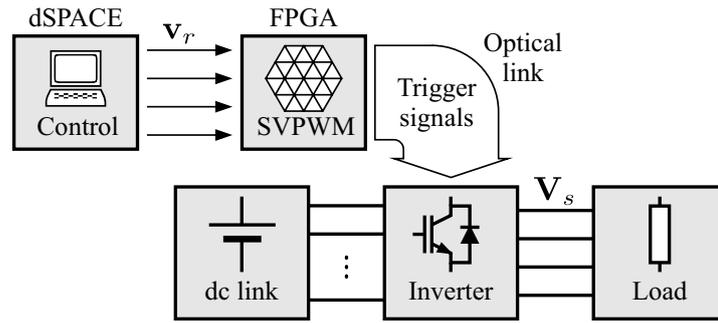
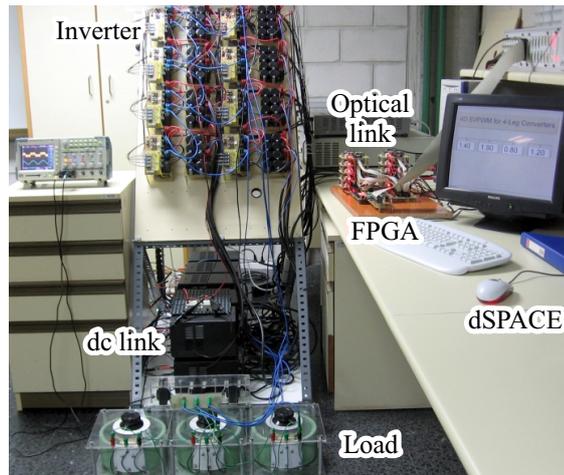


Figure 4.19: Four-leg five-level cascaded full-bridge inverter.



(a) Diagram



(b) Photograph

Figure 4.20: Experimental test setup.

4.7 Conclusion

The chapter develops a novel multilevel multiphase SVPWM algorithm. The algorithm is based on a displacement plus a two-level multiphase SVPWM modulator. It is valid for any number of phases or levels and it can be used with the standard multilevel topologies, such as diode clamped, flying capacitor or cascaded full-bridge converters. The new modulation technique handles all switching states of the inverter and it provides a sorted switching vector sequence that minimizes the number of switchings. In addition, the proposed SVPWM algorithm proves suitable for real-time implementation due to its low computational complexity. The new algorithm was validated by its hardware implementation in a low-cost FPGA and successfully tested by using a laboratory prototype made with a five-level five-phase cascade full-bridge inverter.

In addition, the algorithm was particularized for three-phase converters with three and four legs. For three-leg converters, the resulting algorithm is nearly the same as the existing 3D SVPWM generalized algorithm for multilevel converters presented by Prats et al.. The application of the new multiphase modulation technique to four-leg converters provides a new multilevel 4D SVPWM algorithm.

The multilevel multiphase SVPWM algorithm cannot handle the joint-phase redundancy present in converters with no neutral wire. In such case the SVPWM algorithm

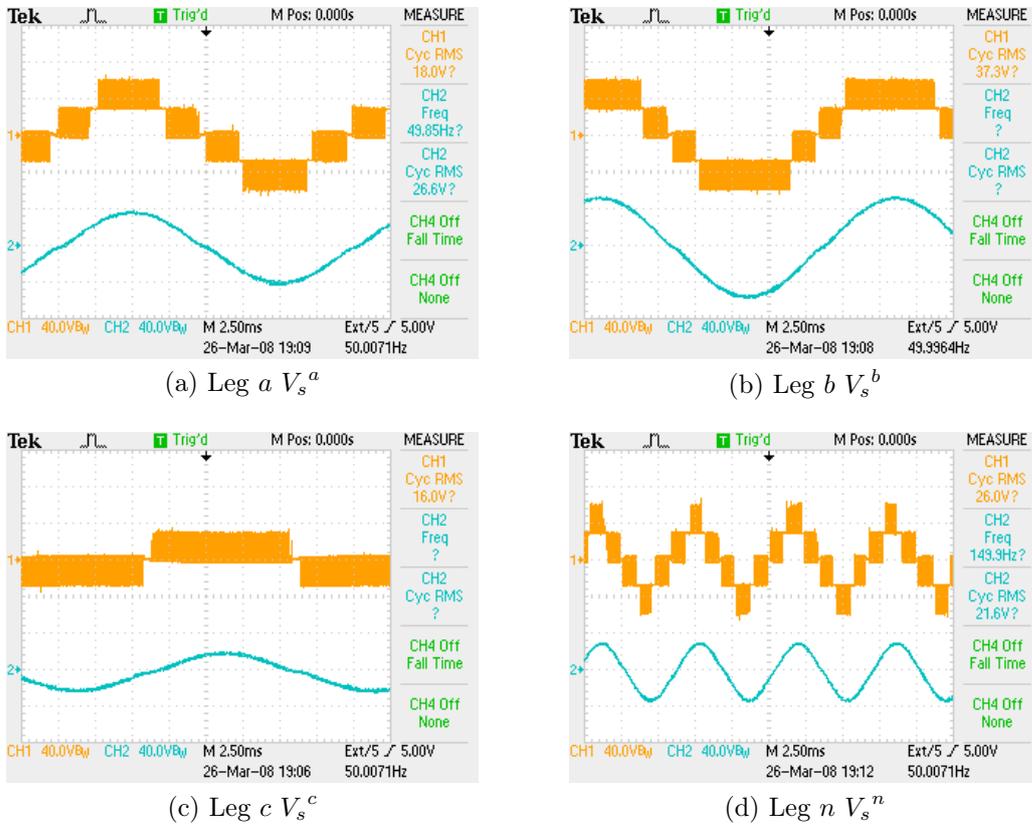


Figure 4.21: Inverter output voltage. Ch1: switched voltage; Ch2: filtered voltage.

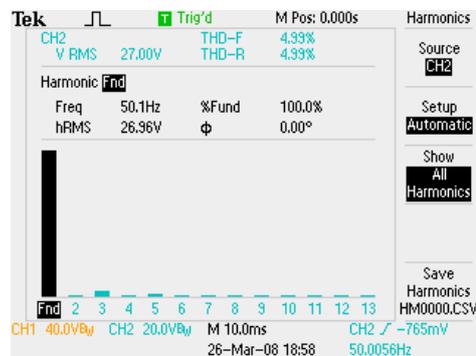


Figure 4.22: Low order harmonics of phase voltage.

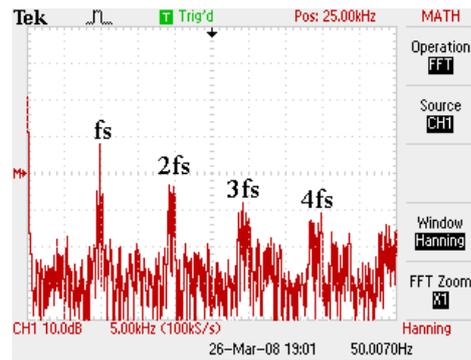


Figure 4.23: Spectrum of phase voltage.

developed in the following chapter is more appropriate.

Contributions of this chapter have been published in a journal [164] and presented at an international conference [165, 166]. A demo of the new modulation technique will be included in the *SimPowerSystems* toolbox of *Simulink*.

Chapter 5

Multilevel Multiphase SVPWM Algorithm With Switching State Redundancy

***Abstract** — The chapter presents a new multilevel multiphase SVPWM algorithm that takes advantage of the joint-phase redundancy present in voltage source converters without neutral wire. This algorithm is formulated in a multidimensional space, it is based on a two-level multiphase SVPWM algorithm without redundancy and it includes a method to carry out the redundant switching state selection. The new SVPWM algorithm can be applied to the classical multilevel topologies; it has low computational complexity and it is suitable for online implementation. It was implemented in an FPGA and it was tested in laboratory with a five-level five-phase VSI. Since the algorithm is formulated for a generic number of phases, it was also particularized for the usual three-phase converters with three and four legs.*

5.1 Introduction

The multilevel multiphase space vector pulse-width modulation (SVPWM) presented in previous chapter does not take into account the joint-phase redundancy present in converters without neutral wire. In such converters, joint-phase redundancy is an important feature that allows extending the modulation index in the linear range and reducing switching losses. It can also be used to balance capacitors in diode-clamped and flying capacitor converters or to control the power delivered by the dc sources in cascaded full-bridge converters.

This chapter follows the same formulation used in the previous algorithm to develop a new SVPWM algorithm that takes into account joint-phase redundancy. It is also formulated in a multidimensional space and it is also based on a two-level multiphase SVPWM modulator. In fact, a significant contribution of the chapter is to demonstrate that the SVPWM for a multilevel P -phase converter with switching state redundancy can be carried out by means of a two-level $(P - 1)$ -phase SVPWM algorithm without redundancy. As in the algorithm developed in the previous chapter, the new algorithm is valid for any

The algorithm presented in the chapter has been accepted for publication in the journal *IEEE Transactions on Industrial Electronics* [173].

number of phases and levels, it can be applied to the classical multilevel topologies and it is well suited for online implementation in low cost devices. The new algorithm makes easy to choose the switching state sequence, even in converters with a large number of levels and phases in which there are huge number of redundant switching states.

The new multilevel multiphase SVPWM algorithm was implemented for a five-level five-phase inverter in a low-cost field-programmable gate array (FPGA). The model of the proposed hardware implementation was verified by simulation with Simulink. Finally, the real performance of the modulator was evaluated in the laboratory using a cascaded full-bridge inverter driving a five-phase induction motor.

This chapter is organized as follows. Section §5.2 describes the mathematical justification of the SVPWM algorithm in depth. It includes the problem formulation, the demonstration that a modulator with switching state redundancy can be made from other without redundancy, and the development of a new technique for choosing the switching state sequence. Finally, from mathematical treatment, the new multilevel multiphase SVPWM algorithm is obtained. Section §5.3 addresses the practical implementation of the algorithm in an FPGA. In section §5.4, the implementation is verified by comparing experimental measurements with simulation results. Some experimental results are also given so as to evaluate the real performance of the implemented modulation algorithm. In sections §5.5 and §5.6 the multilevel multiphase SVPWM technique is particularized for three-phase converters with three and four legs respectively, and new algorithms are compared with existing SVPWM techniques. Section §5.7 presents the conclusions of the work.

5.2 Algorithm Development

5.2.1 Algorithm Formulation

Voltage source converters synthesize a certain output voltage vector \mathbf{V}_s by means of a sequence of switching vectors during each modulation cycle. The output vector can be decomposed into the sum of an homopolar component $\bar{\mathbf{V}}_s$ plus a non-homopolar component $\tilde{\mathbf{V}}_s$ where the sum of the components of the non-homopolar vector is equal to zero:

$$\mathbf{V}_s = \bar{\mathbf{V}}_s + \tilde{\mathbf{V}}_s, \quad \sum_{k=1}^P \tilde{V}_s^k = 0. \quad (5.1)$$

The non homopolar component can be calculated as:

$$\tilde{\mathbf{V}}_s = \tilde{\mathbf{P}}\mathbf{V}_s \quad (5.2)$$

where $\tilde{\mathbf{P}}$ is the $P \times P$ real matrix:

$$\tilde{\mathbf{P}} = \frac{1}{P} \begin{bmatrix} P-1 & -1 & \dots & -1 \\ -1 & P-1 & \dots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & \dots & P-1 \end{bmatrix}. \quad (5.3)$$

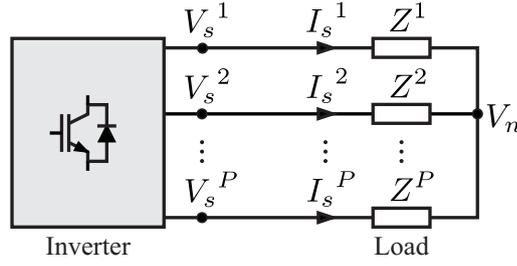


Figure 5.1: Multiphase system without neutral conductor.

In the multiphase system without neutral conductor of Figure 5.1, the current of phase k depends only on the non-homopolar component of \mathbf{V}_s :

$$I_s^k = \frac{\tilde{V}_s^k}{Z^k} - \frac{\sum_{i=1}^P \tilde{V}_s^i / Z^i}{\sum_{i=1}^P Z^k / Z^i}, \quad k = 1, 2, \dots, P. \quad (5.4)$$

As a consequence, to control the output current of the inverter it is necessary to deal with the non-homopolar component of the output voltage. Hence, in the synthesis of the output vector \mathbf{V}_s from the reference vector \mathbf{V}_r , the modulation algorithm must pay attention only to the non-homopolar component of those vectors and it can omit their homopolar component:

$$\tilde{\mathbf{V}}_r = \tilde{\mathbf{V}}_s \quad (5.5)$$

where $\tilde{\mathbf{V}}_r$ is calculated from the reference voltage \mathbf{V}_r as

$$\tilde{\mathbf{V}}_r = \tilde{\mathbf{P}}\mathbf{V}_r \in \mathbb{R}^P. \quad (5.6)$$

The homopolar component of \mathbf{V}_s changes the voltage of the neutral point of the load V_n . As in three-phase systems, this provides a degree of freedom that can be used to achieve different goals such as the reduction of the switching losses and the extension of the modulation index range.

As it was demonstrated in section §4.2.1 with typical multilevel topologies vectors and times can be normalized using expressions from (4.7) to (4.10). Therefore, (5.5) can be normalized as

$$\tilde{\mathbf{v}}_r = \tilde{\mathbf{v}}_s \quad (5.7)$$

where

$$\tilde{\mathbf{v}}_r = \tilde{\mathbf{P}}\mathbf{v}_r \in \mathbb{R}^P \quad (5.8)$$

$$\tilde{\mathbf{v}}_s = \tilde{\mathbf{P}}\mathbf{v}_s \in \mathbb{R}^P. \quad (5.9)$$

If the general modulation law in (4.11) is substituted in (5.7) then the following modulation law for converters with joint-phase redundancy is obtained:

$$\tilde{\mathbf{v}}_r = \sum_{j=1}^l \tilde{\mathbf{v}}_{sj} t_j, \quad \sum_{j=1}^l t_j = 1 \quad (5.10)$$

where

$$\tilde{\mathbf{v}}_{sj} = \tilde{\mathbf{P}}\mathbf{v}_{sj} \in \mathbb{R}^P. \quad (5.11)$$

If the homopolar components of the reference and the switching normalized vectors are both expressed as $\tilde{\mathbf{v}}_r = [\tilde{v}_r^1, \tilde{v}_r^2, \dots, \tilde{v}_r^P]^T$ and $\tilde{\mathbf{v}}_{sj} = [\tilde{v}_{sj}^1, \tilde{v}_{sj}^2, \dots, \tilde{v}_{sj}^P]^T$ then the modulation law in (5.10) can be rewritten in matrix format as:

$$\begin{bmatrix} 1 \\ \tilde{v}_r^1 \\ \tilde{v}_r^2 \\ \vdots \\ \tilde{v}_r^P \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \tilde{v}_{s1}^1 & \tilde{v}_{s2}^1 & \dots & \tilde{v}_{sl}^1 \\ \tilde{v}_{s1}^2 & \tilde{v}_{s2}^2 & \dots & \tilde{v}_{sl}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{v}_{s1}^P & \tilde{v}_{s2}^P & \dots & \tilde{v}_{sl}^P \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_l \end{bmatrix}. \quad (5.12)$$

The above system of linear equations constitutes the modulation law, which must be solved by the multilevel multiphase SVPWM algorithm. This modulation law has the same form of the modulation law in (4.14) derived in §4.2.1 for converters without joint-phase redundancy. Nevertheless, the method presented in previous chapter cannot be applied to the system of linear equations in (5.12) because it requires that the coefficients of the matrix are integer numbers. Therefore, it is necessary to find a new modulation law equivalent to (5.12), where the coefficients of matrix are integer numbers, in order to use such method for solving the modulation problem.

5.2.2 Equivalent Modulation Law

Since the non-homopolar component of any vector always lays in the plane $\tilde{\pi} : x^1 + x^2 + \dots + x^P = 0$ and the non-homopolar component of the vector $\mathbf{u} = [1, 1, \dots, 1]^T$ is the null vector, $\tilde{\mathbf{P}}\mathbf{u} = [0, 0, \dots, 0]^T$, then the transformation given by the matrix $\tilde{\mathbf{P}}$ is a projection onto the plane $\tilde{\pi}$ along the direction of \mathbf{u} . Expression in (5.7) shows that, in order to control the output current, the projections of the reference vector \mathbf{v}_r and the output vector \mathbf{v}_s onto the plane $\tilde{\pi}$ must be the same:

$$\tilde{\mathbf{P}}\mathbf{v}_r = \tilde{\mathbf{P}}\mathbf{v}_s. \quad (5.13)$$

As Figure 5.2 shows, if two vectors have the same projection onto one plane then they will have the same projection onto another plane if the direction of projection does not change. Therefore, by changing the plane of projection it is possible to devise an equivalent modulation law:

$$\check{\mathbf{P}}\mathbf{v}_r = \check{\mathbf{P}}\mathbf{v}_s. \quad (5.14)$$

If the plane $\check{\pi} : x^P = 0$ is selected then the following transformation is obtained:

$$\check{\mathbf{P}} = \begin{bmatrix} 1 & 0 & \dots & 0 & -1 \\ 0 & 1 & \dots & 0 & -1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -1 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}. \quad (5.15)$$

Equivalence of both modulation laws, (5.13) and (5.14), can be verified taking into account that the transformations $\tilde{\mathbf{P}}$ and $\check{\mathbf{P}}$ have the properties $\check{\mathbf{P}}\tilde{\mathbf{P}} = \tilde{\mathbf{P}}$ and $\tilde{\mathbf{P}}\check{\mathbf{P}} = \check{\mathbf{P}}$.

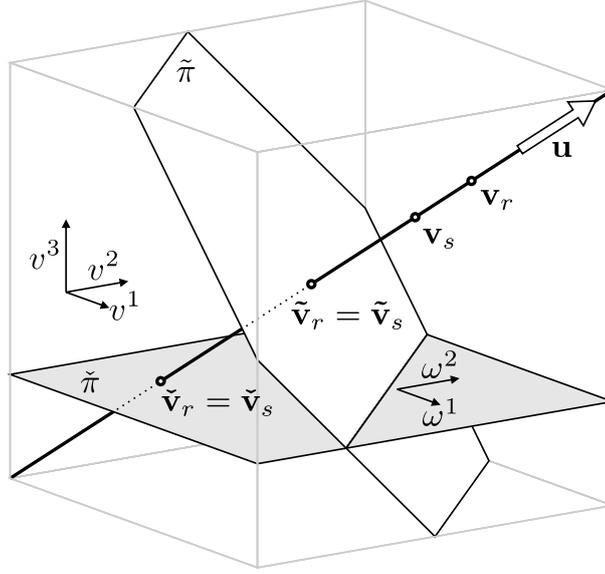


Figure 5.2: Two vectors with the same projection onto different planes.

From (5.14), the equivalent condition of (5.7) is

$$\check{\mathbf{v}}_r = \check{\mathbf{v}}_s \quad (5.16)$$

where the projected voltage reference $\check{\mathbf{v}}_r$ and the projected output vector $\check{\mathbf{v}}_s$ are

$$\check{\mathbf{v}}_r = \check{\mathbf{P}}\mathbf{v}_r \in \mathbb{R}^P \quad (5.17)$$

$$\check{\mathbf{v}}_s = \check{\mathbf{P}}\mathbf{v}_s \in \mathbb{R}^P \quad (5.18)$$

and the new modulation law, which is equivalent to (5.10), is

$$\check{\mathbf{v}}_r = \sum_{j=1}^l \check{\mathbf{v}}_{sj} t_j, \quad \sum_{j=1}^l t_j = 1 \quad (5.19)$$

where $\check{\mathbf{v}}_{sj}$ are the projections of the switching vectors \mathbf{v}_{sj} onto the plane $\tilde{\pi}$:

$$\check{\mathbf{v}}_{sj} = \check{\mathbf{P}}\mathbf{v}_{sj} \in \mathbb{Z}^P. \quad (5.20)$$

Since the application of the transformation $\check{\mathbf{P}}$ only involves a number of differences and the components of the normalized switching vectors \mathbf{v}_{sj} are always integer numbers, the components of the projected switching vectors $\check{\mathbf{v}}_{sj}$ are integer numbers as well.

If the new projected switching vectors are expressed as $\check{\mathbf{v}}_r = [\check{v}_r^1, \check{v}_r^2, \dots, \check{v}_r^P]^T$ and $\check{\mathbf{v}}_{sj} = [\check{v}_{sj}^1, \check{v}_{sj}^2, \dots, \check{v}_{sj}^P]^T$ then the equivalent modulation law in (5.19) can be rewritten in matrix form as

$$\begin{bmatrix} 1 \\ \check{v}_r^1 \\ \check{v}_r^2 \\ \vdots \\ \check{v}_r^P \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \check{v}_{s1}^1 & \check{v}_{s2}^1 & \dots & \check{v}_{sl}^1 \\ \check{v}_{s1}^2 & \check{v}_{s2}^2 & \dots & \check{v}_{sl}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \check{v}_{s1}^P & \check{v}_{s2}^P & \dots & \check{v}_{sl}^P \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_l \end{bmatrix}. \quad (5.21)$$

By using the new projection $\check{\mathbf{P}}$ the coefficients of this system of linear equations are integer numbers. Hence, the method presented in §4.2 can be applied to solve the equivalent modulation law in (5.21). Nevertheless, the new modulation law can be further simplified by removing the last equation of the linear system because, by definition, the last component of all projected vectors is equal to zero ($\check{v}_r^P = 0$ and $\check{v}_{s_j}^P = 0$). Without that equation the simplified modulation law can be written as

$$\begin{bmatrix} 1 \\ \omega_r^1 \\ \omega_r^2 \\ \vdots \\ \omega_r^{P-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \omega_{s_1}^1 & \omega_{s_2}^1 & \dots & \omega_{s_l}^1 \\ \omega_{s_1}^2 & \omega_{s_2}^2 & \dots & \omega_{s_l}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{s_1}^{P-1} & \omega_{s_2}^{P-1} & \dots & \omega_{s_l}^{P-1} \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_l \end{bmatrix} \quad (5.22)$$

where vectors $\boldsymbol{\omega}_r$ and $\boldsymbol{\omega}_{s_j}$ are the projected vectors $\check{\mathbf{v}}_r$ and $\check{\mathbf{v}}_{s_j}$ without the last component:

$$\boldsymbol{\omega}_r = [\omega_r^1, \dots, \omega_r^{P-1}]^T = [\check{v}_r^1, \dots, \check{v}_r^{P-1}]^T \quad (5.23)$$

$$\boldsymbol{\omega}_{s_j} = [\omega_{s_j}^1, \dots, \omega_{s_j}^{P-1}]^T = [\check{v}_{s_j}^1, \dots, \check{v}_{s_j}^{P-1}]^T. \quad (5.24)$$

Since the transformed vectors represent the switching states of the converter, $\boldsymbol{\omega}_{s_j}$ are the space vectors of the converter and $\boldsymbol{\omega}_r$ is the reference space vector. The space vectors can be directly calculated from the reference and the switching vectors as:

$$\boldsymbol{\omega}_r = \mathbf{T}_\omega \mathbf{v}_r \in \mathbb{R}^{P-1} \quad (5.25)$$

$$\boldsymbol{\omega}_{s_j} = \mathbf{T}_\omega \mathbf{v}_{s_j} \in \mathbb{Z}^{P-1} \quad (5.26)$$

where \mathbf{T}_ω is equal to $\check{\mathbf{P}}$ without the last row:

$$\mathbf{T}_\omega = \begin{bmatrix} 1 & 0 & \dots & 0 & -1 \\ 0 & 1 & \dots & 0 & -1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -1 \end{bmatrix}. \quad (5.27)$$

Finally, the coefficients of the linear system of equations are integer numbers. Therefore the transformed modulation law in (5.22) is identical to the modulation law in (4.14) solved in the previous chapter. Thus, this expression demonstrates that a multilevel P -phase modulator with joint-phase redundancy can be made from a multilevel $(P-1)$ -phase modulator without redundancy. In §4.2 the multilevel modulation problem was decomposed into the sum of a two-level problem plus a displacement. As a consequence, the SVPWM for a P -phase converter with redundancy can be carried out by means of two-level $(P-1)$ -phase SVPWM algorithm without redundancy.

5.2.3 Modulation Problem Solution

The algorithm in §4.2.4 will be used to solve the modulation law in (5.22). Such algorithm shows that the switching sequence will have $l = P$ vectors, and its application

requires calculating the integer and the fractional parts of the reference space vector $\boldsymbol{\omega}_r$:

$$\boldsymbol{\omega}_i = \text{integ}(\boldsymbol{\omega}_r) \in \mathbb{Z}^{P-1} \quad (5.28)$$

$$\boldsymbol{\omega}_f = \boldsymbol{\omega}_r - \boldsymbol{\omega}_i \in \mathbb{R}^{P-1}. \quad (5.29)$$

The fractional part of the reference space vector $\boldsymbol{\omega}_f$ is introduced in the two-level multi-phase SVPWM developed in §4.2.3 to calculate the sequence of P displaced space vectors $\{\boldsymbol{\omega}_{d1}, \boldsymbol{\omega}_{d2}, \dots, \boldsymbol{\omega}_{dP}\}$ and their corresponding dwell times $\{\tau_1, \tau_2, \dots, \tau_P\}$. The sequence of space vectors $\{\boldsymbol{\omega}_{s1}, \boldsymbol{\omega}_{s2}, \dots, \boldsymbol{\omega}_{sP}\}$ that solves the transformed modulation law in (5.22) is calculated by adding the integer part of the reference $\boldsymbol{\omega}_i$ to the displaced space vector sequence:

$$\boldsymbol{\omega}_{sj} = \boldsymbol{\omega}_i + \boldsymbol{\omega}_{dj}, \quad j = 1, 2, \dots, P. \quad (5.30)$$

The dwell time corresponding to the vectors $\boldsymbol{\omega}_{sj}$ is the same as the dwell time τ_j of the vector $\boldsymbol{\omega}_{dj}$.

Once the transformed modulation law has been solved the next step is to find the switching vectors \mathbf{v}_{sj} that correspond to each vector $\boldsymbol{\omega}_{sj}$ in the space vector sequence. Nevertheless, all the switching vectors with the same non-homopolar component correspond to the same space vector:

$$\mathbf{T}_\omega \mathbf{v}_{sj} = \mathbf{T}_\omega (\mathbf{v}_{sj} + n\mathbf{u}) = \boldsymbol{\omega}_{sj}, \quad j = 1, 2, \dots, P. \quad (5.31)$$

Hence, the inverse transformation of (5.26) is not unique and must be written as a function of an integer parameter n :

$$\mathbf{v}_{sj} = \mathbf{T}_v \boldsymbol{\omega}_{sj} + n\mathbf{u}, \quad n \in \mathbb{Z} \quad (5.32)$$

where the matrix \mathbf{T}_v adds an extra zero component to the end of the vector $\boldsymbol{\omega}_s$:

$$\mathbf{T}_v = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 0 \end{bmatrix}. \quad (5.33)$$

Consequently, from the space vector sequence it is possible to obtain many different switching sequences, that obey the modulation law in (5.12), by using different values of n . All switching vectors \mathbf{v}_{sj} that correspond to the same space vector $\boldsymbol{\omega}_{sj}$ are redundant, and the switching time that those redundant vectors share is the dwell time τ_j which corresponds to the space vector $\boldsymbol{\omega}_{sj}$.

As the number of levels or phases increases, the number of redundant switching vectors and the complexity of selecting them increase dramatically. Therefore, a technique that makes easy to choose the final switching vectors among all redundant vectors is required.

5.2.4 Selection of the Vectors of the Switching Sequence

Until this point, the number of levels of the inverter has not been taken into account. Multilevel converters have a limited number of levels, therefore not all values of n can be

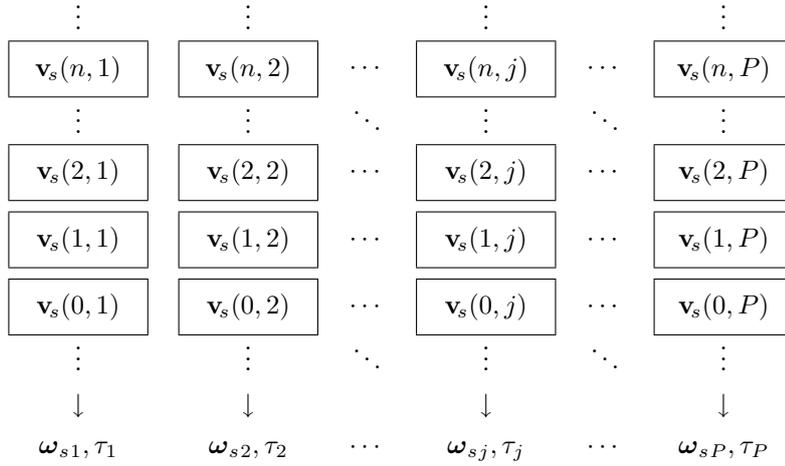


Figure 5.3: All switching vectors arranged in a table.

used in expression (5.32) in order to calculate the final switching vectors. Very high or very low values of n will result in switching vectors with levels that cannot be reached by the inverter. Therefore, it is necessary to calculate the subset of redundant vectors that are really useful from the infinite possibilities provided by expression in (5.32).

If (5.30) is substituted in (5.32) then all the vectors of the switching sequence can be written as a function of two integer parameters n and j :

$$\mathbf{v}_s(n, j) = \mathbf{T}_v(\omega_i + \omega_{dj}) + n\mathbf{u}, \quad n \in \mathbb{Z} \text{ and } j \in [1, P] \in \mathbb{N}. \quad (5.34)$$

Figure 5.3 shows all switching vectors given by (5.34) arranged in a table. The switching vectors that belong to the column j map to the same space vector ω_{sj} . Hence, they are redundant vectors and share the dwell time τ_j . The switching vectors that belong to the row n form an ordered sequence of adjacent vectors because the displaced space vectors obtained with the two-level modulator are a sequence of adjacent vectors too.

In order to obtain the final switching sequence, the modulation strategy must take at least one vector from each column. A straightforward method could be selecting all vectors in a row, that is, by using a constant value of n . Those vectors form a sequence of adjacent vectors so the number of switchings is minimized. Nevertheless, if a row has at least one vector that cannot be reached by the converter then none of the vectors of the row would be used. Hence, this simple method wastes some vectors and it results in a reduction of the maximum modulation index and a bad utilization of the resources of the dc link.

A more efficient method consists of making a sequence of adjacent vectors taking vectors from several rows but without increasing the number of switchings. The last vector of the row n and the first one of the upper row can be obtained by means of (5.34) as

$$\mathbf{v}_s(n, P) = \mathbf{T}_v(\omega_i + \omega_{dP}) + n\mathbf{u} \quad (5.35)$$

$$\mathbf{v}_s(n+1, 1) = \mathbf{T}_v(\omega_i + \omega_{d1}) + (n+1)\mathbf{u}. \quad (5.36)$$

Taking into account that the first and the last vectors of the displaced space vector sequence are always $\omega_{d1} = [0, 0, \dots, 0]^T$ and $\omega_{dP} = [1, 1, \dots, 1]^T$, and substituting them

in (5.35) and (5.36), the above vectors can be calculated as

$$\mathbf{v}_s(n, P) = \mathbf{T}_v \boldsymbol{\omega}_i + [1, 1, \dots, 1, 0]^T + n \mathbf{u} \quad (5.37)$$

$$\mathbf{v}_s(n+1, 1) = \mathbf{T}_v \boldsymbol{\omega}_i + [0, 0, \dots, 0, 0]^T + (n+1) \mathbf{u}. \quad (5.38)$$

The subtraction of two vectors gives

$$\mathbf{v}_s(n+1, 1) - \mathbf{v}_s(n, P) = [0, 0, \dots, 0, 1]^T. \quad (5.39)$$

Therefore, the last vector of a row and the first vector of the upper row are adjacent. Since vectors within the same row are also adjacent then all switching vectors in Figure 5.3 can be rearranged in an infinite string of adjacent vectors:

$$\{\dots, \mathbf{v}_s(n, j-1), \mathbf{v}_s(n, j), \mathbf{v}_s(n, j+1), \dots, \mathbf{v}_s(n, P), \mathbf{v}_s(n+1, 1), \dots\}. \quad (5.40)$$

Each vector in this sequence can be characterized by an integer index q defined as the sum of its components:

$$q = \sum_{k=1}^P v_s^k(n, j) \in \mathbb{Z}. \quad (5.41)$$

If the transformation (5.33) is taken into account and (5.34) is substituted in (5.41) then the index q can be calculated as

$$q = \sum_{k=1}^P \omega_i^k + \sum_{k=1}^P \omega_{d_j}^k + nP. \quad (5.42)$$

Defining q_i as

$$q_i = \sum_{k=1}^P \omega_i^k \quad (5.43)$$

and, since the displaced switching vectors $\boldsymbol{\omega}_{d_j}$ are the result of reordering the rows of a triangular matrix $\hat{\mathbf{D}}$, the sum of their components is

$$\sum_{k=1}^P \omega_{d_j}^k = j - 1. \quad (5.44)$$

Therefore, the index q corresponding to each switching vector $\mathbf{v}_s(n, j)$ can be calculated from the indices n and j as

$$q = q_i + (j - 1) + nP. \quad (5.45)$$

Since the value of j is bounded in the interval $[1, P]$ then there is a one-to-one correspondence between the switching vectors \mathbf{v}_{s_j} and the indices q . Therefore the string of adjacent vectors in (5.40) can be rewritten as a function of q as:

$$\{\mathbf{v}_s(-\infty), \dots, \mathbf{v}_s(q-1), \mathbf{v}_s(q), \mathbf{v}_s(q+1), \dots, \mathbf{v}_s(+\infty)\} \quad (5.46)$$

where

$$\mathbf{v}_s(q) = \mathbf{v}_s(n, j). \quad (5.47)$$

Although it is possible to make converters with different number of levels in each phase, usually, practical converters have all phases identical. Nevertheless, a modulation technique that can handle different number of levels in each phase could be useful for instance during a fault when one or more phases reduce the number of available levels [174]. The case in which each phase k is able to generate all levels within a certain interval will be studied¹:

$$v_s^k \in [N_{\min}^k, N_{\max}^k], \quad k = 1, 2, \dots, P. \quad (5.48)$$

Displaced space vectors ω_{dj} are the result of reordering the rows of the upper triangular matrix $\hat{\mathbf{D}}$. Therefore, in addition to being the adjacent vectors, they are also sorted out in ascending order. Hence, every displaced switching vector is identical to the previous one except by one component that has increased one unit. Since the switching vectors $\mathbf{v}_s(q) = \mathbf{v}_s(n, j)$ are obtained from the displaced switching vectors by means of (5.34), they have the same property too. As a consequence, for each phase k , there are a couple of vectors that bound the interval within the string of adjacent vectors where all switching vectors make use of all available levels in such phase. Therefore, the indices q_{\min}^k and q_{\max}^k , which correspond to those bounding vectors, define the interval of indices q of the switching vectors in which the converter is able to generate the phase k :

$$v_s^k \in [N_{\min}^k, N_{\max}^k] \quad \text{if} \quad q \in [q_{\min}^k, q_{\max}^k]. \quad (5.49)$$

Those indices can be calculated from N_{\min}^k and N_{\max}^k as

$$q_{\min}^k = \Delta q^k + N_{\min}^k P \quad (5.50)$$

$$q_{\max}^k = \Delta q^k + N_{\max}^k P + (P - 1) \quad (5.51)$$

where

$$\Delta q^k = \begin{cases} q_i - \omega_i^k P - \sum_{j=1}^P \omega_{dj}^k, & \text{if } k = 1, \dots, P - 1 \\ q_i, & \text{if } k = P. \end{cases} \quad (5.52)$$

The interval of indices $[q_{\min}, q_{\max}]$ where all switching vectors of the string can be generated by the converter is the intersection of the intervals $[q_{\min}^k, q_{\max}^k]$ of all phases:

$$q_{\min} = \max(q_{\min}^1, q_{\min}^2, \dots, q_{\min}^P) \quad (5.53)$$

$$q_{\max} = \min(q_{\max}^1, q_{\max}^2, \dots, q_{\max}^P). \quad (5.54)$$

After considering the physical limits of the converter, the infinite string of adjacent switching vectors in (5.46) is reduced to the following substring of useful vectors:

$$\{\mathbf{v}_s(q_{\min}), \dots, \mathbf{v}_s(q), \dots, \mathbf{v}_s(q_{\max})\}. \quad (5.55)$$

The set of P consecutive switching vectors, which permits to solve the modulation law in (5.10), must be selected within this substring. If the substring is too short and it has not at least P elements then it is not possible to approximate the reference accurately. That is, the reference lies in the overmodulation region if

$$q_{\max} - q_{\min} + 1 < P. \quad (5.56)$$

¹In this case the number of switching states of the converter is equal to $\prod_{k=1}^P N^k$ where $N^k = N_{\max}^k - N_{\min}^k + 1$ and the number of space vectors is $\prod_{k=1}^P N^k - \prod_{k=1}^P (N^k - 1)$.

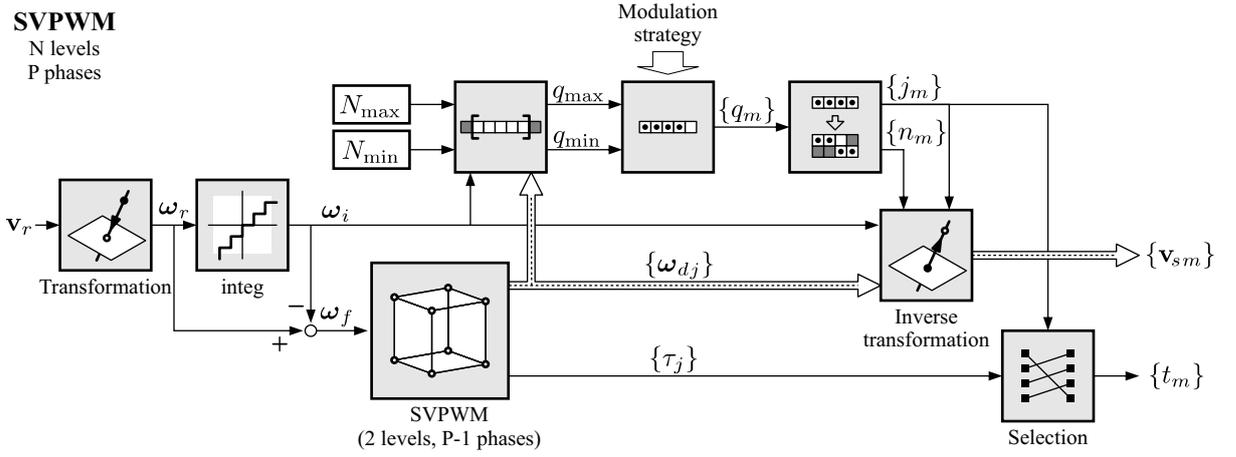


Figure 5.4: Block diagram of the new multilevel multiphase SVPWM algorithm.

The sequence of P switching vectors can be calculated from P consecutive indices selected within the interval $[q_{\min}, q_{\max}]$ according to a certain modulation strategy:

$$\{q_1, \dots, q_m, \dots, q_P\}. \quad (5.57)$$

The selection has influence in the capacitor balancing in diode clamped and flying capacitor converters, or in the power sharing of dc sources in cascaded full-bridge converters. To obtain each vector of the switching sequence $\mathbf{v}_s(q_m)$ it is necessary to calculate the values of n_m and j_m that correspond to the index q_m . If (5.45) is solved taking into account that $1 \leq j \leq P$ then the indices n_m and j_m that correspond to each index q_m in (5.57) can be calculated as

$$n_m = \text{integ} \left(\frac{q_m - q_i}{P} \right) \quad (5.58)$$

$$j_m = q_m - q_i - n_m P + 1. \quad (5.59)$$

If these indices are substituted in (5.34) then each element of the switching vector sequence $\{\mathbf{v}_{s1}, \dots, \mathbf{v}_{sm}, \dots, \mathbf{v}_{sP}\}$ is obtained as

$$\mathbf{v}_{sm} = \mathbf{v}_s(q_m) = \mathbf{v}_s(n_m, j_m) \quad (5.60)$$

and its corresponding switching time is

$$t_m = \tau_{j_m}. \quad (5.61)$$

5.2.5 New Multilevel Multiphase SVPWM Algorithm

Although the previous demonstration is quite cumbersome, the algorithm derived from that mathematical treatment is very simple. The steps of the proposed SVPWM algorithm with joint-phase redundancy, which are summarized in Figure 5.4, are:

1. Obtain the normalized reference vector \mathbf{v}_r from the reference voltage \mathbf{V}_r by means of (4.7).

2. Calculate the reference space vector $\boldsymbol{\omega}_r$ from \mathbf{v}_r by using expression in (5.25).
3. Decompose the reference space vector into the sum of its integer part $\boldsymbol{\omega}_i$ and its fractional part $\boldsymbol{\omega}_f$ with (5.28) and (5.29) respectively.
4. From the fractional part, obtain the sequence of displaced space vectors $\boldsymbol{\omega}_{dj}$ and their corresponding switching times τ_j by means of the two-level $(P - 1)$ -phase SVPWM presented in §4.2.3.
5. From the integer part, obtain the value of q_i by means of expression in (5.43).
6. Calculate the intervals $[q_{\min}^k, q_{\max}^k]$ that correspond to each phase k by means of (5.50) and (5.51).
7. Determine the bounding indices q_{\min} and q_{\max} by using expressions in (5.53) and (5.54) respectively.
8. Test if the reference does not lie in the overmodulation region with the condition given in (5.56).
9. Select P consecutive integer numbers $\{q_m\}$ within the interval $[q_{\max}, q_{\min}]$ according to any desired modulation strategy.
10. Calculate the values of n_m and j_m that correspond to each selected index q_m by means of (5.58) and (5.59) respectively.
11. Obtain the vectors of the final switching sequence $\{\mathbf{v}_{sm}\}$ and their switching times $\{t_m\}$ with expressions in (5.60) and (5.61) respectively.

Finally, the trigger signals can be generated from the switching vectors and the switching times. The relationship between the switching states and the particular trigger signals of transistors depends on the multilevel topology as it was shown in chapter §2.

5.2.6 Example

The simplicity of the algorithm is here shown by means of an example in which the previous steps are followed. Let us consider the same example as in section §4.2.5 where the normalized reference vector is $\mathbf{v}_r = [1.43, 1.13, -0.73, -1.58, -0.25]^T$. In this case, the reference space vector is the following four-dimensional (4D) vector:

$$\boldsymbol{\omega}_r = \mathbf{T}_{\boldsymbol{\omega}} \mathbf{v}_r = [1.68, 1.38, -0.48, -1.33]^T \quad (5.62)$$

which is composed of the following integer and fractional parts:

$$\boldsymbol{\omega}_i = \text{integ}(\boldsymbol{\omega}_r) = [1, 1, -1, -2]^T \quad (5.63)$$

$$\boldsymbol{\omega}_f = \boldsymbol{\omega}_r - \boldsymbol{\omega}_i = [0.68, 0.38, 0.52, 0.67]^T. \quad (5.64)$$

If the four-phase version of the two-level multiphase algorithm in §4.2.3 is applied to the vector $\boldsymbol{\omega}_f$ then the following displaced space vector sequence and with their corresponding dwell times are obtained:

$$\begin{aligned}
\boldsymbol{\omega}_{d1} &= [0, 0, 0, 0]^T & \rightarrow & \tau_1 = 0.32 \\
\boldsymbol{\omega}_{d2} &= [1, 0, 0, 0]^T & \rightarrow & \tau_2 = 0.01 \\
\boldsymbol{\omega}_{d3} &= [1, 0, 0, 1]^T & \rightarrow & \tau_3 = 0.15 \\
\boldsymbol{\omega}_{d4} &= [1, 0, 1, 1]^T & \rightarrow & \tau_4 = 0.14 \\
\boldsymbol{\omega}_{d5} &= [1, 1, 1, 1]^T & \rightarrow & \tau_5 = 0.38.
\end{aligned} \tag{5.65}$$

The value of q_i can be calculated from the integer part of the reference as

$$q_i = \omega_i^a + \omega_i^b + \omega_i^c + \omega_i^d = -1. \tag{5.66}$$

If a five-level cascaded full-bridge inverter is considered then:

$$\begin{aligned}
[N_{\min}^a, N_{\max}^a] &= [N_{\min}^b, N_{\max}^b] = [N_{\min}^c, N_{\max}^c] = \\
&= [N_{\min}^d, N_{\max}^d] = [N_{\min}^e, N_{\max}^e] = [-2, 2].
\end{aligned} \tag{5.67}$$

Further, the intervals of indices $[q_{\min}^k, q_{\max}^k]$ corresponding to the switching vector in which the converter is able to generate the phase k are

$$\begin{aligned}
[q_{\min}^a, q_{\max}^a] &= [-20, 4] \\
[q_{\min}^b, q_{\max}^b] &= [-17, 7] \\
[q_{\min}^c, q_{\max}^c] &= [-8, 16] \\
[q_{\min}^d, q_{\max}^d] &= [-4, 20] \\
[q_{\min}^e, q_{\max}^e] &= [-11, 13].
\end{aligned} \tag{5.68}$$

Consequently, the range of the indices that correspond to useful switching vectors is

$$[q_{\min}, q_{\max}] = [-4, 4]. \tag{5.69}$$

The length of this interval is greater than five, therefore the reference vector is in the linear region and it can be accurately synthesized. Among all available possibilities, the following five consecutive indices in the top of the range have been selected:

$$\{q_1, q_2, q_3, q_4, q_5\} = \{0, 1, 2, 3, 4\}. \tag{5.70}$$

The values of n_m and j_m that correspond to the previous q_m indices are

$$\begin{aligned}
n_1 = 0 \quad n_2 = 0 \quad n_3 = 0 \quad n_4 = 0 \quad n_5 = 1 \\
j_1 = 2 \quad j_2 = 3 \quad j_3 = 4 \quad j_4 = 5 \quad j_5 = 1.
\end{aligned} \tag{5.71}$$

From (5.60), the final switching vector sequence is

$$\begin{aligned}
\mathbf{v}_{s1} &= \mathbf{v}_s(q_1) = \mathbf{v}_s(n_1, j_1) = [2, 1, -1, -2, 0]^T \\
\mathbf{v}_{s2} &= \mathbf{v}_s(q_2) = \mathbf{v}_s(n_2, j_2) = [2, 1, -1, -1, 0]^T \\
\mathbf{v}_{s3} &= \mathbf{v}_s(q_3) = \mathbf{v}_s(n_3, j_3) = [2, 1, 0, -1, 0]^T \\
\mathbf{v}_{s4} &= \mathbf{v}_s(q_4) = \mathbf{v}_s(n_4, j_4) = [2, 2, 0, -1, 0]^T \\
\mathbf{v}_{s5} &= \mathbf{v}_s(q_5) = \mathbf{v}_s(n_5, j_5) = [2, 2, 0, -1, 1]^T.
\end{aligned} \tag{5.72}$$

As expected, all vectors can be generated by the inverter and consecutive vectors of the sequence are adjacent. Therefore, the number of switchings is minimized.

Finally, the switching times are calculated from switching times provided by the two-level modulator by means of (5.61):

$$\begin{aligned}
 t_1 &= \tau_2 = 0.01 \\
 t_2 &= \tau_3 = 0.15 \\
 t_3 &= \tau_4 = 0.14 \\
 t_4 &= \tau_5 = 0.38 \\
 t_5 &= \tau_1 = 0.32.
 \end{aligned} \tag{5.73}$$

Table 5.1 shows a number of vectors $\mathbf{v}_s(n, j)$ that have been calculated with a set of indices n and j using data of the example. The vectors were calculated by means of (5.34), and after that, the index q corresponding to each one of them was obtained using the definition given in (5.41). The table verifies that all calculations made in the previous example are right. All switching vectors form an infinite string of adjacent vectors. In each phase there is a substring in which all output levels corresponding to that phase are in the interval of levels available in the converter. For instance, all vectors within the interval $[q_{\min}^a, q_{\max}^a] = [-20, 4]$ have the levels of phase a in the interval $[N_{\min}^a, N_{\max}^a] = [-2, 2]$. The intersections of all intervals are the switching vectors in the interval between $q_{\min} = -4$ and $q_{\max} = 4$. All those vectors have all levels in all phases between -2 and 2 . Therefore, all of them can be generated by the converter. Finally, the selected switching vectors are the last five elements in this interval.

5.2.7 Algorithm Features

As the new SVPWM algorithm makes use of the two-level modulator without redundancy presented in previous chapter both of them have many common features. Its computational cost is low and independent of the number of levels. Lookup tables, trigonometric functions or memories to store predefined switching sequences are not needed. Hence, the algorithm is well suited for real time implementation in low cost devices. The new proposed modulation technique, even with the high number (N^P) of available space vectors in multilevel multiphase converters, handles all space vectors without discarding any of them. In addition, the provided switching vector sequence is in order and consequently it minimizes the number of switchings. Hence no extra effort is needed to achieve this significant goal.

The performance of the new SVPWM algorithm is better than the previous SVPWM algorithm when it is applied to a converter with joint-phase switching redundancy. For the same number of phases the switching sequence is made with one vector less. Therefore, the number of switchings is approximately reduced by a factor of $P/(P+1)$. The modulation index m , as it was defined in (4.45), has a range of

$$0 \leq m \leq \frac{N-1}{2 \cos \frac{\pi}{2P}}. \tag{5.74}$$

Therefore, the new algorithm extends the modulation index leading to a better utilization of the resources of the dc link.

TABLE 5.1
SWITCHING VECTOR STRING OF THE EXAMPLE.

| n | j | $[v_s^a, v_s^b, v_s^c, v_s^d, v_s^e]^T$ | q |
|----------|----------|---|-------------------|
| \vdots | \vdots | \vdots | \vdots |
| 3 | 2 | [5, 4, 2, 1, 3] ^T | 15 |
| 3 | 1 | [4, 4, 2, 1, 3] ^T | 14 |
| 2 | 5 | [4, 4, 2, 1, 2] ^T | 13 |
| 2 | 4 | [4, 3, 2, 1, 2] ^T | 12 |
| 2 | 3 | [4, 3, 1, 1, 2] ^T | 11 |
| 2 | 2 | [4, 3, 1, 0, 2] ^T | 10 |
| 2 | 1 | [3, 3, 1, 0, 2] ^T | 9 |
| 1 | 5 | [3, 3, 1, 0, 1] ^T | 8 |
| 1 | 4 | [3, 2, 1, 0, 1] ^T | 7 |
| 1 | 3 | [3, 2, 0, 0, 1] ^T | 6 |
| 1 | 2 | [3, 2, 0, -1, 1] ^T | 5 |
| 1 | 1 | [2, 2, 0, -1, 1] ^T | 4 \triangleleft |
| 0 | 5 | [2, 2, 0, -1, 0] ^T | 3 \triangleleft |
| 0 | 4 | [2, 1, 0, -1, 0] ^T | 2 \triangleleft |
| 0 | 3 | [2, 1, -1, -1, 0] ^T | 1 \triangleleft |
| 0 | 2 | [2, 1, -1, -2, 0] ^T | 0 \triangleleft |
| 0 | 1 | [1, 1, -1, -2, 0] ^T | -1 |
| -1 | 5 | [1, 1, -1, -2, -1] ^T | -2 |
| -1 | 4 | [1, 0, -1, -2, -1] ^T | -3 |
| -1 | 3 | [1, 0, -2, -2, -1] ^T | -4 |
| -1 | 2 | [1, 0, -2, -3, -1] ^T | -5 |
| -1 | 1 | [0, 0, -2, -3, -1] ^T | -6 |
| -2 | 5 | [0, 0, -2, -3, -2] ^T | -7 |
| -2 | 4 | [0, -1, -2, -3, -2] ^T | -8 |
| -2 | 3 | [0, -1, -3, -3, -2] ^T | -9 |
| -2 | 2 | [0, -1, -3, -4, -2] ^T | -10 |
| -2 | 1 | [-1, -1, -3, -4, -2] ^T | -11 |
| -3 | 5 | [-1, -1, -3, -4, -3] ^T | -12 |
| -3 | 4 | [-1, -2, -3, -4, -3] ^T | -13 |
| -3 | 3 | [-1, -2, -4, -4, -3] ^T | -14 |
| -3 | 2 | [-1, -2, -4, -5, -3] ^T | -15 |
| -3 | 1 | [-2, -2, -4, -5, -3] ^T | -16 |
| -4 | 5 | [-2, -2, -4, -5, -4] ^T | -17 |
| -4 | 4 | [-2, -3, -4, -5, -4] ^T | -18 |
| -4 | 3 | [-2, -3, -5, -5, -4] ^T | -19 |
| -4 | 2 | [-2, -3, -5, -6, -4] ^T | -20 |
| -4 | 1 | [-3, -3, -5, -6, -4] ^T | -21 |
| \vdots | \vdots | \vdots | \vdots |

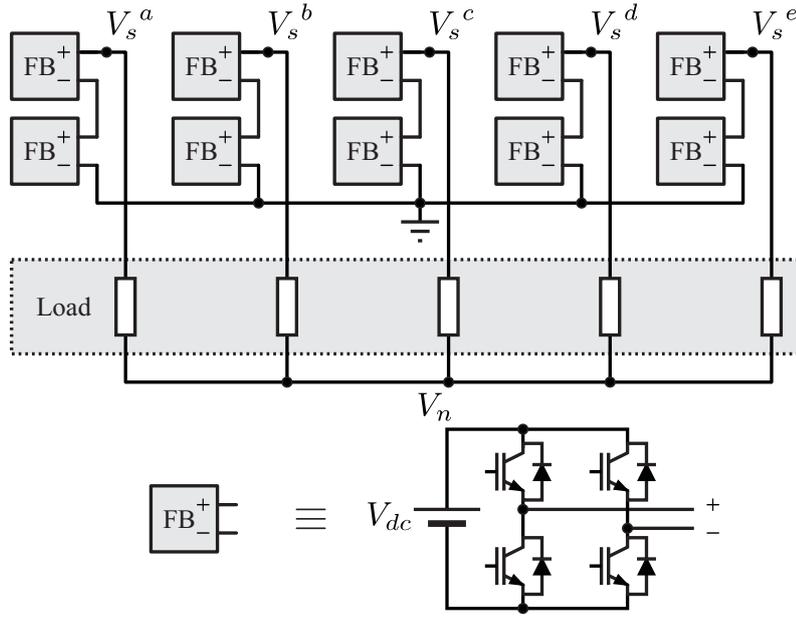


Figure 5.5: Five-level five-phase cascaded full-bridge inverter.

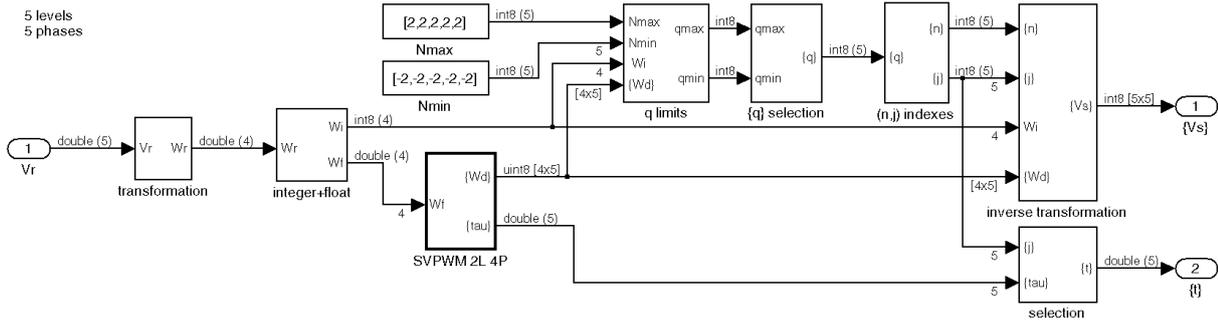


Figure 5.6: Simulink model of the hardware implementation of the SVPWM algorithm.

5.3 Hardware Implementation

The Digilent S3 board, which hosts a XC3S200 FPGA from Xilinx, was used to implement the new SVPWM algorithm for the five-level five-phase cascaded full bridge inverter shown in Figure 5.5. Before carrying out the description of the algorithm using very-high-speed integrated circuit hardware description language (VHDL), the model of the hardware implementation was first tested by simulation using the Simulink model shown in Figure 5.6. The algorithm implementation follows the steps given in section §5.2.5. In that figure, the block `transformation` implements (5.25) to calculate the 4D vector ω_r . The fractional part of the reference space vector ω_f feeds the block `SVPWM 2L 4P`. This block is the two-level multiphase SVPWM, developed in §4.3 and particularized for four phases. The block `q limits` implements (5.50), (5.51), (5.53) and (5.54) to calculate the range of indices $[q_{\min}, q_{\max}]$ that correspond to the set of useful switching vectors. The block `{q} selection` selects, in the middle of the range, the following five consecutive indices: $q_1 = q_3 - 2$, $q_2 = q_3 - 1$, $q_3 = \text{integ}((q_{\max} + q_{\min})/2)$, $q_4 = q_3 + 1$ and $q_5 = q_3 + 2$. The overmodulation condition in (5.56) was not tested in our implementation. Next, the

TABLE 5.2
RESOURCES SUMMARY.

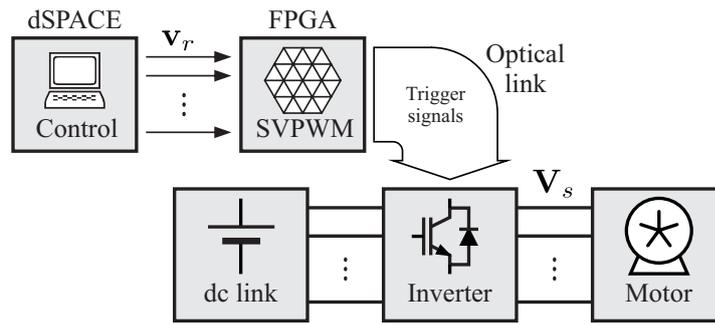
| | | | |
|---|--------------------|------|--|
| Target Device: | xc3s200 | | |
| Number of Slice Flip Flops: | 2,307 out of 3,840 | 60% | |
| Number of 4 input LUTs: | 2,815 out of 3,840 | 73% | |
| Number of occupied Slices: | 1,918 out of 1,920 | 99% | |
| Total Number 4 input LUTs: | 2,856 out of 3,840 | 74% | |
| Number of bonded IOBs: | 91 out of 173 | 52% | |
| IOB Flip Flops: | 67 | | |
| Number of Block RAMs: | 0 out of 12 | 0% | |
| Number of MULT18X18s: | 0 out of 12 | 0% | |
| Number of GCLKs: | 8 out of 8 | 100% | |
| Number of Startups: | 1 out of 1 | 100% | |
| Total equivalent gate count for design: | 38,879 | | |

block (n, j) indices implements the expressions in (5.58) and (5.59) to calculate the set of indices n_m and j_m that corresponds to the selected indices $\{q_m\}$. The block **inverse transformation** implements (5.60) in order to obtain the final switching vector sequence $\{\mathbf{v}_{sm}\}$. The switching times $\{t_m\}$ are obtained by means of the block **selection** which implements (5.61). Finally, the algorithm was described in VHDL by following the block diagram of Simulink. An extra block was added to translate the switching vectors into the gate signals for triggering the five-level cascaded full-bridge inverter. Table 5.2 shows the summary of resources used by the implementation. It is important to remark that block random access memories (BRAMs) and multipliers available in the FPGA were not used because algorithm does not need data storage or multiplication operations. Even though the implementation of the SVPWM algorithm with joint-phase switching state redundancy involves more blocks than the SVPWM algorithm without redundancy, the total equivalent gate count is only a little higher. That is because in the new algorithm most of the operations involve 4D vectors which require less resources than the five-dimensional (5D) vectors used by its counterpart.

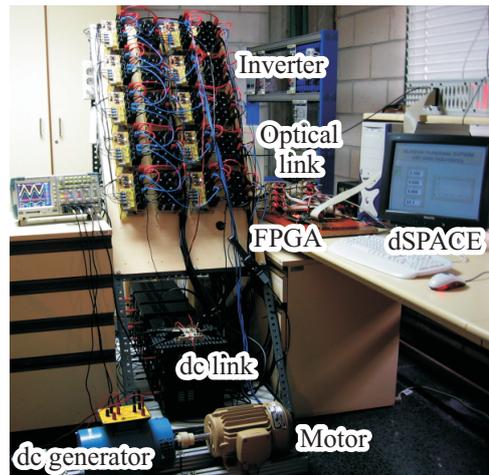
5.4 Experimental Results

The SVPWM algorithm was tested by using a laboratory prototype. Figure 5.7 shows the block diagram and a photograph of the experimental setup that includes the FPGA, a dSPACE platform, the inverter and the load. The DSPACE DS1103 PPC Controller Board provides the reference vectors to the FPGA. The trigger signals generated by the FPGA control the transistors of the multilevel inverter. The five-level five-phase cascaded full-bridge inverter shown in Figure 5.5, with 3125 different switching states and 2101 space vectors, was used in the experiments. The dc source voltage of each full-bridge cell is 60 V. Therefore the inverter voltage step V_{dc} is 60 V as well. The load is a five-phase distributed-concentrated winding induction motor of four poles with isolated neutral. This motor was specifically built for the tests by rewinding the stator phases on the thirty stator slots of a 1.1 kW three-phase motor.

Experimental measurements are compared with simulation results in Figure 5.8 when a 50 Hz balanced sinusoidal reference is considered. The simulation model of the experimen-



(a) Diagram



(b) Photograph

Figure 5.7: Experimental test setup.

tal setup was done in Simulink including the algorithm implementation previously shown in Figure 5.6. A low switching frequency (2 kHz) was selected to make the comparison easier. There is a good agreement between the simulation model and the experimental setup, except for some lost switching pulses due to the dead time included in trigger signals which was not considered in the simulation model.

The performance of the proposed SVPWM algorithm was tested at high and at low modulation indices. In the first case the maximum modulation index was used ($m = 2.102$) and in the second one a modulation index of 0.8 was considered. In both tests, the voltage reference is a 50 Hz balanced sinusoidal wave and the switching frequency is 10 kHz. Figure 5.9 shows the inverter leg voltage V_s^a and the low-order voltage harmonics. The filtered leg voltage has a fundamental component that is set by the voltage reference and a fifth harmonic that depends on the selected indices q . This fifth harmonic, which is not present in the SVPWM without redundancy of previous chapter, is what permits the new algorithm to extend the modulation index beyond $m = 2$ up to $m = 2.102$. Voltages have been measured in the points displayed in Figure 5.5 taking the neutral voltage of the inverter as reference. Figure 5.10 shows the load voltage and load current together with the low-order voltage harmonics. The channel one of the oscilloscope shows the line-to-line voltage $V_s^a - V_s^b$, the channel two shows the same waveform after been filtered, the channel three is the filtered voltage of the neutral of the load V_n and the channel four

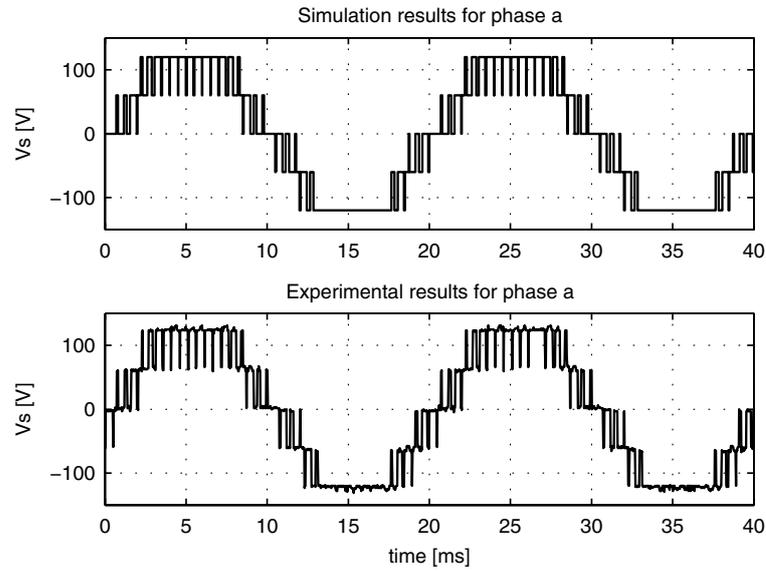
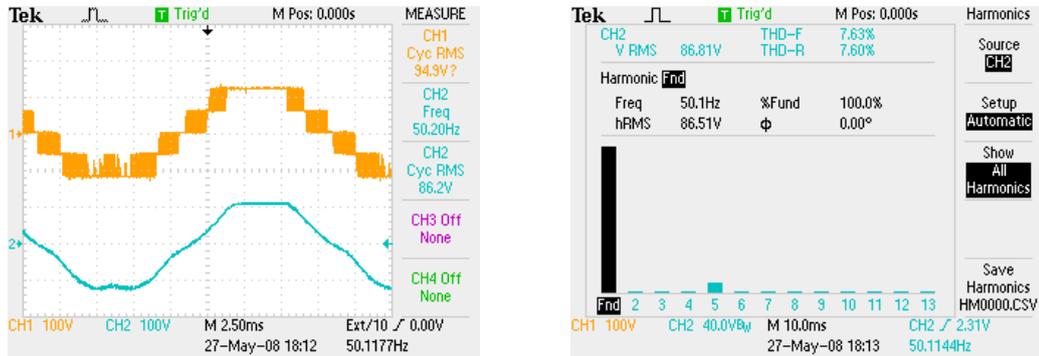
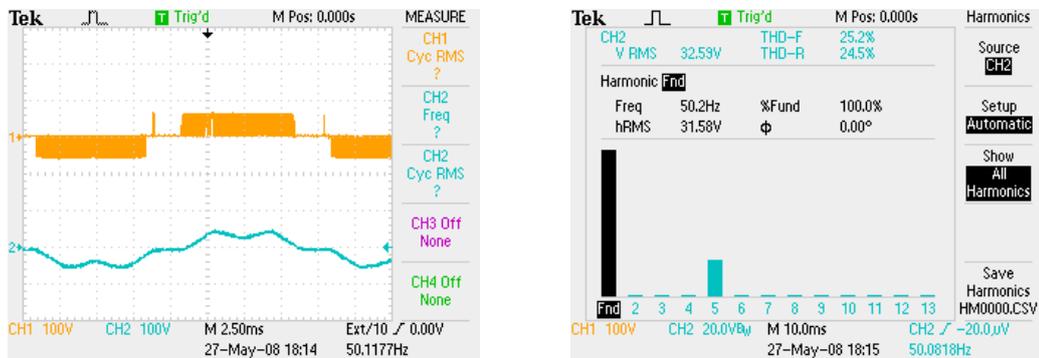


Figure 5.8: Comparison of experimental measurements with simulation results.



(a) $m = 2.102$



(b) $m = 0.8$

Figure 5.9: Experimental results. Ch1: leg voltage V_s^a ; Ch2: filtered leg voltage V_s^a .

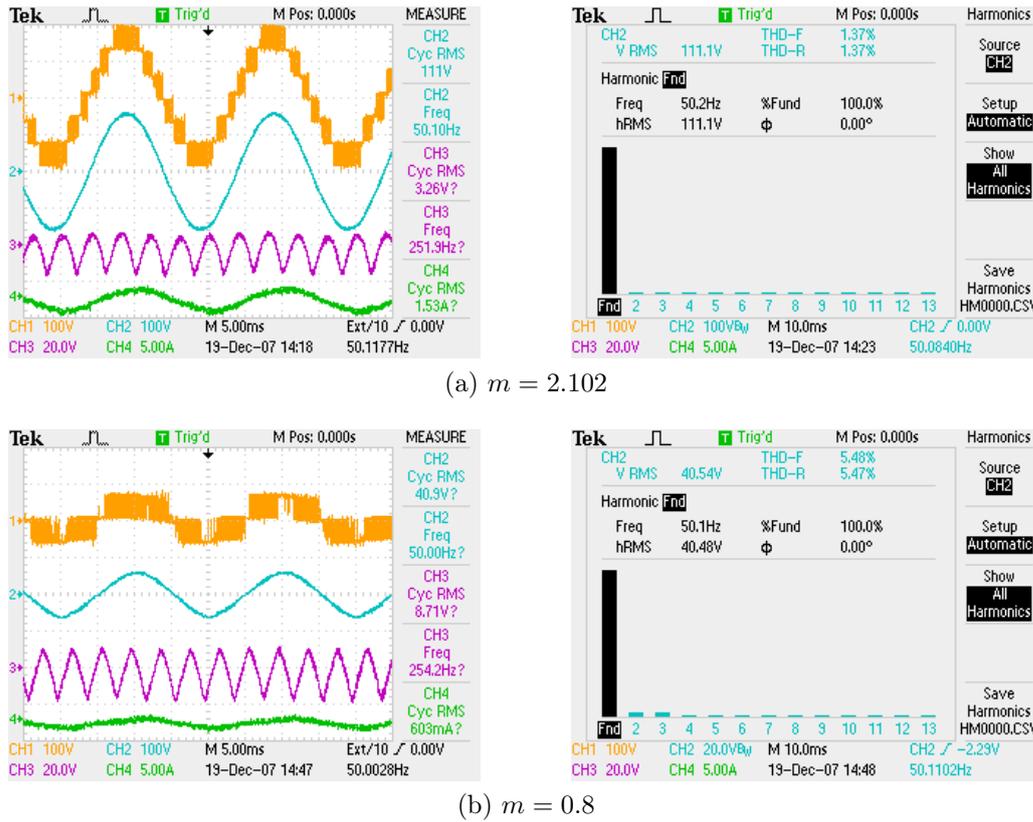


Figure 5.10: Experimental results. Ch1: line-to-line voltage $V_s^a - V_s^b$; Ch2: filtered line-to-line voltage $V_s^a - V_s^b$; Ch3: filtered neutral voltage V^n ; Ch4: phase current I_s^a .

shows the phase current I_s^a . At a high modulation index, the line-to-line voltage is a nine-level waveform because the modulation algorithm takes advantage of all five levels of the inverter. In this case, the low-order harmonics are negligible and the total harmonic distortion (THD) is 1.4%. In the line-to-line waveform there is not the fifth harmonic present (the same applies to the phase voltage) because it is a common mode signal that appears as a fluctuation in the neutral voltage. If the modulation index is low, then the low-order harmonics of the line-to-line voltage increase because the modulation algorithm uses fewer levels. In this case the line-to-line voltage has only three different levels and the THD increases up to 5.5%.

Figure 5.11 shows the trajectories of the inverter output voltage \mathbf{V}_s and the load current vectors in stationary dq frames [98] in the case of $m = 2.102$. Both vectors move, at constant speed, along a circular trajectory in the $dq1$ plane. There is no third harmonic present in the output voltage; hence, as expected, vectors in the $dq3$ plane stay close to the origin. Trajectories of the new SVPWM algorithm in the stationary frame are similar to the trajectories of the SVPWM without redundancy of previous chapter.

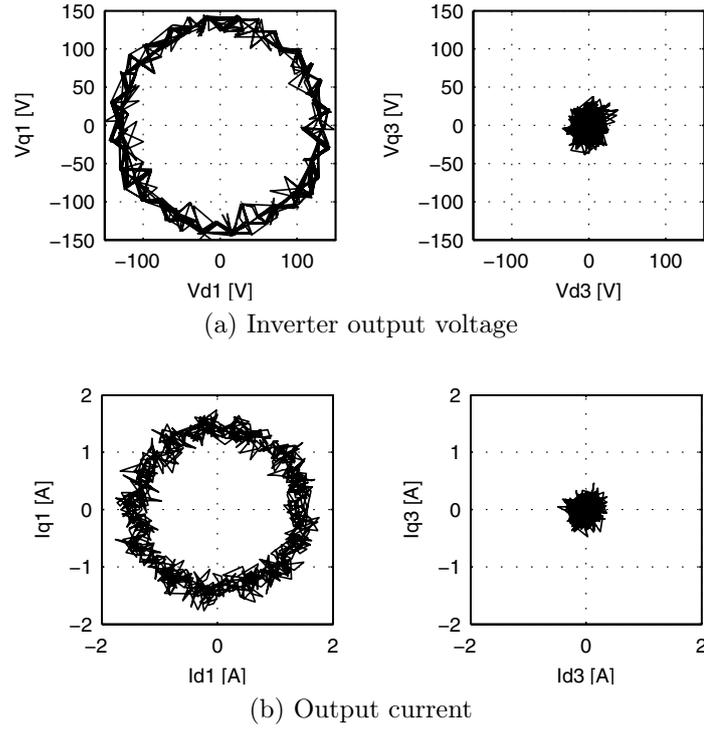


Figure 5.11: Trajectories of the output voltage and the current vectors in stationary dq axes.

5.5 Application to Three-Phase Three-Leg Converters

5.5.1 2D SVPWM Algorithm With State Redundancy

The modulation algorithm for three-phase systems is obtained by making $P = 3$ in the general multiphase algorithm. In this case, the normalized reference vector $\mathbf{v}_r = [v_r^a, v_r^b, v_r^c]^T$ can be calculated by means of (4.48), and its corresponding reference space vector $\boldsymbol{\omega}_r = [\omega_r^a, \omega_r^b]^T$ is a two-dimensional (2D) vector that can be calculated from (5.25) as

$$\boldsymbol{\omega}_r = \mathbf{T}_\omega \mathbf{v}_r \in \mathbb{R}^2 \quad (5.75)$$

where the transformation matrix \mathbf{T}_ω is

$$\mathbf{T}_\omega = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix}. \quad (5.76)$$

Hence, the components of the reference space vector can be easily calculated from the reference vectors as

$$\begin{aligned} \omega_r^a &= v_r^a - v_r^c \\ \omega_r^b &= v_r^b - v_r^c \end{aligned} \quad (5.77)$$

In accordance with (5.28) and (5.29), the integer and fractional parts of the reference

TABLE 5.3
PERMUTATION MATRIX IN THE 2D SVPWM ALGORITHM.

| C_{ab} | Ordered vector $\hat{\omega}_f$ | Matrix \mathbf{P} |
|----------|--|---|
| 0 | $\hat{\omega}_f^a = \omega_f^b$ $\hat{\omega}_f^b = \omega_f^a$ | $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ |
| 1 | $\hat{\omega}_f^a = \omega_f^a$ $\hat{\omega}_f^b = \omega_f^b$ | $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ |

space vector are

$$\boldsymbol{\omega}_i = \text{integ}(\boldsymbol{\omega}_r) = [\omega_i^a, \omega_i^b]^T \in \mathbb{Z}^2 \quad (5.78)$$

$$\boldsymbol{\omega}_f = \boldsymbol{\omega}_r - \boldsymbol{\omega}_i = [\omega_f^a, \omega_f^b]^T \in \mathbb{R}^2. \quad (5.79)$$

The displaced space vectors $\boldsymbol{\omega}_{d1} = [\omega_{d1}^a, \omega_{d1}^b]^T$, $\boldsymbol{\omega}_{d2} = [\omega_{d2}^a, \omega_{d2}^b]^T$ and $\boldsymbol{\omega}_{d3} = [\omega_{d3}^a, \omega_{d3}^b]^T$ that approximate the fractional part of the reference space vector $\boldsymbol{\omega}_f$ are obtained by means of a two-level 2D SVPWM algorithm. The application of the algorithm developed in §4.2.3 requires calculating the permutation matrix \mathbf{P} that sorts the components of vector $\boldsymbol{\omega}_f$. Such matrix can be easily determined by testing the following logical condition:

$$C_{ab} = [\omega_f^a \geq \omega_f^b]. \quad (5.80)$$

Relationships between the result of this condition and the permutation matrix \mathbf{P} are shown in Table 5.3. Next, the matrix \mathbf{D} is calculated from the upper triangular matrix $\hat{\mathbf{D}}$ in (4.30) by means of (4.32) as

$$\mathbf{D} = \mathbf{P}^T \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}. \quad (5.81)$$

The sequence of displaced space vectors $\{\boldsymbol{\omega}_{d1}, \boldsymbol{\omega}_{d2}, \boldsymbol{\omega}_{d3}\}$ is extracted from the matrix \mathbf{D} taking into account the expression in (4.22):

$$\mathbf{D} = \begin{bmatrix} 1 & 1 & 1 \\ \omega_{d1}^a & \omega_{d2}^a & \omega_{d3}^a \\ \omega_{d1}^b & \omega_{d2}^b & \omega_{d3}^b \end{bmatrix} \quad (5.82)$$

The dwell time corresponding to each displaced switching vector is calculated directly from the components of $\hat{\boldsymbol{\omega}}_f$ by means of (4.33) as

$$\begin{aligned} \tau_1 &= 1 - \hat{\omega}_f^a \\ \tau_2 &= \hat{\omega}_f^a - \hat{\omega}_f^b \\ \tau_3 &= \hat{\omega}_f^b. \end{aligned} \quad (5.83)$$

TABLE 5.4

DISPLACED SPACE VECTOR SEQUENCE AND DWELL TIMES OF THE 2D SVPWM ALGORITHM.

| C_{ab} | Displaced space vector sequence ω_{sj} | Dwell times τ_j |
|----------|---|------------------------------------|
| 0 | $\omega_{d1} = [0, 0]^T$ | $\tau_1 = 1 - \omega_f^b$ |
| | $\omega_{d2} = [0, 1]^T$ | $\tau_2 = \omega_f^b - \omega_f^a$ |
| | $\omega_{d3} = [1, 1]^T$ | $\tau_3 = \omega_f^a$ |
| 1 | $\omega_{d1} = [0, 0]^T$ | $\tau_1 = 1 - \omega_f^a$ |
| | $\omega_{d2} = [1, 0]^T$ | $\tau_2 = \omega_f^a - \omega_f^b$ |
| | $\omega_{d3} = [1, 1]^T$ | $\tau_3 = \omega_f^b$ |

If the two-level modulation problem is solved in the two cases in Table 5.3 then the results shown in Table 5.4 are obtained.

The value of q_i is obtained from the integer part of the reference space vector ω_i by means of expression in (5.43) as

$$q_i = \omega_i^a + \omega_i^b. \quad (5.84)$$

The intervals $[q_{\min}^a, q_{\max}^a]$, $[q_{\min}^b, q_{\max}^b]$ and $[q_{\min}^c, q_{\max}^c]$ that correspond to every phase are calculated by means of (5.50) and (5.51) as

$$\begin{aligned} [q_{\min}^a, q_{\max}^a] &= [\Delta q^a + 3N_{\min}^a, \Delta q^a + 3N_{\max}^a + 2] \\ [q_{\min}^b, q_{\max}^b] &= [\Delta q^b + 3N_{\min}^b, \Delta q^b + 3N_{\max}^b + 2] \\ [q_{\min}^c, q_{\max}^c] &= [\Delta q^c + 3N_{\min}^c, \Delta q^c + 3N_{\max}^c + 2] \end{aligned} \quad (5.85)$$

where from (5.52) and taking into account the particular values of ω_{d1} and ω_{d3} in Table 5.4 one has

$$\begin{aligned} \Delta q^a &= q_i - 3\omega_i^a - \omega_{d2}^a - 1 \\ \Delta q^b &= q_i - 3\omega_i^b - \omega_{d2}^b - 1 \\ \Delta q^c &= q_i. \end{aligned} \quad (5.86)$$

The bounding indices q_{\min} and q_{\max} are determined by using expressions in (5.53) and (5.54), respectively, as

$$q_{\min} = \max(q_{\min}^a, q_{\min}^b, q_{\min}^c) \quad (5.87)$$

$$q_{\max} = \min(q_{\max}^a, q_{\max}^b, q_{\max}^c). \quad (5.88)$$

By means of the condition given in (5.56) it can be tested if the reference does not lie in the overmodulation region as

$$q_{\max} - q_{\min} < 2. \quad (5.89)$$

The next step is to choose three consecutive integer numbers $\{q_1, q_2, q_3\}$ within the interval $[q_{\max}, q_{\min}]$ according to any desired modulation strategy. The values of n_1 , n_2 and n_3 that

correspond to selected indices are calculated from (5.58) as

$$\begin{aligned} n_1 &= \text{integ} \left(\frac{q_1 - q_i}{3} \right) \\ n_2 &= \text{integ} \left(\frac{q_2 - q_i}{3} \right) \\ n_3 &= \text{integ} \left(\frac{q_3 - q_i}{3} \right) \end{aligned} \quad (5.90)$$

and the values of j_1 , j_2 and j_3 by means of (5.59) as

$$\begin{aligned} j_1 &= q_1 - q_i - 3n_1 + 1 \\ j_2 &= q_2 - q_i - 3n_2 + 1 \\ j_3 &= q_3 - q_i - 3n_3 + 1. \end{aligned} \quad (5.91)$$

Finally, the vectors of the switching vector are calculated from (5.60) as

$$\begin{aligned} \mathbf{v}_{s1} &= [\omega_i^a + \omega_{d_{j_1}}^a + n_1, \omega_i^b + \omega_{d_{j_1}}^b + n_1, n_1]^T \\ \mathbf{v}_{s2} &= [\omega_i^a + \omega_{d_{j_2}}^a + n_2, \omega_i^b + \omega_{d_{j_2}}^b + n_2, n_2]^T \\ \mathbf{v}_{s3} &= [\omega_i^a + \omega_{d_{j_3}}^a + n_3, \omega_i^b + \omega_{d_{j_3}}^b + n_3, n_3]^T \end{aligned} \quad (5.92)$$

and their corresponding switching times by means of (5.61) as

$$\begin{aligned} t_1 &= \tau_{j_1} \\ t_2 &= \tau_{j_2} \\ t_3 &= \tau_{j_3}. \end{aligned} \quad (5.93)$$

5.5.2 Example

Let us consider the same example as in section §4.5.2 where the normalized reference vector is $\mathbf{v}_r = [0.59, -1.86, 1.27]^T$. In this case, the reference space vector, calculated by means of (5.75), is the following 2D vector:

$$\boldsymbol{\omega}_r = \mathbf{T}_\omega \mathbf{v}_r = [-0.68, -3.13]^T \quad (5.94)$$

In accordance with (5.78) and (5.79), the integer and fractional parts of the reference space vector are

$$\boldsymbol{\omega}_i = \text{integ}(\boldsymbol{\omega}_r) = [-1, -4]^T \quad (5.95)$$

$$\boldsymbol{\omega}_f = \boldsymbol{\omega}_r - \boldsymbol{\omega}_i = [0.32, 0.87]^T. \quad (5.96)$$

After evaluating the condition in (5.80)

$$C_{ab} = [\omega_f^a \geq \omega_f^b] = 0. \quad (5.97)$$

Table 5.4 provides the following displaced space vector sequence:

$$\begin{aligned} \boldsymbol{\omega}_{d1} &= [0, 0]^T \\ \boldsymbol{\omega}_{d2} &= [0, 1]^T \\ \boldsymbol{\omega}_{d3} &= [1, 1]^T \end{aligned} \quad (5.98)$$

and the corresponding dwell times:

$$\begin{aligned}\tau_1 &= 1 - \omega_f^b = 0.13 \\ \tau_2 &= \omega_f^b - \omega_f^a = 0.55 \\ \tau_3 &= \omega_f^a = 0.32.\end{aligned}\tag{5.99}$$

The value of the index q_i that corresponds to the integer part of the reference space vector can be calculated by means of (5.84) as

$$q_i = \omega_i^a + \omega_i^b = -5.\tag{5.100}$$

If a five-level cascaded full-bridge inverter is considered then

$$[N_{\min}^a, N_{\max}^a] = [N_{\min}^b, N_{\max}^b] = [N_{\min}^c, N_{\max}^c] = [-2, 2].\tag{5.101}$$

In this case, the intervals of indices that correspond to the switching vectors in which the converter is able to generate every phase voltage are calculated by means of (5.85) as

$$\begin{aligned}[q_{\min}^a, q_{\max}^a] &= [-9, 5] \\ [q_{\min}^b, q_{\max}^b] &= [-1, 13] \\ [q_{\min}^c, q_{\max}^c] &= [-11, 3].\end{aligned}\tag{5.102}$$

Consequently, from (5.87) and (5.88), the range of the indices that correspond to useful switching vectors is

$$[q_{\min}, q_{\max}] = [-1, 3].\tag{5.103}$$

The length of this interval is greater than three, therefore the reference vector is not in the overmodulation region and it can be accurately synthesized. Among all available possibilities, the following three first indices within the range have been selected:

$$\{q_1, q_2, q_3\} = \{-1, 0, 1\}.\tag{5.104}$$

The values of n_m and j_m that correspond to these q_m indices can be calculated by means of (5.90) and (5.91) as

$$\begin{aligned}n_1 &= 1 & n_2 &= 1 & n_3 &= 2 \\ j_1 &= 2 & j_2 &= 3 & j_3 &= 1.\end{aligned}\tag{5.105}$$

Finally, the following switching vector sequence

$$\begin{aligned}\mathbf{v}_{s1} &= \mathbf{v}_s(q_1) = \mathbf{v}_s(n_1, j_1) = [0, -2, 1]^T \\ \mathbf{v}_{s2} &= \mathbf{v}_s(q_2) = \mathbf{v}_s(n_2, j_2) = [1, -2, 1]^T \\ \mathbf{v}_{s3} &= \mathbf{v}_s(q_3) = \mathbf{v}_s(n_3, j_3) = [1, -2, 2]^T.\end{aligned}\tag{5.106}$$

together with their corresponding switching times

$$\begin{aligned}t_1 &= \tau_2 = 0.55 \\ t_2 &= \tau_3 = 0.32 \\ t_3 &= \tau_1 = 0.13.\end{aligned}\tag{5.107}$$

are obtained by means of (5.92) and (5.93), respectively.

5.5.3 Comparison With Existing 2D Algorithm

The fast multilevel SVPWM algorithm for multilevel three-phase converters presented by Celanovic and Boroyevich in [72] is a 2D modulation algorithm which uses the transformation already given with (3.4) and repeated here for convenience

$$\begin{bmatrix} v^g \\ v^h \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \end{bmatrix} \begin{bmatrix} v^a - v^b \\ v^b - v^c \\ v^c - v^a \end{bmatrix}. \quad (5.108)$$

to represent the reference vector and the switching vectors in the gh space. The above expression can be rewritten as

$$\begin{bmatrix} v^g \\ v^h \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} v^a \\ v^b \\ v^c \end{bmatrix} \quad (5.109)$$

which is very similar to transformation in (5.75) used by the new 2D SVPWM algorithm. Once in the gh space, the fast multilevel SVPWM algorithm searches the three nearest space vectors to the reference vector. This is done by means of a comparison operation following a similar procedure as in (5.97). Dwell times corresponding to the three nearest space vector are obtained from the fractional parts of the reference space vector as in (5.99). Consequently, both algorithms have a similar computational cost to calculate space vectors and dwell times.

The application of the fast multilevel SVPWM algorithm to the reference vector $\mathbf{v}_r = [0.59, -1.86, 1.27]^T$, considered in the example of §5.5.2 provides the following results:

$$\begin{aligned} [v_{s1}^g, v_{s1}^h]^T &= [2, -3]^T \rightarrow \tau_1 = 0.13 \\ [v_{s2}^g, v_{s2}^h]^T &= [3, -4]^T \rightarrow \tau_3 = 0.32 \\ [v_{s3}^g, v_{s3}^h]^T &= [3, -3]^T \rightarrow \tau_2 = 0.55 \end{aligned} \quad (5.110)$$

If the vectors of the switching sequence in (5.106), which have been obtained in that example, are transformed by means of (5.108) then the following space vectors in the gh frame are obtained

$$\begin{aligned} \mathbf{v}_{s1} : [v_{s1}^g, v_{s1}^h]^T &= [2, -3]^T \rightarrow \tau_1 = 0.13 \\ \mathbf{v}_{s2} : [v_{s2}^g, v_{s2}^h]^T &= [3, -3]^T \rightarrow \tau_2 = 0.55, \\ \mathbf{v}_{s3} : [v_{s3}^g, v_{s3}^h]^T &= [3, -3]^T \rightarrow \tau_3 = 0.32 \end{aligned} \quad (5.111)$$

which are the same as the previous vectors, except for the way in which they are sorted. Figure 5.12 shows that if space vectors of the fast multilevel SVPWM algorithm are properly sorted then the result obtained with both modulation techniques in the gh frame is the same.

As it was shown in section §3.2, the fast multilevel SVPWM algorithm provides an unsorted space vector sequence and it does not include an efficient method to obtain the switching vector sequence from space vectors. Consequently, the new 2D SVPWM algorithm can be considered as an extension of the fast algorithm that includes such feature.

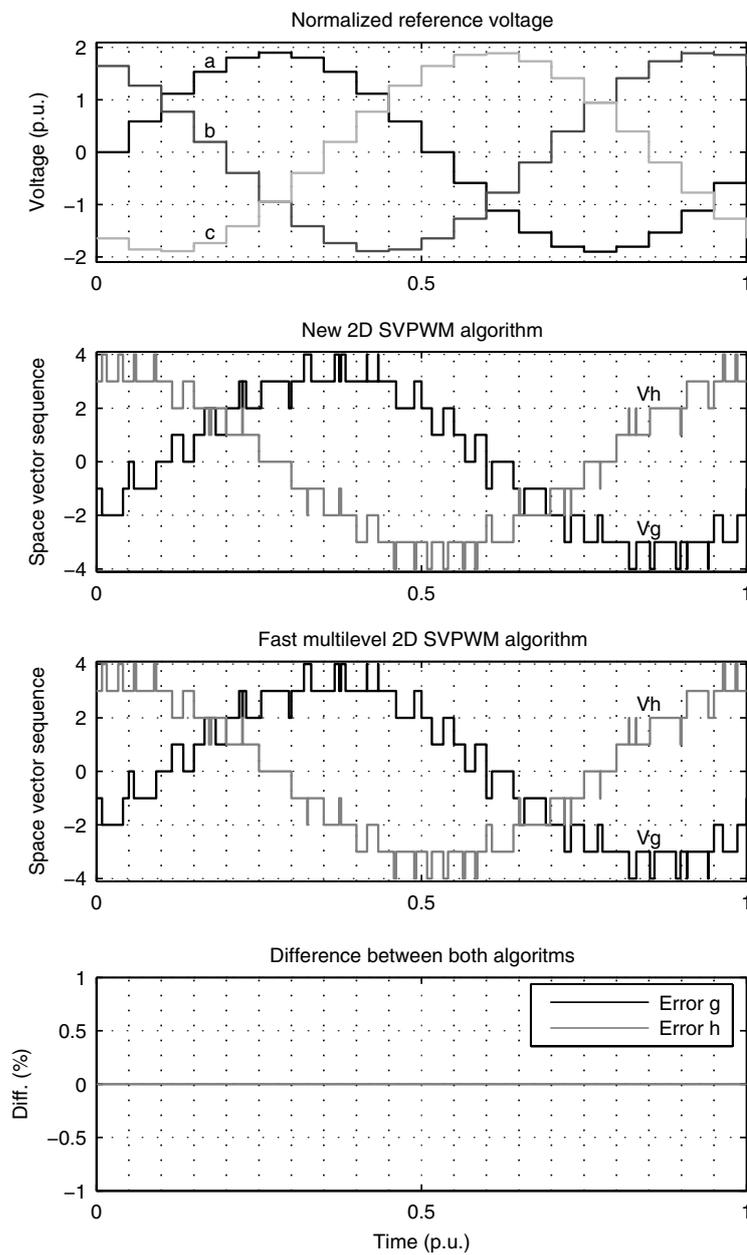
Figure 5.12: 2D algorithms comparison in the gh space.

TABLE 5.5
RESOURCES SUMMARY.

| | | | |
|---|--------------------|------|--|
| Target Device | : xc3s200 | | |
| Number of Slice Flip Flops: | 2,057 out of 3,840 | 53% | |
| Number of 4 input LUTs: | 2,439 out of 3,840 | 63% | |
| Number of occupied Slices: | 1,563 out of 1,920 | 81% | |
| Total Number 4 input LUTs: | 2,231 out of 3,840 | 58% | |
| Number of bonded IOBs: | 63 out of 173 | 36% | |
| IOB Flip Flops: | 39 | | |
| Number of Block RAMs: | 0 out of 12 | 0% | |
| Number of MULT18X18s: | 0 out of 12 | 0% | |
| Number of GCLKs: | 8 out of 8 | 100% | |
| Number of Startups: | 1 out of 1 | 100% | |
| Total equivalent gate count for design: | 31,063 | | |

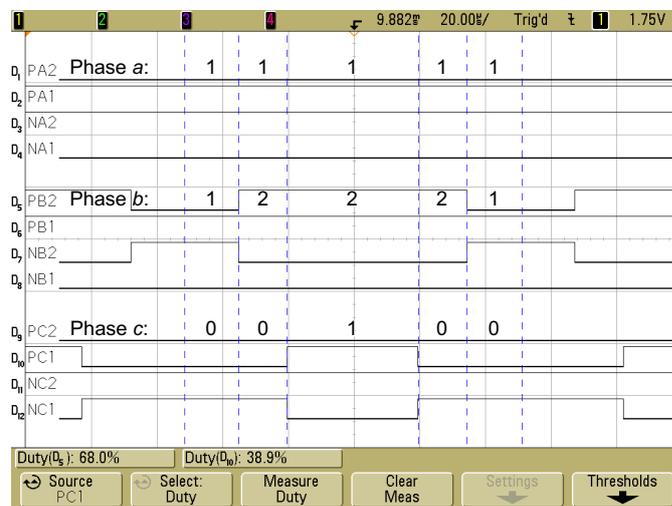


Figure 5.13: Trigger signals.

5.5.4 Experimental Results

The SVPWM algorithm was described for a three-level inverter by using VHDL and it was implemented in a Digilent S3 board. Table 5.5 shows a summary of the resources used by the implementation. Neither BRAMs or the multipliers available in the FPGA were used because the algorithm does not need data storage or multiplication operations.

Figure 5.13 shows the FPGA output waveforms that correspond to the trigger signals in the case $\mathbf{v}_r = [-0.16, 0.52, -0.77]^T$. The new algorithm was tested with a three-level neutral-point clamped (NPC) inverter driving a star connected induction motor. The experimental setup is the same as used in §4.5.4 so the measurement points and the reference voltage are the same displayed in Figure 3.2. A reference voltage with 50 Hz fundamental frequency and a 10 kHz output switching frequency were considered. Figure 5.14 shows the inverter output leg voltage, and the low-order voltage harmonics, in the case of balanced sinusoidal reference with modulation index $m = 1.15$. The SVPWM algorithm injects an homopolar component in the leg voltage (channels one and two of the oscilloscope) as Figure 5.14a shows. This homopolar component is the voltage of the neutral point of the load V_n that is shown in channel three. The homopolar component

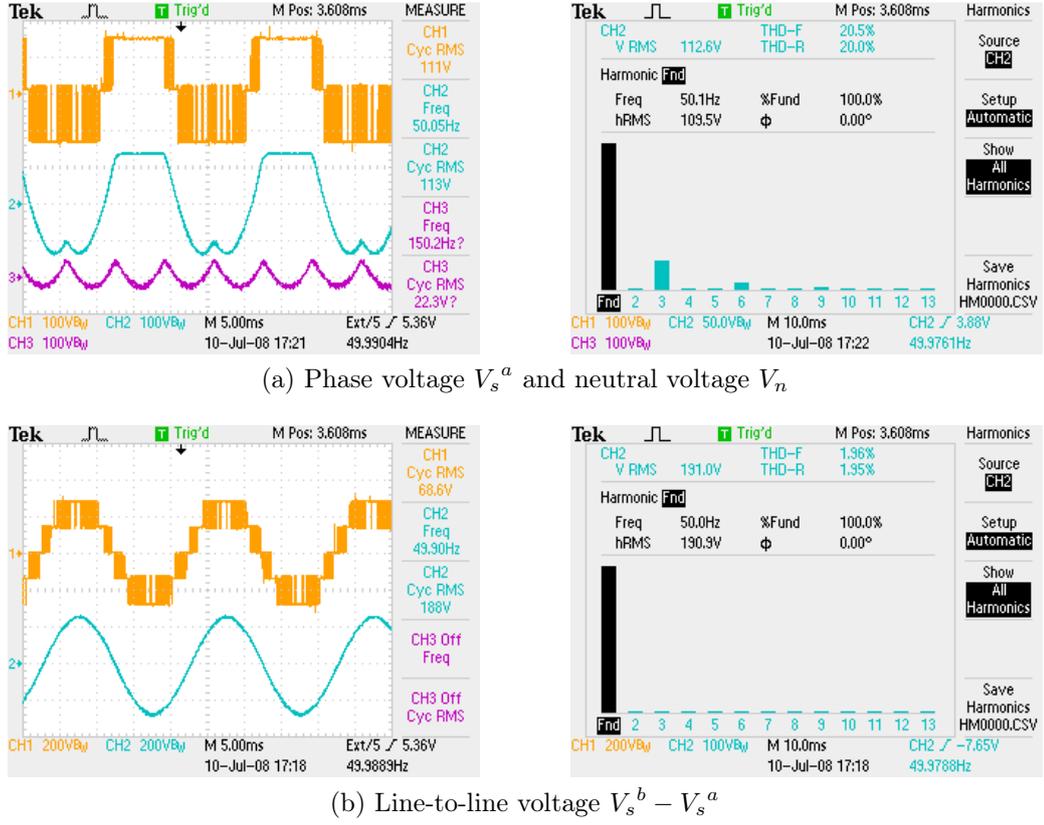


Figure 5.14: Inverter output voltage. Ch1: switched output voltage; Ch2: filtered output voltage; Ch3: filtered neutral voltage.

has no influence on the line-to-line voltage shown in Figure 5.14b. It is a sinusoidal wave with negligible low order harmonics and a very low THD.

5.6 Application to Three-Phase Four-Leg Converters

5.6.1 3D SVPWM Algorithm With State Redundancy

Section §4.6 has shown that switching vectors of four-leg converters include the switching states of each leg. Therefore, the modulation algorithm for four-leg converters can be obtained by making $P = 4$ in the multiphase algorithm. In this case, the normalized reference vector $\mathbf{v}_r = [v_r^a, v_r^b, v_r^c, v_r^n]^T$ can be obtained by means of (4.67), and its corresponding reference space vector $\boldsymbol{\omega}_r = [\omega_r^a, \omega_r^b, \omega_r^c]^T$ is a three-dimensional (3D) vector that can be calculated from (5.25) as

$$\boldsymbol{\omega}_r = \mathbf{T}_\omega \mathbf{v}_r \in \mathbb{R}^3 \quad (5.112)$$

where the transformation matrix \mathbf{T}_ω is

$$\mathbf{T}_\omega = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}. \quad (5.113)$$

Hence, the components of the reference space vector can be easily calculated from the reference vector as

$$\begin{aligned}\omega_r^a &= v_r^a - v_r^n \\ \omega_r^b &= v_r^b - v_r^n \\ \omega_r^c &= v_r^c - v_r^n.\end{aligned}\tag{5.114}$$

In accordance with (5.28) and (5.29), the integer and fractional parts of the reference space vector are

$$\boldsymbol{\omega}_i = \text{integ}(\boldsymbol{\omega}_r) = [\omega_i^a, \omega_i^b, \omega_i^c]^T \in \mathbb{Z}^3\tag{5.115}$$

$$\boldsymbol{\omega}_f = \boldsymbol{\omega}_r - \boldsymbol{\omega}_i = [\omega_f^a, \omega_f^b, \omega_f^c]^T \in \mathbb{R}^3.\tag{5.116}$$

The sequence of displaced space vectors, $\boldsymbol{\omega}_{d1} = [\omega_{d1}^a, \omega_{d1}^b, \omega_{d1}^c]^T$, $\boldsymbol{\omega}_{d2} = [\omega_{d2}^a, \omega_{d2}^b, \omega_{d2}^c]^T$, $\boldsymbol{\omega}_{d3} = [\omega_{d3}^a, \omega_{d3}^b, \omega_{d3}^c]^T$ and $\boldsymbol{\omega}_{d4} = [\omega_{d4}^a, \omega_{d4}^b, \omega_{d4}^c]^T$, that approximate the fractional part of the reference space vector $\boldsymbol{\omega}_f$ must be obtained by means of a two-level 3D SVPWM algorithm without redundancy. If the 3D SVPWM algorithm developed in §4.5.1 is used to calculate the displaced space vector sequence and their dwell times then the Table 5.6 is obtained, in which

$$\begin{aligned}C_{ab} &= [\omega_f^a \geq \omega_f^b] \\ C_{bc} &= [\omega_f^b \geq \omega_f^c] \\ C_{ca} &= [\omega_f^c \geq \omega_f^a].\end{aligned}\tag{5.117}$$

The value of q_i is obtained from the integer part of the reference space vector by means of expression in (5.43) as

$$q_i = \omega_i^a + \omega_i^b + \omega_i^c.\tag{5.118}$$

The intervals $[q_{\min}^a, q_{\max}^a]$, $[q_{\min}^b, q_{\max}^b]$, $[q_{\min}^c, q_{\max}^c]$ and $[q_{\min}^n, q_{\max}^n]$ that correspond to every phase are calculated by means of (5.50) and (5.51) as

$$\begin{aligned}[q_{\min}^a, q_{\max}^a] &= [\Delta q^a + 3N_{\min}^a, \Delta q^a + 3N_{\max}^a + 2] \\ [q_{\min}^b, q_{\max}^b] &= [\Delta q^b + 3N_{\min}^b, \Delta q^b + 3N_{\max}^b + 2] \\ [q_{\min}^c, q_{\max}^c] &= [\Delta q^c + 3N_{\min}^c, \Delta q^c + 3N_{\max}^c + 2] \\ [q_{\min}^n, q_{\max}^n] &= [\Delta q^n + 3N_{\min}^n, \Delta q^n + 3N_{\max}^n + 2]\end{aligned}\tag{5.119}$$

where from (5.52) and taking into account the particular values of $\boldsymbol{\omega}_{d1}$ and $\boldsymbol{\omega}_{d4}$ in Table 5.6 then

$$\begin{aligned}\Delta q^a &= q_i - 4\omega_i^a - \omega_{d2}^a - \omega_{d3}^a - 1 \\ \Delta q^b &= q_i - 4\omega_i^b - \omega_{d2}^b - \omega_{d3}^b - 1 \\ \Delta q^c &= q_i - 4\omega_i^c - \omega_{d2}^c - \omega_{d3}^c - 1 \\ \Delta q^n &= q_i.\end{aligned}\tag{5.120}$$

The bounding indices q_{\min} and q_{\max} are determined by using expressions in (5.53) and (5.54), respectively, as

$$q_{\min} = \max(q_{\min}^a, q_{\min}^b, q_{\min}^c, q_{\min}^n)\tag{5.121}$$

$$q_{\max} = \min(q_{\max}^a, q_{\max}^b, q_{\max}^c, q_{\max}^n).\tag{5.122}$$

TABLE 5.6
DISPLACED SPACE VECTOR SEQUENCE AND DWELL TIMES OF THE 3D SVPWM ALGORITHM.

| C_{ab} | C_{bc} | C_{ca} | Displaced space vector sequence ω_{sj} | Dwell times τ_j |
|----------|----------|----------|---|------------------------------------|
| 0 | 0 | 1 | $\omega_{d1} = [0, 0, 0]^T$ | $\tau_1 = 1 - \omega_f^c$ |
| | | | $\omega_{d2} = [0, 0, 1]^T$ | $\tau_2 = \omega_f^c - \omega_f^b$ |
| | | | $\omega_{d3} = [0, 1, 1]^T$ | $\tau_3 = \omega_f^b - \omega_f^a$ |
| | | | $\omega_{d4} = [1, 1, 1]^T$ | $\tau_4 = \omega_f^a$ |
| 0 | 1 | 0 | $\omega_{d1} = [0, 0, 0]^T$ | $\tau_1 = 1 - \omega_f^b$ |
| | | | $\omega_{d2} = [0, 1, 0]^T$ | $\tau_2 = \omega_f^b - \omega_f^a$ |
| | | | $\omega_{d3} = [1, 1, 0]^T$ | $\tau_3 = \omega_f^a - \omega_f^c$ |
| | | | $\omega_{d4} = [1, 1, 1]^T$ | $\tau_4 = \omega_f^c$ |
| 0 | 1 | 1 | $\omega_{d1} = [0, 0, 0]^T$ | $\tau_1 = 1 - \omega_f^b$ |
| | | | $\omega_{d2} = [0, 1, 0]^T$ | $\tau_2 = \omega_f^b - \omega_f^c$ |
| | | | $\omega_{d3} = [0, 1, 1]^T$ | $\tau_3 = \omega_f^c - \omega_f^a$ |
| | | | $\omega_{d4} = [1, 1, 1]^T$ | $\tau_4 = \omega_f^a$ |
| 1 | 0 | 0 | $\omega_{d1} = [0, 0, 0]^T$ | $\tau_1 = 1 - \omega_f^a$ |
| | | | $\omega_{d2} = [1, 0, 0]^T$ | $\tau_2 = \omega_f^a - \omega_f^c$ |
| | | | $\omega_{d3} = [1, 0, 1]^T$ | $\tau_3 = \omega_f^c - \omega_f^b$ |
| | | | $\omega_{d4} = [1, 1, 1]^T$ | $\tau_4 = \omega_f^b$ |
| 1 | 1 | 0 | $\omega_{d1} = [0, 0, 0]^T$ | $\tau_1 = 1 - \omega_f^a$ |
| | | | $\omega_{d2} = [1, 0, 0]^T$ | $\tau_2 = \omega_f^a - \omega_f^b$ |
| | | | $\omega_{d3} = [1, 1, 0]^T$ | $\tau_3 = \omega_f^b - \omega_f^c$ |
| | | | $\omega_{d4} = [1, 1, 1]^T$ | $\tau_4 = \omega_f^c$ |
| 1 | 0 | 1 | $\omega_{d1} = [0, 0, 0]^T$ | $\tau_1 = 1 - \omega_f^c$ |
| | | | $\omega_{d2} = [0, 0, 1]^T$ | $\tau_2 = \omega_f^c - \omega_f^a$ |
| | | | $\omega_{d3} = [1, 0, 1]^T$ | $\tau_3 = \omega_f^a - \omega_f^b$ |
| | | | $\omega_{d4} = [1, 1, 1]^T$ | $\tau_4 = \omega_f^b$ |

By means of the condition given in (5.56) it can be tested if the reference does not lie in the overmodulation region by means of

$$q_{\max} - q_{\min} < 3. \quad (5.123)$$

The next step is to select four consecutive integer numbers $\{q_1, q_2, q_3, q_4\}$ within the interval $[q_{\max}, q_{\min}]$ according to any desired modulation strategy. The values of n_1, n_2, n_3 and n_4 that correspond to the previously selected indices are calculated from (5.58) as

$$\begin{aligned} n_1 &= \text{integ} \left(\frac{q_1 - q_i}{4} \right) \\ n_2 &= \text{integ} \left(\frac{q_2 - q_i}{4} \right) \\ n_3 &= \text{integ} \left(\frac{q_3 - q_i}{4} \right) \\ n_4 &= \text{integ} \left(\frac{q_4 - q_i}{4} \right) \end{aligned} \quad (5.124)$$

and the values of j_1, j_2, j_3 and j_4 by means of (5.59) as

$$\begin{aligned} j_1 &= q_1 - q_i - 4n_1 + 1 \\ j_2 &= q_2 - q_i - 4n_2 + 1 \\ j_3 &= q_3 - q_i - 4n_3 + 1 \\ j_4 &= q_4 - q_i - 4n_4 + 1. \end{aligned} \quad (5.125)$$

Finally, the vectors of the switching sequence are calculated from (5.60) as

$$\begin{aligned} \mathbf{v}_{s1} &= [\omega_i^a + \omega_{d_{j_1}}^a + n_1, \omega_i^b + \omega_{d_{j_1}}^b + n_1, \omega_i^c + \omega_{d_{j_1}}^c + n_1, n_1]^T \\ \mathbf{v}_{s2} &= [\omega_i^a + \omega_{d_{j_2}}^a + n_2, \omega_i^b + \omega_{d_{j_2}}^b + n_2, \omega_i^c + \omega_{d_{j_2}}^c + n_2, n_2]^T \\ \mathbf{v}_{s3} &= [\omega_i^a + \omega_{d_{j_3}}^a + n_3, \omega_i^b + \omega_{d_{j_3}}^b + n_3, \omega_i^c + \omega_{d_{j_3}}^c + n_3, n_3]^T \\ \mathbf{v}_{s4} &= [\omega_i^a + \omega_{d_{j_4}}^a + n_4, \omega_i^b + \omega_{d_{j_4}}^b + n_4, \omega_i^c + \omega_{d_{j_4}}^c + n_4, n_4]^T \end{aligned} \quad (5.126)$$

and their corresponding switching times by means of (5.61) as

$$\begin{aligned} t_1 &= \tau_{j_1} \\ t_2 &= \tau_{j_2} \\ t_3 &= \tau_{j_3} \\ t_4 &= \tau_{j_4}. \end{aligned} \quad (5.127)$$

5.6.2 Example

Let us consider the same example as in section §4.6.2 where the normalized reference vector is $\mathbf{v}_r = [1.39, -1.15, -0, 31, 1.12]^T$. In this case, the reference space vector, calculated by means of (5.112), is the following 3D vector:

$$\boldsymbol{\omega}_r = [-0.27, -2.16, -1.43]^T \quad (5.128)$$

In accordance with (5.115) and (5.116), the integer and fractional parts of the reference space vector are

$$\boldsymbol{\omega}_i = \text{integ}(\boldsymbol{\omega}_r) = [0, -3, -2]^T \quad (5.129)$$

$$\boldsymbol{\omega}_f = \boldsymbol{\omega}_r - \boldsymbol{\omega}_i = [0.27, 0.74, 0.57]^T. \quad (5.130)$$

If the conditions in (5.117) are calculated then the results obtained are

$$\begin{aligned} C_{ab} &= [\omega_f^a \geq \omega_f^b] = 0 \\ C_{bc} &= [\omega_f^b \geq \omega_f^c] = 1 \\ C_{ca} &= [\omega_f^c \geq \omega_f^a] = 1. \end{aligned} \quad (5.131)$$

The case 011 of Table 5.6 provides the following displaced space vector sequence:

$$\begin{aligned} \boldsymbol{\omega}_{d1} &= [0, 0, 0]^T \\ \boldsymbol{\omega}_{d2} &= [0, 1, 0]^T \\ \boldsymbol{\omega}_{d3} &= [0, 1, 1]^T \\ \boldsymbol{\omega}_{d4} &= [1, 1, 1]^T \end{aligned} \quad (5.132)$$

together with the following dwell times:

$$\begin{aligned} \tau_1 &= 1 - \omega_f^b = 0.26 \\ \tau_2 &= \omega_f^b - \omega_f^c = 0.17 \\ \tau_2 &= \omega_f^c - \omega_f^a = 0.30 \\ \tau_3 &= \omega_f^a = 0.27. \end{aligned} \quad (5.133)$$

The value of the index q_i that corresponds to the integer part of the reference space vector can be calculated by means of (5.118) as

$$q_i = \omega_i^a + \omega_i^b + \omega_i^c = -5. \quad (5.134)$$

If a five-level cascaded full-bridge inverter is considered then

$$[N_{\min}^a, N_{\max}^a] = [N_{\min}^b, N_{\max}^b] = [N_{\min}^c, N_{\max}^c] = [N_{\min}^n, N_{\max}^n] = [-2, 2]. \quad (5.135)$$

In this case, the intervals of indices that correspond to the switching vectors in which the converter is able to generate every phase are calculated by means of (5.119) as

$$\begin{aligned} [q_{\min}^a, q_{\max}^a] &= [-14, 5] \\ [q_{\min}^b, q_{\max}^b] &= [-4, 15] \\ [q_{\min}^c, q_{\max}^c] &= [-7, 12] \\ [q_{\min}^n, q_{\max}^n] &= [-13, 6]. \end{aligned} \quad (5.136)$$

Consequently, from (5.121) and (5.122), the range of the indices that correspond to useful switching vectors is

$$[q_{\min}, q_{\max}] = [-4, 5]. \quad (5.137)$$

The length of this interval is greater than four, therefore the reference vector is not in the overmodulation region and it can be accurately synthesized. Among all available possibilities, the following four last indices within the range have been selected:

$$\{q_1, q_2, q_3, q_4\} = \{2, 3, 4, 5\}. \quad (5.138)$$

The values of the n_m and j_m that correspond to the previous q_m indices can be calculated by means of (5.124) and (5.125)

$$\begin{aligned} n_1 = 1 \quad n_2 = 2 \quad n_3 = 2 \quad n_4 = 2 \\ j_1 = 4 \quad j_2 = 1 \quad j_3 = 2 \quad j_4 = 3. \end{aligned} \quad (5.139)$$

Finally, the following switching vector sequence

$$\begin{aligned} \mathbf{v}_{s1} &= \mathbf{v}_s(q_1) = \mathbf{v}_s(n_1, j_1) = [2, -1, 0, 1]^T \\ \mathbf{v}_{s2} &= \mathbf{v}_s(q_2) = \mathbf{v}_s(n_2, j_2) = [2, -1, 0, 2]^T \\ \mathbf{v}_{s3} &= \mathbf{v}_s(q_3) = \mathbf{v}_s(n_3, j_3) = [2, 0, 0, 2]^T \\ \mathbf{v}_{s4} &= \mathbf{v}_s(q_4) = \mathbf{v}_s(n_4, j_4) = [2, 0, 1, 2]^T. \end{aligned} \quad (5.140)$$

together with following switching times

$$\begin{aligned} t_1 = \tau_4 = 0.27 \\ t_2 = \tau_1 = 0.26 \\ t_2 = \tau_2 = 0.16 \\ t_3 = \tau_3 = 0.30. \end{aligned} \quad (5.141)$$

are obtained by means of (5.126) and (5.127), respectively.

5.6.3 Comparison With Existing 3D Algorithm

The 3D SVPWM algorithm for four-leg multilevel converters presented by Franquelo et al. in [89] uses the line-to-neutral values of the reference vector and the switching vectors to carry out the space vector representation of the system in a 3D space. This transformation is the same as the transformation in (5.112) that is used by the new 3D algorithm to calculate the space vectors.

The algorithm in [89] makes use of the 3D SVPWM generalized algorithm in [81] to find the four space vectors nearest to the reference vector, and to calculate their dwell times. In section §4.5.3, it was shown that the 3D SVPWM generalized algorithm is the same as the 3D SVPWM algorithm for three-phase converters without redundancy obtained in previous chapter. This algorithm is based on a two-level SVPWM algorithm which is also used by the new 3D SVPWM for four-leg converters. Therefore, the space vectors and dwell times obtained with both 3D SVPWM algorithms for four-leg converters are the same

$$\begin{aligned} [S_{an}^1, S_{bn}^1, S_{cn}^1]^T &\equiv \boldsymbol{\omega}_{s1} = \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d1} \rightarrow d_1 \equiv \tau_1 \\ [S_{an}^2, S_{bn}^2, S_{cn}^2]^T &\equiv \boldsymbol{\omega}_{s2} = \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d2} \rightarrow d_2 \equiv \tau_2 \\ [S_{an}^3, S_{bn}^3, S_{cn}^3]^T &\equiv \boldsymbol{\omega}_{s3} = \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d3} \rightarrow d_3 \equiv \tau_3 \\ [S_{an}^4, S_{bn}^4, S_{cn}^4]^T &\equiv \boldsymbol{\omega}_{s4} = \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d4} \rightarrow d_4 \equiv \tau_4. \end{aligned} \quad (5.142)$$

The application of the 3D SVPWM algorithm in [89] to the reference vector $\mathbf{v}_r = [1.39, -1.15, -0, 31, 1.12]^T$, which has been considered in the example of §5.5.2, provides the following results:

$$\begin{aligned} [S_{an}^1, S_{bn}^1, S_{cn}^1] &= [0, -3, -2]^T \rightarrow d_1 = 0.26 \\ [S_{an}^2, S_{bn}^2, S_{cn}^2] &= [0, -2, -2]^T \rightarrow d_2 = 0.16 \\ [S_{an}^3, S_{bn}^3, S_{cn}^3] &= [0, -2, -1]^T \rightarrow d_3 = 0.30 \\ [S_{an}^4, S_{bn}^4, S_{cn}^4] &= [1, -2, -1]^T \rightarrow d_4 = 0.27. \end{aligned} \quad (5.143)$$

This sequence of vectors is the same as the sequence of space vectors obtained adding the integer part of the reference space vector to the displaced space vector sequence

$$\begin{aligned} \boldsymbol{\omega}_{s1} &= \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d1} = [0, -3, -2]^T \rightarrow \tau_1 = 0.26 \\ \boldsymbol{\omega}_{s2} &= \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d2} = [0, -2, -2]^T \rightarrow \tau_2 = 0.16 \\ \boldsymbol{\omega}_{s3} &= \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d3} = [0, -2, -1]^T \rightarrow \tau_3 = 0.30 \\ \boldsymbol{\omega}_{s4} &= \boldsymbol{\omega}_i + \boldsymbol{\omega}_{d4} = [1, -2, -1]^T \rightarrow \tau_4 = 0.27. \end{aligned} \quad (5.144)$$

Figure 5.15 shows that the space vector sequences obtained with both algorithms are identical.

The 3D SVPWM algorithm presented in [89] does not address the redundant switching state selection from space vectors. Therefore, the new 3D SVPWM algorithm for four-leg converters can be considered as an extension of such an algorithm where this feature is added.

5.6.4 Experimental Results

The algorithm was tested by simulation and in laboratory. In both cases, the considered conditions are a balanced voltage reference, where $v_r^a = 1.9 \sin(\omega t)$, $v_r^b = 1.9 \sin(\omega t + 2\pi/3)$ and $v_r^c = 1.9 \sin(\omega t - 2\pi/3)$, with a third-harmonic zero sequence, $v_r^n = 1.5 \sin(3\omega t + \pi)$. The fundamental frequency is 50 Hz and the switching frequency is 10 kHz. Figure 5.16 shows the simulation results obtained with Simulink. The four traces in the first plot are the reference voltage for each leg. The subsequent plots are the leg a output voltage, the leg n output voltage, the line-to-neutral output voltage and the line-to-line output voltage. Traces in black are the switched voltages and traces in gray are the same signals after filtering. Leg voltages have an homopolar component which is canceled in the line-to-neutral and line-to-line voltages. Therefore, even though the leg voltages do not follow exactly the reference voltages the line-to-neutral and line-to-line voltages are the expected signals.

The SVPWM algorithm was described for a five-level inverter by using VHDL and it was implemented in a Digilent S3 board. Table 5.7 shows a summary of the resources used by the implementation. It is important to remark that the BRAMs and the multipliers available in the FPGA were not used because the algorithm does not need data storage or multiplication operations. Figure 5.17 shows the FPGA output waveforms, which correspond to the trigger signals, for the same case illustrated in the example, where $\mathbf{v}_r = [1.39, -1.10, -0, 33, 1.15]^T$. The experimental results are in accordance with the theoretical results calculated in previous section.

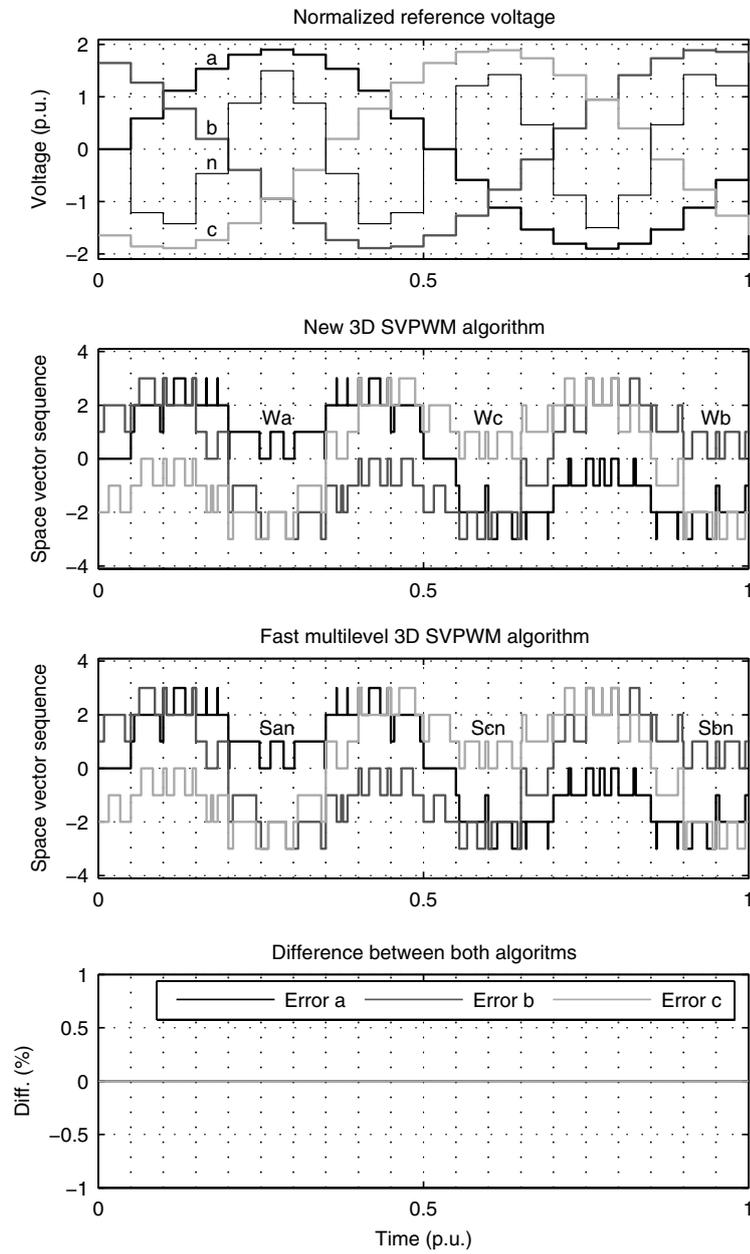


Figure 5.15: 3D algorithms comparison.

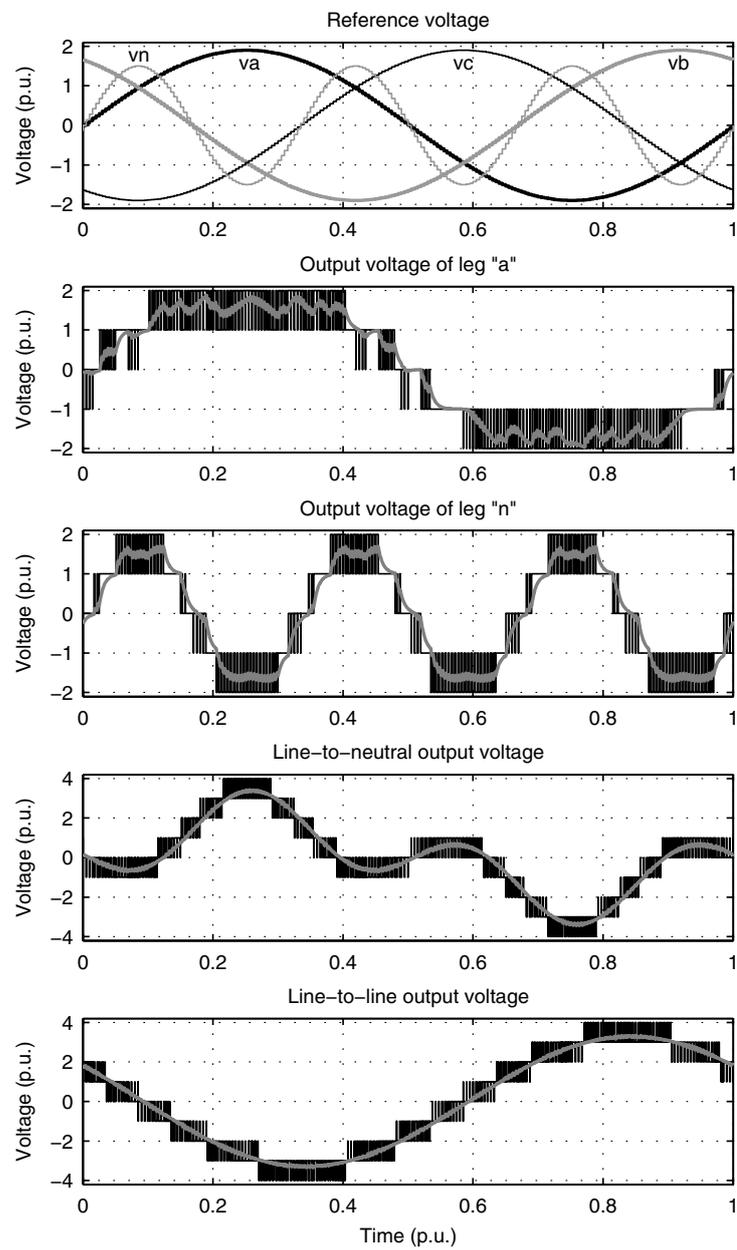


Figure 5.16: Simulation results.

TABLE 5.7
RESOURCES SUMMARY.

| | | | |
|---|--------------------|------|--|
| Target Device | : xc3s200 | | |
| Number of Slice Flip Flops: | 2,024 out of 3,840 | 52% | |
| Number of 4 input LUTs: | 2,330 out of 3,840 | 60% | |
| Number of occupied Slices: | 1,725 out of 1,920 | 89% | |
| Total Number 4 input LUTs: | 2,597 out of 3,840 | 67% | |
| Number of bonded IOBs: | 63 out of 173 | 36% | |
| IOB Flip Flops: | 66 | | |
| Number of Block RAMs: | 0 out of 12 | 0% | |
| Number of MULT18X18s: | 0 out of 12 | 0% | |
| Number of GCLKs: | 8 out of 8 | 100% | |
| Number of Startups: | 1 out of 1 | 100% | |
| Total equivalent gate count for design: | 32,226 | | |

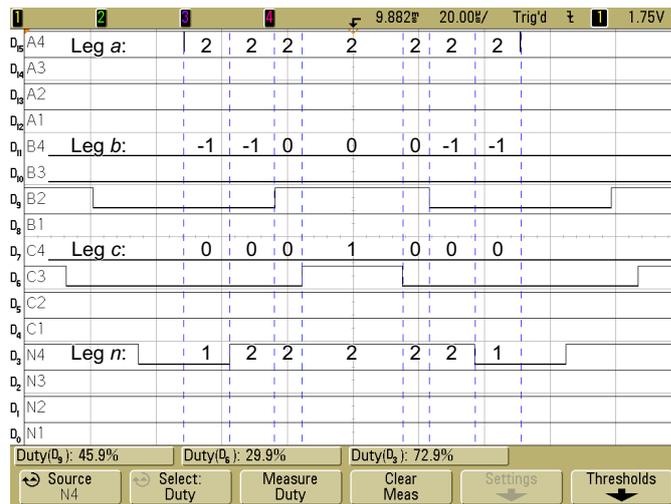


Figure 5.17: Trigger signals.

The new algorithm was tested in laboratory by using the same setup as in §4.6.3 so the measurement points and the reference voltage are the same displayed in Figure 4.19.

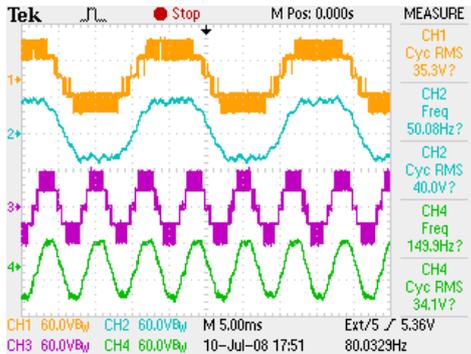
Figure 5.18 shows the measured output voltage of the inverter in the same case simulated in Figure 5.16. The experimental results agree with the simulation results. Line-to-line voltage, shown in Figure 5.18b, has a large third harmonic which corresponds to the leg n reference. Line-to-neutral voltage, shown in Figure 5.18c, is a nine-level voltage with a very low THD. As expected, neither the homopolar components of the leg voltages nor the third harmonic of the neutral voltage are present in this voltage.

5.7 Conclusion

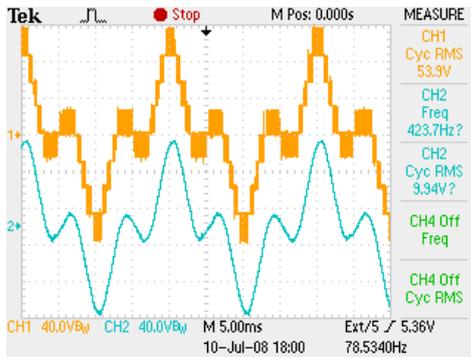
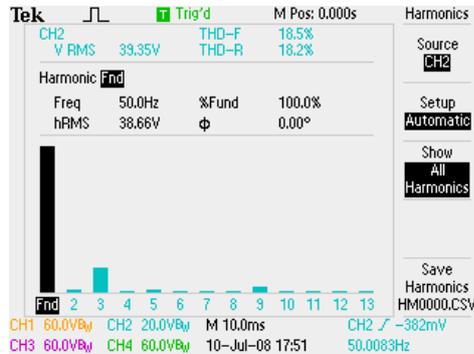
This chapter develops a novel multilevel multiphase SVPWM algorithm that takes advantage of joint-phase redundancy present in converters with no neutral wire. It is valid for any number of phases or levels and it can be used with the standard multilevel topologies. The presented modulation technique handles all switching states of the inverter and it provides a sorted switching vector sequence that minimizes the number of switchings. The algorithm makes it easy to select the switching sequence which allows implementing a switching strategy besides extending the modulation index. The proposed SVPWM algorithm is suitable for real-time implementation due to its low computational complexity. A five-level five-phase version was implemented in a low-cost FPGA and successfully tested by using a laboratory prototype.

In addition, the algorithm was particularized for three-phase converters with three and four legs. In both cases the new algorithms can be considered as an extension of two existing SVPWM algorithms where the problem of the redundant switching state selection has been solved.

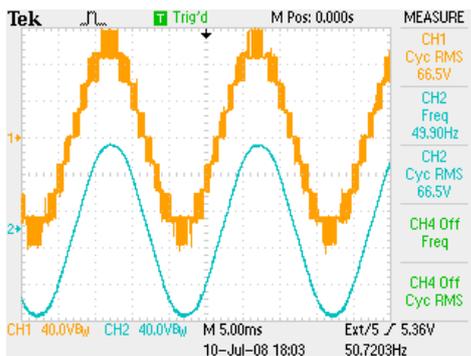
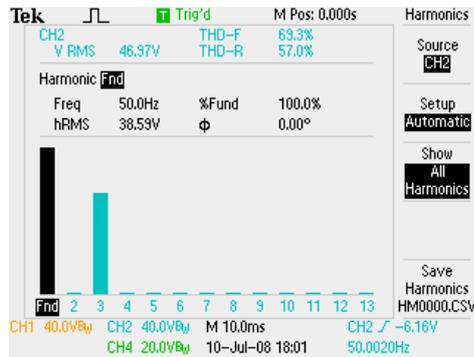
The new multilevel multiphase SVPWM developed in the chapter has been accepted for publication in a journal [173].



(a) Leg voltages V_s^a and V_s^n



(b) Line-to-neutral voltage $V_s^a - V_s^n$



(c) Line-to-line voltage $V_s^b - V_s^a$

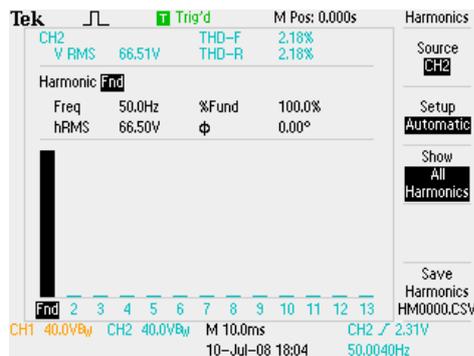


Figure 5.18: Inverter output voltage. Ch1 and Ch3: switched voltages; Ch2 and Ch4: filtered voltages.

Chapter 6

Conclusion and Future Research

6.1 Conclusion

This dissertation addresses the space vector pulse-width modulation (SVPWM) in multilevel multiphase voltage source converters. Its main contributions and conclusions are summarized below.

- The switching laws and per-phase redundancy of the diode clamped, flying capacitor and cascaded full bridge topologies have been studied. Analytical expressions to calculate the number of redundant switching states for each converter output level have been obtained. Principles to adapt the trigger signals for different multilevel topologies have been provided.
- Two well known multilevel three-phase SVPWM algorithms, one two-dimensional (2D) and the other three-dimensional (3D), have been implemented in a low-cost field-programmable gate array (FPGA). A modular design of the implementation of both algorithms has been presented. All modules developed can be easily extended to a higher number of levels, except for the component that carries out the redundant switching state selection in the 2D algorithm. This component implements a new selection technique to choose the switching states that has been designed to reduce switching losses in three-level three-phase inverters. The implementation of this task was identified as the major difficulty in the implementation process because of the absence of a general technique to manage joint-phase redundancy in multilevel converters. The benefits of using an FPGA to implement the modulation algorithm for converters with a high number of power switches, such as multilevel multiphase converters, have been verified.
- A novel formulation of the modulation problem in multilevel multiphase converters has been presented. This new formulation that was carried out in a multidimensional space has proved to be simple and very useful to study the modulation process. By using this new formulation the multilevel multiphase SVPWM problem has been solved for converters with and without switching state redundancy. In both cases, it was demonstrated that a multilevel modulator can be realized by means of a two-level modulator.

- The developed multilevel multiphase SVPWM algorithm without switching state redundancy is based on a two-level modulator plus a displacement. It is valid for any number of phases or levels and it can be used with the standard multilevel topologies. The new modulation technique handles all switching states of the converter and it provides a sorted switching vector sequence that minimizes the number of switchings. The algorithm has a low computational cost and it does not use trigonometric functions or tables to store predefined switching sequences. Therefore it is suitable for real-time implementation in low-cost devices. It was validated by the hardware implementation of the five-level five-phase version of the algorithm in a FPGA. The implementation was successfully tested in laboratory by using a cascaded full-bridge inverter.
- Since the new SVPWM technique can be used with any number of phases it was particularized for three-phase converters with three and four legs. In the former case the resulting algorithm is a 3D algorithm almost identical to the SVPWM generalized algorithm presented in [81] for multilevel converters. The algorithm obtained in the latter case is a new four-dimensional (4D) algorithm.
- The multilevel multiphase SVPWM algorithm with switching state redundancy is also based on the same two-level modulator used by the previous SVPWM algorithm without redundancy. This algorithm takes advantage of joint-phase redundancy present in converters without neutral wire, which allows extending the modulation index and including a switching strategy in the modulation process. The switching strategy can be implemented using the method included in the algorithm that makes easy the switching state selection. As in the previous algorithm, it can be applied to any number of levels or phases, it can be used with the standard topologies, it handles all switching states of the converter, and it provides a sorted switching sequence that minimizes the number of switchings. It has also a low computational cost and it does not use trigonometric functions or tables to store predefined switching sequences. Therefore, it is also suitable for real-time implementation in low-cost devices. It was validated by the hardware implementation of the five-level five-phase version of the algorithm in an FPGA. The implementation was successfully tested in laboratory by using a cascaded full-bridge inverter.
- The SVPWM algorithm with state redundancy was also particularized for three-phase converters with three and four legs. In the former case the resulting algorithm is a 2D algorithm that can be considered as an extension of the fast SVPWM algorithm for multilevel converters presented in [72]. The algorithm obtained in the latter case is a 3D algorithm that is an extension of the SVPWM algorithm for four-leg converters proposed in [89].
- Research work included in the dissertation has given rise to four journal papers [117, 137, 164, 173] and two conference papers [165, 166].

6.2 Future Research

There are several interesting topics suggested for further research in SVPWM for multilevel multiphase converters. These topics are described in the following.

- Further particularization for different number of phases and its comparison with existing SVPWM methods can be made. One interesting case is the particularization of the SVPWM with state redundancy for single-phase converters that can be compared with the recent SVPWM technique presented in [90].
- Further studies should be done on the spectrum of the output voltage to get a deeper knowledge on the new modulation techniques, and to compare them with classical modulation techniques.
- The switching strategy that can be included in the SVPWM with switching state redundancy greatly influences the performance of the power system. A research on possible switching strategies, such as capacitor balancing, dc-link current control or fault tolerance, and the way to implement them with the new modulation algorithm should be done. In particular, it is interesting to know its effect on the switching losses and the voltage spectrum. The results of the analysis will suggest the best switching strategy to be used depending on specifications required in each application.
- The case when the reference vector lies in the overmodulation region has not been studied. Therefore, the SVPWM problem should be also solved in this case.

References

- [1] R. H. Baker and L. H. Bannister, “Electric power converter,” U.S. Patent 3 867 643, Feb. 18, 1975.
- [2] R. H. Baker, “High-voltage converter circuit,” U.S. Patent 4 203 151, May 13, 1980.
- [3] A. Nabae, I. Takahashi, and H. Akagi, “A new neutral–point–clamped PWM inverter,” *IEEE Trans. Ind. Appl.*, vol. IA-17, no. 5, pp. 518–523, Sep.–Oct. 1981.
- [4] B. Singh, B. N. Singh, A. Chandra, K. Al-Haddad, A. Pandey, and D. Kothari, “A review of three-phase improved power quality AC-DC converters,” *IEEE Trans. Ind. Electron.*, vol. 51, no. 3, pp. 641–660, Jun. 2004.
- [5] N. Choi, J. Cho, and G. Cho, “A general circuit topology of multilevel inverter,” in *Proc. IEEE Power Electronics Specialists Conf. PESC*, Cambridge, MA, USA, 24–27 Jun. 1991.
- [6] T. A. Meynard and H. Foch, “Multi-level conversion: high voltage choppers and voltage-source inverters,” in *Proc. IEEE Power Electronics Specialists Conf. PESC*, vol. 1, Toledo, Spain, 29 Jun.–3 Jul. 1992, pp. 397–403.
- [7] J. Lai and F. Z. Peng, “Multilevel converters—A new breed of power converters,” *IEEE Trans. Ind. Appl.*, vol. 32, no. 3, pp. 509–517, May.–Jun. 1996.
- [8] J. Rodriguez, J.-S. Lai, and F. Z. Peng, “Multilevel inverters: a survey of topologies, controls, and applications,” *IEEE Trans. Ind. Electron.*, vol. 49, no. 4, pp. 724–738, Aug. 2002.
- [9] K. A. Corzine, “Operation and design of multilevel inverters,” University of Missouri - Rolla, Tech. Rep., 2005. [Online]. Available: <http://www.motorlab.com/>
- [10] T. Bruckner and D. Holmes, “Optimal pulse-width modulation for three-level inverters,” *IEEE Trans. Power Electron.*, vol. 20, no. 1, pp. 82–89, Jan. 2005.
- [11] M. Fracchia, T. Ghiara, M. Marchesoni, and M. Mazzucchelli, “Optimized modulation techniques for the generalized N-level converter,” in *Proc. IEEE Power Electronics Specialists Conf. PESC*, vol. 2, Toledo, Spain, 29 Jun.–3 Jul. 1992, pp. 1205–1213.
- [12] C. Feng, J. Liang, and V. Agelidis, “A novel voltage balancing control method for flying capacitor multilevel converters,” in *Proc. IEEE Industrial Electronics Society Conf. IECON*, vol. 2, Roanoke, VA, 2–6 Nov. 2003, pp. 1179–1184.

- [13] J. Huang and K. Corzine, "Extended operation of flying capacitor multilevel inverters," *IEEE Trans. Power Electron.*, vol. 21, no. 1, pp. 140–147, Jan. 2006.
- [14] P. Tekwani, R. Kanchan, and K. Gopakumar, "A dual five-level inverter-fed induction motor drive with common-mode voltage elimination and dc-link capacitor voltage balancing using only the switching-state redundancy—Part II," *IEEE Trans. Ind. Electron.*, vol. 54, no. 5, pp. 2609–2617, Oct. 2007.
- [15] M. Saeedifard, R. Iravani, and J. Pou, "Analysis and control of dc-capacitor-voltage-drift phenomenon of a passive front-end five-level converter," *IEEE Trans. Ind. Electron.*, vol. 54, no. 6, pp. 3255–3266, Dec. 2007.
- [16] L. M. Tolbert, F. Z. Peng, and T. G. Habetler, "Multilevel PWM methods at low modulation indices," *IEEE Trans. Power Electron.*, vol. 15, no. 4, pp. 719–725, Jul. 2000.
- [17] M. Calais, L. Borle, and V. Agelidis, "Analysis of multicarrier PWM methods for a single-phase five level inverter," in *Proc. IEEE Power Electronics Specialists Conf. PESC*, vol. 3, Vancouver, BC, 17–21 Jun. 2001, pp. 1351–1356.
- [18] D.-W. Kang, Y.-H. Lee, B.-S. Suh, C.-H. Choi, and D.-S. Hyun, "An improved carrier-based SVPWM method using leg voltage redundancies in generalized cascaded multilevel inverter topology," *IEEE Trans. Power Electron.*, vol. 18, no. 1, pp. 180–187, Jan. 2003.
- [19] X. Kou, K. Corzine, and Y. Familiant, "A unique fault-tolerant design for flying capacitor multilevel inverter," *IEEE Trans. Power Electron.*, vol. 19, no. 4, pp. 979–987, Jul. 2004.
- [20] J. Rodriguez, J. Pontt, P. Correa, P. Cortes, and C. Silva, "A new modulation method to reduce common-mode voltages in multilevel inverters," *IEEE Trans. Ind. Electron.*, vol. 51, no. 4, pp. 834–839, Aug. 2004.
- [21] R. H. Baker, "Switching circuit," U.S. Patent 4 210 826, Jul. 1, 1980.
- [22] T. A. Meynard and H. Foch, "Multi-level choppers for high voltage applications," *EPE Journal (European Power Electronics and Drives Journal)*, vol. 2, no. 1, pp. 45–50, 1992.
- [23] C. Hochgraf, R. Lasseter, D. Divan, and T. Lipo, "Comparison of multilevel inverters for static VAR compensation," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 2, Denver, CO, 2–6 Oct. 1994, pp. 921–928.
- [24] P. Hammond, "A new approach to enhance power quality for medium voltage AC drives," *IEEE Trans. Ind. Appl.*, vol. 33, no. 1, pp. 202–208, Jan.–Feb. 1997.
- [25] E. Cengelci, S. Sulistijo, B. Woo, P. Enjeti, R. Teoderescu, and F. Blaabjerg, "A new medium-voltage PWM inverter topology for adjustable-speed drives," *IEEE Trans. Ind. Appl.*, vol. 35, no. 3, pp. 628–637, May.–Jun. 1999.

- [26] J. Rodriguez, S. Bernet, B. Wu, J. Pontt, and S. Kouro, "Multilevel voltage-source-converter topologies for industrial medium-voltage drives," *IEEE Trans. Ind. Electron.*, vol. 54, no. 6, pp. 2930–2945, Dec. 2007.
- [27] J. R. Rodriguez, J. W. Dixon, J. R. Espinoza, J. Pontt, and P. Lezana, "PWM regenerative rectifiers: state of the art," *IEEE Trans. Ind. Electron.*, vol. 52, no. 1, pp. 5–22, Feb. 2005.
- [28] M. Marchesoni, P. Segarich, and E. Soressi, "A new control strategy for neutral-point-clamped active rectifiers," *IEEE Trans. Ind. Electron.*, vol. 52, no. 2, pp. 462–470, Apr. 2005.
- [29] L. Yacoubi, K. Al-Haddad, L.-A. Dessaint, and F. Fnaiech, "Linear and nonlinear control techniques for a three-phase three-level NPC boost rectifier," *IEEE Trans. Ind. Electron.*, vol. 53, no. 6, pp. 1908–1918, Dec. 2006.
- [30] S. Alepuz, S. Busquets-Monge, J. Bordonau, J. Gago, D. Gonzalez, and J. Balcells, "Interfacing renewable energy sources to the utility grid using a three-level inverter," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1504–1511, Oct. 2006.
- [31] J. M. Carrasco, L. G. Franquelo, J. T. Bialasiewicz, E. Galvan, R. Portillo, M. M. Prats, J. I. Leon, and N. Moreno-Alfonso, "Power-electronic systems for the grid integration of renewable energy sources: A survey," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1002–1016, Jun. 2006.
- [32] R. Portillo, M. M. Prats, J. I. Leon, J. A. Sanchez, J. M. Carrasco, E. Galvan, and L. G. Franquelo, "Modeling strategy for back-to-back three-level converters applied to high-power wind turbines," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1483–1491, Oct. 2006.
- [33] Y. Cheng, C. Qian, M. Crow, S. Pekarek, and S. Atcitty, "A comparison of diode-clamped and cascaded multilevel converters for a STATCOM with energy storage," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1512–1521, Oct. 2006.
- [34] B. Renz, A. Keri, A. Mehraban, C. Schauder, E. Stacey, L. Kovalsky, L. Gyugyi, and A. Edris, "AEP unified power flow controller performance," *IEEE Trans. Power Del.*, vol. 14, no. 4, pp. 1374–1381, Oct. 1999.
- [35] J. Lyons, V. Vlatkovic, P. Espelage, F. Boettner, and E. Larsen, "Innovation IGCT main drives," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 4, Phoenix, AZ, 3–7 Oct. 1999, pp. 2655–2661.
- [36] J. Rodriguez, J. Pontt, G. Alzamora, N. Becker, O. Einkenkel, J. Cornet, and A. Weinstein, "Novel 20 MW downhill conveyor system using three-level converters," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 2, Chicago, IL, 30 Sep.–4 Oct. 2001, pp. 1396–1403.
- [37] J. Ainsworth, M. Davies, P. Fitz, K. Owen, and D. Trainer, "Static VAr compensator (STATCOM) based on single-phase chain circuit converters," *IEE Proc. Generation, Transmission and Distribution*, vol. 145, no. 4, pp. 381–386, Jul. 1998.

- [38] M. Baker, B. Gemmell, C. Horwill, and D. Hanson, "STATCOM helps to guarantee a stable system," in *Proc. IEEE/PES Transmission and Distribution Conf. and Exposition*, vol. 2, Atlanta, GA, 28 Oct.–2 Nov. 2001, pp. 1129–1132.
- [39] E. Ward and H. Harer, "Preliminary investigation of an inverter-fed 5-phase induction motor," *Proc. IEE.*, vol. 116, no. 6, pp. 980–984, 1969.
- [40] G. K. Singh, "Multi-phase induction machine drive research—A survey," *Electric Power Systems Research*, vol. 61, no. 2, pp. 139–147, 28 Mar. 2002.
- [41] M. Jones and E. Levi, "A literature survey of state-of-the-art in multiphase AC drives," in *Proc. Universities Power Engineering Conf. UPEC*, Stafford, UK, 9–11 Sep. 2002, pp. 505–510.
- [42] R. Bojoi, F. Farina, F. Profumo, and A. Tenconi, "Dual-three phase induction machine drives control—A survey," *IEEE Transactions on Industry Applications*, vol. 126, no. 4, pp. 420–429, 2006.
- [43] E. Levi, R. Bojoi, F. Profumo, H. A. Toliyat, and S. Williamson, "Multiphase induction motor drives—A technology status review," *IET Electric Power Applications*, vol. 1, pp. 489–516, Jul. 2007.
- [44] E. Levi, "Multiphase electric machines for variable-speed applications," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1893–1909, May 2008.
- [45] L. Zheng, J. Fletcher, B. Williams, and X. He, "Dual-plane vector control of a five-phase induction machine for an improved flux pattern," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1996–2005, May 2008.
- [46] M. Duran, F. Salas, and M. Arahal, "Bifurcation analysis of five-phase induction motor drives with third harmonic injection," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 2006–2014, May 2008.
- [47] M.-A. Shamsi-Nejad, B. Nahid-Mobarakeh, S. Pierfederici, and F. Meibody-Tabar, "Fault tolerant and minimum loss control of double-star synchronous machines under open phase conditions," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1956–1965, May 2008.
- [48] R. Kianinezhad, B. Nahid-Mobarakeh, L. Baghli, F. Betin, and G.-A. Capolino, "Modeling and control of six-phase symmetrical induction machine under fault condition due to open phases," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1966–1977, May 2008.
- [49] N. Bianchi, S. Bolognani, and M. Dai Pre, "Impact of stator winding of a five-phase permanent-magnet motor on postfault operations," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1978–1987, May 2008.
- [50] S. Dwari and L. Parsa, "An optimal control technique for multiphase PM machines under open-circuit faults," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1988–1995, May 2008.

- [51] E. Levi, M. Jones, S. N. Vukosavic, A. Iqbal, and H. A. Toliyat, "Modeling, control, and experimental investigation of a five-phase series-connected two-motor drive with single inverter supply," *IEEE Trans. Ind. Electron.*, vol. 54, no. 3, pp. 1504–1516, Jun. 2007.
- [52] K. Mohapatra, R. Kanchan, M. Baiju, P. Tekwani, and K. Gopakumar, "Independent field-oriented control of two split-phase induction motors from a single six-phase inverter," *IEEE Trans. Ind. Electron.*, vol. 52, no. 5, pp. 1372–1382, Oct. 2005.
- [53] F. Terrien, S. Siala, and P. Noy, "Multiphase induction motor sensorless control for electric ship propulsion," in *Proc. IEE Inter. Conf. on Power Electronics, Machines and Drives PEMD*, vol. 2, Edinburgh, Scotland, 31 Mar.–2 Apr. 2004, pp. 556–561.
- [54] D. Gritter, S. Kalsi, and N. Henderson, "Variable speed electric drive options for electric ships," in *Proc. IEEE Electric Ship Technologies Symposium ESTS*, Philadelphia, PA, 2005, pp. 347–354.
- [55] L. Parsa and H. A. Toliyat, "Five-phase permanent magnet motor drives for ship propulsion applications," in *Proc. IEEE Electric Ship Technologies Symposium ESTS*, Philadelphia, PA, 25–27 Jul. 2005, pp. 371–378.
- [56] G. Atkinson, B. Mecrow, A. Jack, D. Atkinson, P. Sangha, and M. Benarous, "The design of fault tolerant machines for aerospace applications," in *Proc. IEEE Inter. Electric Machines and Drives Conf. IEMDC*, San Antonio, TX, 15–18 May 2005, pp. 1863–1869.
- [57] J. Bennett, B. Mecrow, A. Jack, D. Atkinson, C. Sewell, G. Mason, S. Sheldon, and B. Cooper, "Choice of drive topologies for electrical actuation of aircraft flaps and slats," in *Proc. IEE Inter. Conf. on Power Electronics, Machines and Drives PEMD*, vol. 1, Edinburgh, UK, 31 Mar.–2 Apr. 2004, pp. 332–337.
- [58] G. Atkinson, B. Mecrow, A. Jack, D. Atkinson, P. Sangha, and M. Benarous, "The analysis of losses in high-power fault-tolerant machines for aerospace applications," *IEEE Trans. Ind. Appl.*, vol. 42, no. 5, pp. 1162–1170, Sep./Oct. 2006.
- [59] X. Huang, K. Bradley, A. Goodman, C. Gerada, P. Wheeler, J. Clare, and C. Whitley, "Fault-tolerant brushless dc motor drive for electro-hydrostatic actuation system in aerospace application," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 1, Tampa, FL, 8–12 Oct. 2006, pp. 473–480.
- [60] M. G. Simoes and P. Vieira, "A high-torque low-speed multiphase brushless machine—A perspective application for electric vehicles," *IEEE Trans. Ind. Electron.*, vol. 49, no. 5, pp. 1154–1164, Oct. 2002.
- [61] S. Jiang, K. Chau, and C. Chan, "Spectral analysis of a new six-phase pole-changing induction motor drive for electric vehicles," *IEEE Trans. Ind. Electron.*, vol. 50, no. 1, pp. 123–131, Feb. 2003.

- [62] R. Bojoi, A. Tenconi, F. Profumo, and F. Farina, "Dual-source fed multi-phase induction motor drive for fuel cell vehicles: Topology and control," in *Proc. IEEE Power Electronics Specialists Conf. PESC*, Recife, Brazil, 11–16 Jun. 2005, pp. 2676–2683.
- [63] S. Niu, K. Chau, D. Zhang, J. Jiang, and Z. Wang, "Design and control of a double-stator permanent-magnet motor drive for electric vehicles," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, New Orleans, LA, 23–27 Sep. 2007, pp. 1293–1300.
- [64] M. Steiner, R. Deplazes, and H. Stemmler, "New transformerless topology for ac-fed traction vehicles using multi-star induction motors," *EPE Journal (European Power Electronics and Drives Journal)*, vol. 10, no. 3-4, pp. 45–53, Sep. 2000.
- [65] M. Abolhassani, "A novel multiphase fault tolerant high torque density permanent magnet motor drive for traction application," in *Proc. IEEE Inter. Electric Machines and Drives Conf. IEMDC*, San Antonio, TX, 15–18 May 2005, pp. 728–734.
- [66] S. Lu and K. Corzine, "Multilevel multi-phase propulsion drives," in *Proc. IEEE Electric Ship Technologies Symposium ESTS*, Philadelphia, PA, 25–27 Jul. 2005, pp. 363–370.
- [67] K. Corzine, S. Sudhoff, E. Lewis, D. Schmucker, R. Youngs, and H. Hegner, "Use of multi-level converters in ship propulsion drives," in *Proc. All Electric Ship Conf.*, vol. 1, London, UK, Sep. 1998, pp. 155–163.
- [68] Z. Oudjebour, E. Berkouk, N. Sami, S. Belgasmi, S. Arezki, and I. Messaif, "Indirect space vector control of a double star induction machine fed by two five-levels NPC VSI," in *Proc. Inter. Conf. on Electrical Machines ICEM*, Krakow, Poland, 2004, cD-ROM paper no. 155.
- [69] Q. Song, X. Zhang, F. Yu, and C. Zhang, "Research on space vector PWM of five-phase three-level inverter," in *Proc. Inter. Conf. Electrical Machines and Systems ICEMS*, vol. 2, Nanjing, China, 27–29 Sep. 2005, pp. 1418–1421.
- [70] —, "Research on PWM techniques of five-phase three-level inverter," in *Proc. Inter. Symposium Power Electronics, Electrical Drives, Automation and Motion SPEEDAM*, Taormina, Italy, 23–26 May 2006, pp. 561–565.
- [71] K. Mohapatra, K. Gopakumar, V. Somasekhar, and L. Umanand, "A novel modulation scheme for a six phase induction motor with open-end windings," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, vol. 1, Sevilla, Spain, 5–8 Nov. 2002, pp. 810–815.
- [72] N. Celanovic and D. Boroyevich, "A fast space-vector modulation algorithm for multilevel three-phase converters," *IEEE Trans. Ind. Appl.*, vol. 37, no. 2, pp. 637–641, Mar. 2001.

- [73] L. Li, D. Czarkowski, Y. Liu, and P. Pillay, "Multilevel selective harmonic elimination PWM technique in series-connected voltage inverters," *IEEE Trans. Ind. Appl.*, vol. 36, no. 1, pp. 160–170, Jan.–Feb. 2000.
- [74] S. Sirisukprasert, J.-S. Lai, and T.-H. Liu, "Optimum harmonic reduction with a wide range of modulation indexes for multilevel converters," *IEEE Trans. Ind. Electron.*, vol. 49, no. 4, pp. 875–881, Aug. 2002.
- [75] J. Rodriguez, L. Moran, P. Correa, and C. Silva, "A vector control technique for medium-voltage multilevel inverters," *IEEE Trans. Ind. Electron.*, vol. 49, no. 4, pp. 882–888, Aug. 2002.
- [76] J. Rodriguez, L. Moran, J. Pontt, P. Correa, and C. Silva, "A high-performance vector control of an 11-level inverter," *IEEE Trans. Ind. Electron.*, vol. 50, no. 1, pp. 80–85, Feb. 2003.
- [77] A. Hava, R. Kerkman, and T. Lipo, "Simple analytical and graphical methods for carrier-based PWM-VSI drives," *IEEE Trans. Power Electron.*, vol. 14, no. 1, pp. 49–61, Jan. 1999.
- [78] L. M. Tolbert and T. G. Habetler, "Novel multilevel inverter carrier-based PWM method," *IEEE Trans. Ind. Appl.*, vol. 35, no. 5, pp. 1098–1107, Sep.–Oct. 1999.
- [79] B. McGrath and D. Holmes, "Multicarrier PWM strategies for multilevel inverters," *IEEE Trans. Ind. Electron.*, vol. 49, no. 4, pp. 858–867, Aug. 2002.
- [80] H. van der Broeck, H. Skudelny, and G. Stanke, "Analysis and realization of a pulsewidth modulator based on voltage space vectors," *IEEE Trans. Ind. Appl.*, vol. 2, no. 1, pp. 142–150, Jan.–Feb. 1988.
- [81] M. M. Prats, L. G. Franquelo, J. I. Leon, R. Portillo, E. Galvan, and J. M. Carrasco, "A 3-D space vector modulation generalized algorithm for multilevel converters," *IEEE Power Electron. Lett.*, vol. 1, no. 4, pp. 110–114, Dec. 2003.
- [82] W. Hill and C. Harbourt, "Performance of medium voltage multi-level inverters," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 2, Phoenix, AZ, 3–7 Oct. 1999, pp. 1186–1192.
- [83] J. Kelly, E. Strangas, and J. Miller, "Multi-phase inverter analysis," in *Proc. IEEE Inter. Electric Machines and Drives Conf. IEMDC*, Cambridge, MA, 17–20 Jun. 2001, pp. 147–155.
- [84] O. Ojo and G. Dong, "Generalized discontinuous carrier-based PWM modulation scheme for multi-phase converter-machine systems," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 2, Hong Kong, China, 2–6 Oct. 2005, pp. 1374–1381.
- [85] A. Iqbal, E. Levi, M. Jones, and S. N. Vukosavic, "Generalised sinusoidal PWM with harmonic injection for multi-phase VSIs," in *Proc. IEEE Power Electronics Specialists Conf. PESC*, Jeju, Korea, 18–22 Jun. 2006, pp. 2871–2877.

- [86] L. Hou, Y. Su, and L. Chen, "DSP-based indirect rotor flux oriented control for multiphase induction machines," in *Proc. IEEE Inter. Electric Machines and Drives Conf. IEMDC*, Madison, WI, 1–4 Jun. 2003, pp. 976–980.
- [87] A. Iqbal, E. Levi, M. Jones, and S. N. Vukosavic, "A PWM scheme for a five-phase VSI supplying a five-phase two-motor drive," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, Paris, France, 6–10 Nov. 2006, pp. 2575–2580.
- [88] S. R. Bowes and D. Holliday, "Optimal regular-sampled PWM inverter control techniques," *IEEE Trans. Ind. Electron.*, vol. 54, no. 3, pp. 1547–1559, Jun. 2007.
- [89] L. G. Franquelo, M. M. Prats, R. Portillo, J. I. Leon, M. A. Perales, J. M. Carrasco, E. Galvan, and J. L. Mora, "Three-dimensional space-vector modulation algorithm for four-leg multilevel converters using abc coordinates," *IEEE Trans. Ind. Electron.*, vol. 53, no. 2, pp. 458–466, Apr. 2006.
- [90] J. I. Leon, R. Portillo, L. G. Franquelo, S. Vazquez, J. M. Carrasco, and E. Dominguez, "New space vector modulation technique for single-phase multilevel converters," in *Proc. IEEE Inter. Symposium on Industrial Electronics ISIE*, Vigo, 4–7 Jun. 2007, pp. 617–622.
- [91] H. Zhang, A. Von Jouanne, S. Dai, A. Wallace, and F. Wang, "Multilevel inverter modulation schemes to eliminate common-mode voltages," *IEEE Trans. Ind. Appl.*, vol. 36, no. 6, pp. 1645–1653, Nov.–Dec. 2000.
- [92] J. H. Seo, C. H. Choi, and D. S. Hyun, "A new simplified space-vector PWM method for three-level inverters," *IEEE Trans. Power Electron.*, vol. 16, no. 4, pp. 545–550, Jul. 2001.
- [93] P. C. Loh and D. Holmes, "Flux modulation for multilevel inverters," *IEEE Trans. Ind. Appl.*, vol. 38, no. 5, pp. 1389–1399, Sept.–Oct. 2002.
- [94] A. Gupta and A. Khambadkone, "A space vector PWM scheme for multilevel inverters based on two-level space vector PWM," *IEEE Trans. Ind. Electron.*, vol. 53, no. 5, pp. 1631–1639, Oct. 2006.
- [95] A. Cataliotti, F. Genduso, A. Raciti, and G. R. Galluzzo, "Generalized PWM–VSI control algorithm based on a universal duty-cycle expression: Theoretical analysis, simulation results, and experimental validations," *IEEE Trans. Ind. Electron.*, vol. 54, no. 3, pp. 1569–1580, Jun. 2007.
- [96] P. de Silva, J. Fletcher, and B. Williams, "Development of space vector modulation strategies for five phase voltage source inverters," in *Proc. IEE Inter. Conf. on Power Electronics, Machines and Drives PEMD*, vol. 2, Edinburgh, UK, 31 Mar.–2 Apr. 2004, pp. 650–655.
- [97] S. Xue and X. Wen, "Simulation analysis of two novel multiphase SVPWM strategies," in *Proc. IEEE Inter. Conf. Industrial Technology ICIT*, Hong Kong, China, 14–17 Dec. 2005, pp. 1337–1342.

- [98] H.-M. Ryu, J.-H. Kim, and S.-K. Sul, "Analysis of multiphase space vector pulse-width modulation based on multiple d-q spaces concept," *IEEE Trans. Power Electron.*, vol. 20, no. 6, pp. 1364–1371, Nov. 2005.
- [99] S. Xue, X. Wen, and Z. Feng, "Multiphase permanent magnet motor drive system based on a novel multiphase SVPWM," in *Proc. CES/IEEE Inter. Power Electronics and Motion Control Conf. IPEMC*, vol. 1, Shanghai, China, 14–16 Aug. 2006.
- [100] G. Grandi, G. Serra, and A. Tani, "Space vector modulation of a seven-phase voltage source inverter," in *Proc. Inter. Symposium Power Electronics, Electrical Drives, Automation and Motion SPEEDAM*, Taormina, Italy, 23–26 May 2006, pp. 1149–1156.
- [101] Y. Zhao and T. Lipo, "Space vector PWM control of dual three-phase induction machine using vector space decomposition," *IEEE Trans. Ind. Appl.*, vol. 31, no. 5, pp. 1100–1109, Sep. 1995.
- [102] H. A. Toliyat and H. Xu, "DSP-based direct torque control (DTC) for five-phase induction machines," in *Proc. Inter. Power Electronics Conf. IPEC*, Tokyo, Japan, 3–7 Apr. 2000, pp. 1195–1200.
- [103] H. A. Toliyat, R. Shi, and H. Xu, "A DSP-based vector control of five-phase synchronous reluctance," in *Proc. Industry Applications Conf., IAS Annual Meeting*, vol. 3, Rome, Italy, 8–12 Oct. 2000, pp. 1759–1765.
- [104] J. Kelly, E. Strangas, and J. Miller, "Multiphase space vector pulse width modulation," *IEEE Trans. Energy Convers.*, vol. 18, no. 2, pp. 259–264, Jun. 2003.
- [105] A. Iqbal and E. Levi, "Space vector PWM techniques for sinusoidal output voltage generation with a five-phase voltage source inverter," *Electric Power Components and Systems*, vol. 34, no. 2, pp. 119–140, 2006.
- [106] G. Grandi, G. Serra, and A. Tani, "Space vector modulation of a nine-phase voltage source inverter," in *Proc. IEEE Inter. Symposium on Industrial Electronics ISIE*, Vigo, Spain, 4–7 Jun. 2007, pp. 431–436.
- [107] R. Lyra and T. Lipo, "Torque density improvement in a six-phase induction motor with third harmonic current injection," *IEEE Trans. Ind. Appl.*, vol. 38, no. 5, pp. 1351–1360, Sept.–Oct 2002.
- [108] D. Hadiouche, L. Baghli, and A. Rezzoug, "Space-vector PWM techniques for dual three-phase AC machine: analysis, performance evaluation, and DSP implementation," *IEEE Trans. Ind. Appl.*, vol. 42, no. 4, pp. 1112–1122, Jul.–Aug. 2006.
- [109] O. Ojo, G. Dong, and Z. Wu, "Pulse-width modulation for five-phase converters based on device turn-on times," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 2, Tampa, FL, 8–12 Oct. 2006, pp. 627–634.
- [110] E. Semail and C. Rombaut, "New method to calculate the conduction durations of the switches in a n-leg 2-level voltage source inverter," in *Proc. European Conf. on Power Electronics and Applications EPE*, Graz, Austria, 27–29 Aug. 2001.

- [111] X. Kestelyn, E. Semail, and J. P. Hautier, "Multi-phase system supplied by SVM VSI: A new fast algorithm to compute duty cycles," *EPE Journal (European Power Electronics and Drives Journal)*, vol. 14, no. 3, pp. 25–31, Jun. 2004.
- [112] A. Iqbal and E. Levi, "Space vector PWM for a five-phase VSI supplying two five-phase series-connected machines," in *Proc. IEEE Inter. Power Electronics and Motion Control Conf. PEMC*, Portoroz, Slovenia, 30 Aug.–1 Sep. 2006, pp. 222–227.
- [113] D. Dujic, G. Grandi, M. Jones, and E. Levi, "A space vector PWM scheme for multifrequency output voltage generation with multiphase voltage-source inverters," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1943–1955, May 2008.
- [114] S. Gataric, "A polyphase cartesian vector approach to control of polyphase ac machines," in *Proc. IEEE Industry Applications Conf., IAS Annual meeting*, vol. 3, Rome, Italy, 8–12 Oct. 2000, pp. 1648–1654.
- [115] M. J. Duran and E. Levi, "Multi-dimensional approach to multi-phase space vector pulse width modulation," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, Paris, France, 7–10 Nov. 2006, pp. 2103–2108.
- [116] M. Duran, S. Toral, F. Barrero, and E. Levi, "Real-time implementation of multi-dimensional five-phase space vector pulse-width modulation," *Electronics Letters*, vol. 43, no. 17, pp. 949–950, Aug. 2007.
- [117] O. López, J. Doval-Gandoy, C. M. Peñalver, J. Rey, and F. D. Freijedo, "Redundancy and basic switching rules in multilevel converters," *Int. Review of Electrical Engineering*, vol. 0, no. 0, pp. 66–73, Jan.–Feb. 2006.
- [118] B. McGrath, D. Holmes, and T. Lipo, "Optimized space vector switching sequences for multilevel inverters," *IEEE Trans. Power Electron.*, vol. 18, no. 6, pp. 1293–1301, Nov. 2003.
- [119] A. Hava, R. Kerkman, and T. Lipo, "Carrier-based PWM-VSI overmodulation strategies: analysis, comparison, and design," *IEEE Trans. Power Electron.*, vol. 13, no. 4, pp. 674–689, Jul. 1998.
- [120] O. Ojo, "The generalized discontinuous PWM scheme for three-phase voltage source inverters," *IEEE Trans. Ind. Electron.*, vol. 51, no. 6, pp. 1280–1289, Dec. 2004.
- [121] F. Z. Peng, Z. Pan, K. Corzine, V. Stefanovic, and M. Leuthen, "Voltage balancing control of diode-clamped multilevel rectifier/inverter systems," in *Proc. IEEE Industry Applications Conf., IAS Annual Meeting*, vol. 1, Salt Lake City, UT, 12–16 Oct. 2003, pp. 182–189.
- [122] Z. Pan, F. Peng, K. Corzine, V. Stefanovic, J. Leuthen, and S. Gataric, "Voltage balancing control of diode-clamped multilevel rectifier/inverter systems," *IEEE Trans. Ind. Appl.*, vol. 41, no. 6, pp. 1698–1706, Nov.–Dec. 2005.
- [123] P. C. Krause, O. Wasynczuk, and S. D. Sudhoff, *Analysis of electric machinery and drive systems*, Wiley, Ed. IEEE Press, 2002.

- [124] M. Marchesoni and P. Tenca, "Diode-clamped multilevel converters: a practicable way to balance dc-link voltages," *IEEE Trans. Ind. Electron.*, vol. 49, no. 4, pp. 752–765, Aug. 2002.
- [125] J. Pou Fèlix, "Modulation and control of three-phase PWM multilevel converters," Ph.D. dissertation, Universitat Politècnica de Catalunya, Nov. 2002. [Online]. Available: <http://www.tdx.cesca.es/TDX-0921104-155059/>
- [126] M. Marchesoni and P. Tenca, "Theoretical and practical limits in multilevel MPC inverters with passive front ends," in *Proc. European Conf. on Power Electronics and Applications EPE*, Graz, Austria, 27–29 Aug. 2004.
- [127] M. Marchesoni, M. Mazzucchelli, and S. Tenconi, "A nonconventional power converter for plasma stabilization," *IEEE Trans. Power Electron.*, vol. 5, no. 2, pp. 212–219, Apr. 1990.
- [128] F. Z. Peng, J. Lai, J. W. McKeever, and J. VanCoevering, "A multilevel voltage-source inverter with separate dc sources for static VAR generation," *IEEE Trans. Ind. Appl.*, vol. 32, no. 5, pp. 1130–1138, Sep.–Oct. 1996.
- [129] M. Manjrekar, P. Steimer, and T. Lipo, "Hybrid multilevel power conversion system: a competitive solution for high-power applications," *IEEE Trans. Ind. Appl.*, vol. 36, no. 3, pp. 834–841, May./Jun. 2000.
- [130] K. Corzine and S. Lu, "Comparison of hybrid propulsion drive schemes," in *Proc. IEEE Electric Ship Technologies Symposium ESTS*, Philadelphia, PA, 25–27 Jul. 2005, pp. 355–362.
- [131] M. Veenstra and A. Rufer, "Control of a hybrid asymmetric multilevel inverter for competitive medium-voltage industrial drives," *IEEE Trans. Ind. Appl.*, vol. 41, no. 2, pp. 655–664, Mar.–Apr. 2005.
- [132] L. M. Tolbert, F. Z. Peng, T. Cunnyngham, and J. N. Chiasson, "Charge balance control schemes for cascade multilevel converter in hybrid electric vehicles," *IEEE Trans. Ind. Appl.*, vol. 49, no. 5, pp. 1058–1064, Oct. 2002.
- [133] B. Ozpineci, L. Tolbert, and Z. Du, "Optimum fuel cell utilization with multilevel inverters," in *Proc. IEEE Power Electronics Specialists Conf. PESC*, vol. 6, Aachen, Germany, 20–25 Jun. 2004, pp. 4798–4802.
- [134] F. Z. Peng and J. Lai, "Dynamic performance and control of a static VAR generator using cascade multilevel inverters," *IEEE Trans. Ind. Appl.*, vol. 33, no. 3, pp. 748–755, May.–Jun. 1997.
- [135] M. Calais, V. G. Agelidis, and M. Meinhardt, "Multilevel converters for single-phase grid connected photovoltaic systems: An overview," *Solar Energy*, vol. 66, no. 5, pp. 325–335, Aug. 1999.

- [136] O. Alonso, P. Sanchis, E. Gubia, and L. Marroyo, "Cascaded H-bridge multilevel converter for grid connected photovoltaic generators with independent maximum power point tracking of each solar array," in *Proc. IEEE Power Electronics Specialists Conf. PESC*, vol. 2, Acapulco, Mexico, 15–19 Jun. 2003, pp. 731–735.
- [137] O. López, J. Alvarez, J. Doval-Gandoy, F. D. Freijedo, A. Nogueiras, A. Lago, and C. M. Peñalver, "Comparison of the FPGA implementation of two multilevel space vector PWM algorithms," *IEEE Trans. Ind. Electron.*, vol. 55, no. 4, pp. 1537–1547, Apr. 2008.
- [138] K. Corzine and J. Baker, "Multilevel voltage-source duty-cycle modulation: analysis and implementation," *IEEE Trans. Ind. Electron.*, vol. 49, no. 5, pp. 1009–1016, Oct. 2002.
- [139] J. J. Rodriguez-Andina, M. J. Moure, and M. D. Valdes, "Features, design tools, and application domains of FPGAs," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1810–1823, Aug. 2007.
- [140] K. Sridharan and T. Priya, "The design of a hardware accelerator for real-time complete visibility graph construction and efficient FPGA implementation," *IEEE Trans. Ind. Electron.*, vol. 52, no. 4, pp. 1185–1187, Aug. 2005.
- [141] S. Sanchez-Solano, A. J. Cabrera, I. Baturone, F. J. Moreno-Velo, and M. Brox, "FPGA implementation of embedded fuzzy controllers for robotic applications," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1937–1945, Aug. 2007.
- [142] M.-W. Naouar, E. Monmasson, A. A. Naassani, I. Slama-Belkhodja, and N. Patin, "FPGA-based current controllers for AC machine drives—A review," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1907–1925, Aug. 2007.
- [143] E. Monmasson and M. N. Cirstea, "FPGA design methodology for industrial control systems—A review," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1824–1842, Aug. 2007.
- [144] M. Aime, G. Gateau, and T. A. Meynard, "Implementation of a peak-current-control algorithm within a field-programmable gate array," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 406–418, Feb. 2007.
- [145] B. Lu, X. Wu, H. Figueroa, and A. Monti, "A low-cost real-time hardware-in-the-loop testing approach of power electronics controls," *IEEE Trans. Ind. Electron.*, vol. 54, no. 2, pp. 919–931, Apr. 2007.
- [146] Y. F. Chan, M. Moallem, and W. Wang, "Design and implementation of modular FPGA-based PID controllers," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1898–1906, Aug. 2007.
- [147] Y.-A. Chapuis, L. Zhou, Y. Fukuta, Y. Mita, and H. Fujita, "FPGA-based decentralized control of arrayed MEMS for microrobotic application," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1926–1936, Aug. 2007.

- [148] M. N. Cirstea and A. Dinu, "A VHDL holistic modeling approach and FPGA implementation of a digital sensorless induction motor control scheme," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1853–1864, Aug. 2007.
- [149] J. Qin, C. Stroud, and F. F. Dai, "FPGA-based analog functional measurements for adaptive control in mixed-signal systems," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1885–1897, Aug. 2007.
- [150] H. Du, H. Qi, and X. Wang, "Comparative study of VLSI solutions to independent component analysis," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 548–558, Feb. 2007.
- [151] H. Zhuang, K.-S. Low, and W.-Y. Yau, "A pulsed neural network with on-chip learning and its practical applications," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 34–42, Feb. 2007.
- [152] A.-M. Lienhardt, G. Gateau, and T. A. Meynard, "Digital sliding-mode observer implementation using FPGA," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1865–1875, Aug. 2007.
- [153] S. Jung and S. su Kim, "Hardware implementation of a real-time neural network controller with a DSP and an FPGA for nonlinear systems," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 265–271, Feb. 2007.
- [154] R. Dubey, P. Agarwal, and M. K. Vasantha, "Programmable logic devices for motion control—A review," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 559–566, Feb. 2007.
- [155] A. Polic, M. Curkovic, and K. Jezernik, "Matrix based design of the current source inverter with FPGA based implementation," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, vol. 2, Busan, Korea, 2–6 Nov. 2004, pp. 1640–1645.
- [156] R. Garcia-Gil, J. Espi, E. Dede, and E. Maset, "An all-digital controlled AC-DC matrix converter with high-frequency isolation and power factor correction," in *Proc. IEEE Inter. Symposium on Industrial Electronics ISIE*, vol. 2, 4–7 May 2004, pp. 1075–1080.
- [157] J. Van den Keybus, B. Bolsens, K. De Brabandere, J. Driesen, and R. Belmans, "DSP and FPGA based platform for rapid prototyping of power electronic converters and its application to a sampled-data three-phase dual-band hysteresis current controller," in *Proc. IEEE Power Electronics Specialists Conf. PESC*, vol. 4, Cairns, Australia, 23–27 Jun. 2002, pp. 1722–1727.
- [158] Y.-Y. Tzou and H.-J. Hsu, "FPGA realization of space-vector PWM control IC for three-phase PWM inverters," *IEEE Trans. Power Electron.*, vol. 12, no. 6, pp. 953–963, Nov. 1997.
- [159] S.-L. Jung, M.-Y. Chang, J.-Y. Jyang, L.-C. Yeh, and Y.-Y. Tzou, "Design and implementation of an FPGA-based control IC for ac-voltage regulation," *IEEE Trans. Power Electron.*, vol. 14, no. 3, pp. 522–532, May 1999.

- [160] M. Malinowski, W. Kolomyjski, M. Kazmierkowski, and S. Stynski, "DSP/FPGA control for variable-speed wind turbines," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, Paris, France, 7–10 Nov. 2006, pp. 5161–5166.
- [161] *MATLAB Getting Started Guide*, MathWorks, Natick, MA, 2008. [Online]. Available: <http://www.mathworks.com>
- [162] *Xilinx System Generator User Guide*, Xilinx, Inc., 2005. [Online]. Available: <http://www.xilinx.com>
- [163] *ISE 4 User Guide*, Xilinx, Inc., 2005. [Online]. Available: <http://www.xilinx.com>
- [164] O. López, J. Alvarez, J. Doval-Gandoy, and F. D. Freijedo, "Multilevel multiphase space vector PWM algorithm," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1933–1942, May 2008.
- [165] O. López, J. Alvarez, J. Doval-Gandoy, F. D. Freijedo, A. Nogueiras, and C. M. Peñalver, "Multilevel multiphase space vector PWM algorithm applied to three-phase converters," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, Orlando, FL, 10–13 Nov. 2008, pp. 3288–3293.
- [166] O. López, J. Alvarez, J. Doval-Gandoy, F. D. Freijedo, A. Lago, and C. M. Peñalver, "Four-dimensional space-vector pulse-width modulation algorithm for multilevel four-leg converters," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, Orlando, FL, 10–13 Nov. 2008, pp. 3250–3257.
- [167] *SimPowerSystems User's guide*, TransÉnergie Technologies Inc., Montreal, Canada, 2008, release 4. [Online]. Available: <http://www.mathworks.com/products/simpower/>
- [168] A. Iqbal and E. Levi, "Space vector modulation schemes for a five-phase voltage source inverter," in *Proc. European Conf. on Power Electronics and Applications EPE*, Dresden, Germany, 11–14 Sept. 2005.
- [169] N.-Y. Dai, M.-C. Wong, Y.-H. Chen, and Y.-D. Han, "A 3-D generalized direct PWM algorithm for multilevel converters," *IEEE Power Electron. Lett.*, vol. 3, no. 3, pp. 85–88, Sep. 2005.
- [170] R. Zhang, V. Prasad, D. Boroyevich, and F. Lee, "Three-dimensional space vector modulation for four-leg voltage-source converters," *IEEE Trans. Power Electron.*, vol. 17, no. 3, pp. 314–326, May. 2002.
- [171] R. Baghernejad, A. Bakhshai, and D. Yazdani, "A real-time neuro-computing three-dimensional space vector algorithm for three-phase four-leg converters," in *Proc. IEEE Industrial Electronics Society Conf. IECON*, Raleigh, NC, 6–10 Nov. 2005, pp. 1066–1070.
- [172] M. Perales, M. M. Prats, R. Portillo, J. L. Mora, J. I. Leon, and L. G. Franquelo, "Three-dimensional space vector modulation in abc coordinates for four-leg voltage source converters," *IEEE Power Electron. Lett.*, vol. 1, no. 4, pp. 104–109, Dec. 2003.

- [173] O. López, J. Alvarez, J. Doval-Gandoy, and F. D. Freijedo, “Multilevel multiphase space vector PWM algorithm with switching state redundancy,” *IEEE Trans. Ind. Electron.*, accepted for publication.
- [174] C. Turpin, P. Baudesson, F. Richardeau, F. Forest, and T. Meynard, “Fault management of multicell converters,” *IEEE Trans. Ind. Electron.*, vol. 49, no. 5, pp. 988–997, Oct. 2002.

Appendix A

Further Information About Testing Facilities

The new modulation algorithms have been implemented in a field-programmable gate array (FPGA) and tested in open loop with a multilevel voltage source inverter. Figure A.1 shows the general block diagram of all setups. The main components are the dSPACE platform that calculates the reference voltage, the FPGA that implements the algorithm under test, the optical link, the inverter and the load. Main characteristics of each part are given in the following.

A.1 dSPACE Platform

The dSPACE DS1103 PPC Controller Board is a rapid control prototyping system that can be mounted in a personal computer (PC). It is fully programmable from the Simulink block diagram environment. The main components and features of this board are the following:

- Master PowerPC running at 400 MHz.
- Slave digital signal processor (DSP) running at 20 MHz for special input/output (I/O) tasks.
- Two general purpose timers.
- 2 MB local static random access memory (RAM) as program memory.
- 128 MB global dynamic RAM for data storage and data exchange with the host.

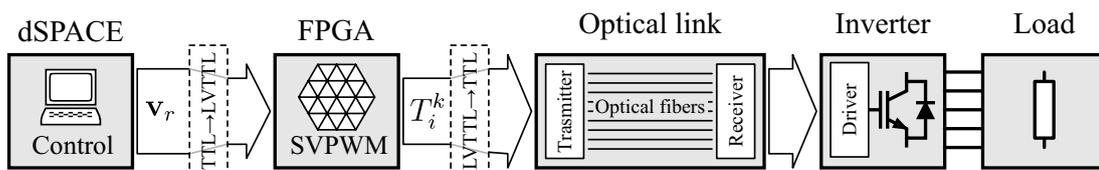


Figure A.1: Block diagram of the setups used for testing the algorithms.

- Interrupts by host PC, controller-area network (CAN), slave DSP, serial interface, incremental encoders and four external inputs (user interrupts).
- Four 16-bit analog-to-digital converter (ADC) and one 12-bit ADC, of four channels each.
- Eight 14-bit analog output channels.
- 24-bit individually programmable digital I/O channels, and 32-bit I/O digital port with four 8-bit channels.
- Incremental encoder interface with six digital and one analog input channels.
- Serial RS232 and RS422 interface.
- CAN Interface.

The PowerPC is a 604 Risc Microprocessor with the main following characteristics:

- Reduced instruction set microprocessor.
- 32-bit addressing.
- 8-, 16-, and 32-bit integer data types.
- 32- and 64-bit floating point data types.

The slave DSP subsystem is based on the Texas Instruments TMS320C240 DSP controller with the main following characteristics:

- 16-bit fixed point DSP.
- Event manager module with:
 - Twelve compare/pulse-width modulation (PWM) channels.
 - Three general purpose timers.
 - Three compare units with deadband.
 - Four capture units.
- 28 individually programmable, multiplexed I/O pins.
- Two 10-bit ADCs with 16 inputs.
- 64K×16-bit program memory and 28K×16-bit data memory.
- 16-bit dual-port memory for communication with the master processor.
- Several on-chip peripherals for motor/motion control applications.
- All on-chip peripherals accessible by the master processor.

The duty of the DS1103 board is to calculate the normalized reference vector \mathbf{v}_r and to send it to the FPGA in every switching period. The FPGA indicates the beginning of the switching period by activating an external interruption of the DS1103 board. Then, the inverter reference voltage for each leg V_r^k is calculated in open loop. Many dSPACE implementations have been carried out according to the number of inverter legs: one for the three-leg inverter, one for the four-leg inverter, and one for the five-leg inverter. In all cases, the reference voltage has a fundamental component of 50 Hz that can be either balanced or unbalanced. Harmonic injection is also possible by adding a third harmonic in the three- and four-leg cases, and by adding third and fifth harmonic in the five-leg case. After that, the reference voltage vector \mathbf{V}_r is normalized taking into account the voltage step of the inverter V_{dc} and sent back to the FPGA through the 32-bit digital I/O port.

A.2 Field-Programmable Gate Array

All modulation algorithms have been implemented in the Digilent S3 board. The main components and features of this board are:

- Hosts a Spartan-3 XC3S200 FPGA from Xilinx, which has
 - 4.320 logic cells, each constituted by two 16×1 lookup tables (LUTs) and two flip-flops.
 - Twelve 18×18 hardware multipliers.
 - Twelve 18 kb block random access memories BRAMs.
 - Four digital clock managers.
 - 173 user-defined I/O signals.
- 50 MHz crystal oscillator clock source.
- 2 Mb Xilinx XCF02S Platform Flash.
- Four-character, seven-segment light-emitting-diode (LED) display and eight individual LED outputs.
- Eight slide switches and four momentary-contact push button switches.
- Three 40-pin expansion connection ports (A1, A2 and B1).
- RS232 serial port.
- Mouse/keyboard port.

The S3 board gets the normalized reference vector \mathbf{v}_r from the dSPACE and calculates the trigger signals for every transistor of the inverter. Figure A.2 shows the S3 board together with three boards developed to adapt the low-voltage transistor-transistor logic (LVTTTL) signals used by the FPGA to the transistor-transistor logic (TTL) signals used by the rest of the hardware. Those boards are based on the 74LVC4245A integrated

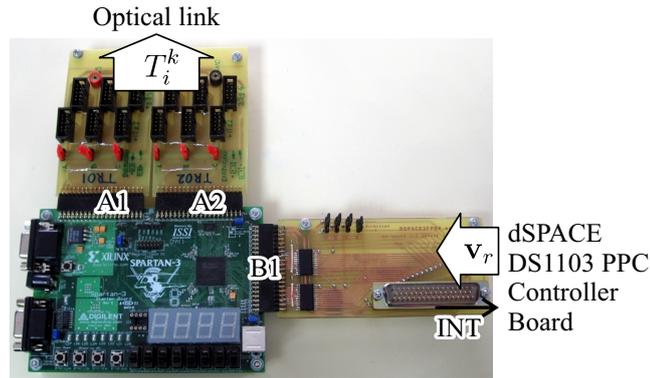


Figure A.2: S3 board with the TTL/LVTTL adaptation boards connected.

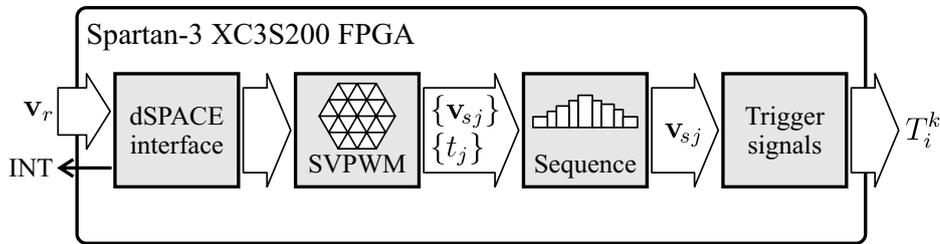


Figure A.3: Block diagram of the circuits implemented in the FPGA.

circuit, which is an octal dual-supply translating transceiver. The board attached to the expansion connector B1 adapts the signals between the DS1103 board and the FPGA. The two boards attached to the expansion connectors A1 and A2 translate the trigger signals into TTL levels used by the optical link.

Figure A.3 shows the block diagram of the main circuits implemented in the FPGA. The block **dSPACE interface** activates the user interruption signal of the DS1103 board at the beginning of every switching period, and later collect sthe normalize reference vector \mathbf{v}_r . The block **SVPWM** is the modulation algorithm itself that is being tested. It calculates the switching vector sequence $\{\mathbf{v}_{s,j}\}$ and the switching times $\{t_j\}$. The next block **Sequence** sequences the switching vectors in time arranging them symmetrically within the switching period. Figure A.4 shows the example of the sequence of process in the case of a five-vector sequence. The block **Trigger signals** translates the current switching vector $\mathbf{v}_{s,j}$ into the transistor gate signals of the converter. This block also adds the required deadtime to the complementary trigger signals.

A.3 Optical Link

The trigger signals from the modulator are transferred to the converter through a fiber optical link that provides galvanic isolation between control electronics and power electronics. The fiber optical link is realized with the components of the HFBR-0501 series from Agilent Technologies. It consists of many 1 mm plastic optical fiber cables, one for each trigger signal, with each having a HFBR-1251 transmitter and a HFBR-2521

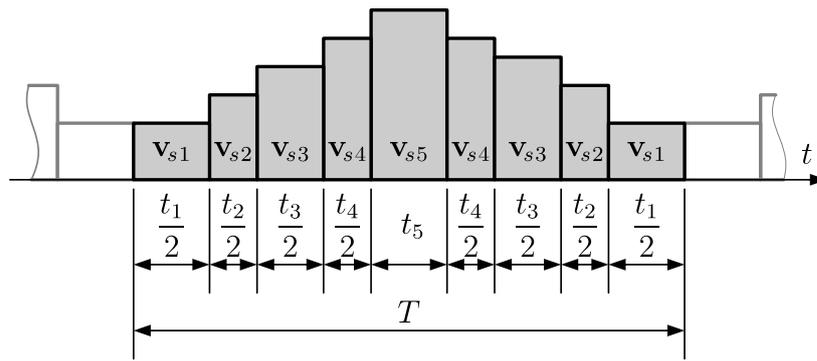


Figure A.4: Five-vector sequence example.

receiver. The electrical interface of the link is TTL compatible.

A.4 Inverters

Three inverters are used in tests: a three-leg inverter for three-phase systems, a four-leg inverter for three-phase systems with neutral leg and a five-leg inverter for five-phase systems.

A.4.1 Three-Leg Inverter

The three-leg inverter is the three-level neutral-point clamped (NPC) topology shown in Figure 3.2. It is made with components from Semikron. Each leg utilizes one module SKM 100GB123D, one module SKM 100GAL123D and one module SKM 100GAR123D. The two inner switches of the topology are realized with the SKM 100GB123 module, which includes two insulated-gate bipolar transistors (IGBTs). The upper IGBT and the upper clamping diode of each leg are realized with the SKM 100GAL123D module, which contains an IGBT and a diode properly connected for this purpose. The lower IGBT and the lower clamping diode are realized with the SKM 100GAR123D module, which also contains an IGBT and a diode. All IGBTs inside the modules have a blocking capability of 1200 V and a maximum conduction current of 100 A. Each IGBT is triggered with the SKHI 10 driver, which is TTL compatible. The dc bus uses four SKC2M2-45A-1 capacitors, two in parallel for each half of the dc bus. They are electrolytic capacitors with a capacitance of 2.2 mF.

In order to avoid voltage unbalance in the capacitors of the dc bus, two dc power supplies have been used to feed the dc bus as it is shown in Figure 3.2. The upper capacitors are supported by the SPS 800-4 power supply, and the lower capacitors are supported by the HP6053A power supply. The SPS 800-4 is 3.3 kW dc source from American Reliance that can give a constant voltage of up to 800 V. The HP6035A is a dc source from Hewlett Packard that provides up to 500 V and 5 A with a maximum power output of 1.0 kW.

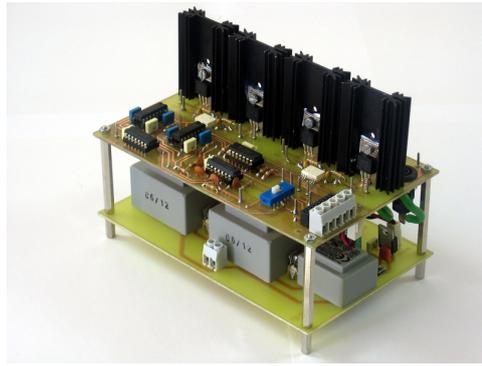


Figure A.5: Full-bridge module.

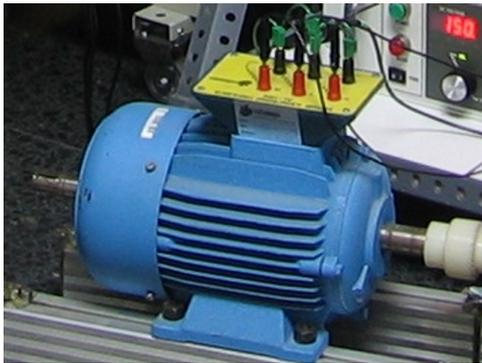
A.4.2 Four- and Five-Leg Inverters

The four- and five-leg inverters are a five-level converter in which each leg is a two-cell cascaded full-bridge inverter (see Figure 2.3). Each full-bridge inverter is a module made with four IRGB6B60KD power transistors. The IRGB6B60KD is a 600 V/7 A rated IGBTs from International Rectifier that includes an ultrafast soft recovery diode. Each pair of IGBT is driven by the dual-channel optocoupler HCPL315J integrated circuit from Agilent Technologies. The module includes also the logic circuits that allow to trigger the IGBTs using TTL-compatible logic levels. All dc sources needed to supply the full-bridge board directly from the ac mains are built-in as well. Figure A.5 shows a photograph of one module.

The dc link of each full-bridge inverter is made with a low cost dc source because of the high number of those devices needed; eight ones for the four-leg inverter and ten ones for the five-leg inverter. Two commercial dc sources are used in experiments: the Promax FA-325 and the Promax FAC-662B. The FA-325 dc source provides a variable voltage between 0 V and 30 V, with a current up to 2.5 A. The FAC-663B is a dual dc source of 0–30 V and 1.5 A that is configured in series to obtain 60 V at the output. A bank of capacitors of 940 μF is connected in parallel with each dc source to supply the high frequency current components to the full-bridge module. It is made with eight electrolytic capacitors of 470 μF each.

A.5 Loads

The modulation algorithms are tested in some cases with a static load and in some cases with a three- or a five-phase induction motor. The loads are always star connected with the neutral point isolated, except for the five-phase setup used to test the space vector pulse-width modulation (SVPWM) algorithm without redundancy, in which the neutral of the load was connected to the neutral of the inverter. The static load consists of a 100 Ω resistor connected in series with a 15 mH inductance in each phase. Precise data of the three-phase motor used in tests are not available. Figure A.6 shows a photograph and the nameplate of the three-phase induction motor, which has four poles and 1.0 kW nominal power. Figure A.7 shows the five-phase induction motor used in tests and its nameplate. This is a distributed-concentrated winding induction motor that was built by



(a) Photograph.



(b) Nameplate.

Figure A.6: Three-phase motor.



(a) Photograph.



(b) Nameplate.

Figure A.7: Five-phase motor.

rewinding the stator phases in the thirty stator slots of a 1.1 kW three-phase induction motor. It is a four pole machine, therefore it has 1.5 slots per phase per pole. The rotor is a squirrel cage with twenty two aluminium bars and short circuit rings.

A.6 Measurements

Measurements are done with the TPS2014 digital oscilloscope from Tektronix. This is a digital oscilloscope with four fully isolated channels with 100 MHz bandwidth (1 GS/s). It is equipped with the TPS2PWR1 power measurement and analysis software that allows, among others, to display individual harmonic levels of a waveform and to calculate the harmonic distortion up to the 50th harmonic.

The reference point for voltage measurements is the dc-bus midpoint in the case of the NPC inverter and the neutral point of the inverter in the cases of the cascaded full-bridge inverters. Filtered voltage waveforms have been obtained by means of a low-pass RC filter with $R = 8.2 \text{ k}\Omega$ and $C = 10 \text{ nF}$. This filter has a cut-off frequency of 1.9 kHz, thus it filters out all high frequency components due to the switching process ($f_s = 10 \text{ kHz}$).