Carrier Modulation of Four-Leg Matrix Converter Based on FPGA

Mei SU, Lixun XIA, Yao SUN, Hengsi QIN, Hongjun XIE
School of Information Science and Engineering, Central South University, Changsha, 410083, P.R. China

***Abstract*-Four-leg matrix converter usually requires complicate three-dimensional space vector modulation. Nevertheless the converter can be modulated in a carrier PWM approach much easier than the original. It turns out that this novel approach is less dependent on computing resources and allows an easier commutation design. Owing to the state-of-the-art Field Programmable Gate Array technology, the entire controller can be realized within a FPGA chip using programmable state machine technique. The supporting experiment has verified the validity of the proposed method and revealed that FPGA has been a cost-effective tool for power electronics applications.**

***Key Words*: Four-leg matrix converter, Carrier PWM, Programmable state machine, RISC, FPGA**

I. INTRODUCTION

Four-leg matrix converter features a neutral connection for three-phase four-wire systems. Therefore the converter has the ability to generate arbitrary line-to-neutral voltages, a merit when the load exhibits severe imbalance or nonlinearity [1-2]. Although a space-vector Pulse-Width Modulation algorithm has been proposed in [3], the need for three-dimensional reference synthesis poses a challenge to many engineers. To reduce the engineering efforts, a carrier PWM scheme has been proposed in this paper. The modulation algorithm is so simplified that it can be realized in one simple-structure Programmable State Machine (PSM) inside a FPGA chip instead of the bulky DSP-CPLD combination [4].

FPGA devices have been widely adopted in industrial applications for its reconfigurability, lower costs for non-recurring engineering and shorter time to market. Reference [5] demonstrated a SVPWM modulator in FPGA for ASIC purpose. In reference [6], a FPGA platform was used in control of a multilevel matrix converter. In this work, principles of carrier PWM of four-leg dual bridge matrix converter are introduced. In section III, the author elaborates on the control module design based on FPGA. A 2.2kW four-leg dual-bridge matrix converter prototype has been constructed to verify the effectiveness of the proposed scheme.

II. PRINCIPLES OF PROPOSED MODULATION METHOD

A. Topology

The topology of a four-leg dual-bridge matrix converter is illustrated in Fig.1. Semiconductor switches A_p B_p C_p A_n B_n C_n and neighboring diodes constitute the bidirectional rectifier stage called Current Source Bridge (CSB) [7]. The Voltage

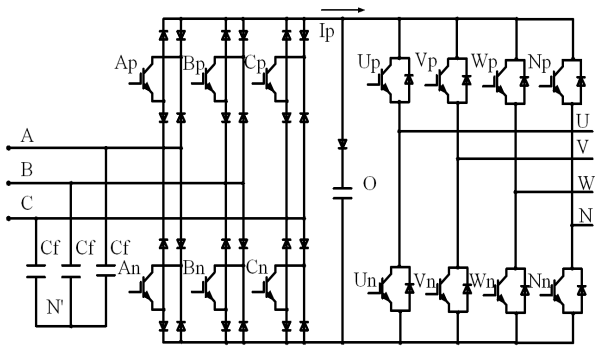

Fig. 1. Topology of the four-leg dual-bridge matrix converter.

Source Bridge (VSB), which contains the rest of the power switches, has an additional neutral leg N as a path for zero-sequence current flow.

B. Rectifier stage modulation

The PWM strategy for rectifier stage remains the same as traditional dual-bridge matrix converters [8]. A single period of the line-to-neutral voltage input can be divided into six sectors, separated by zero crossings of the three-phase input. The dc-link voltage of every modulation cycle is made up of two different line-to-line voltages, depending on which sector the modulation cycle resides. Consequently, a modulation cycle involves two rectifier swtiching states (Fig.3), with the corresponding duty ratios evaluated by the following equations.

$$d_{rec}^{I} = -U_1 / U_3$$
$$d_{rec}^{II} = -U_2 / U_3 \quad (1)$$

U_1 , U_2 and U_3 belong to line-to-neutral voltage input, while U_1 and U_2 are two with the identical polarities.

C. Inverter Stage Modulation

A striking advantage of carrier PWM is that the modulation of each inverter leg is independent of others. This makes the vector representation of the concurrent switching states unnecessary. The relationships between space vector and carrier modulation for three-leg inverter has been analyzed in [9]. For four-leg condition, modifications however are needed.

The voltage references can be evaluated from equations below.

$$Uuo = Uun + Uno$$
$$Uvo = Uvn + Uno \quad (2)$$
$$Uwo = Uwn + Uno$$

U_{un} U_{vn} and U_{wn} are the fundamental components of the line-to-neutral voltage output. U_{no} is the neutral leg (zero-sequence) voltage reference. U_{un}, U_{vn} and U_{wn} correspond to the reference vector's alpha-beta-plane projection in three-dimensional space, while U_{no} relates to the reference vector's projection on the gamma axis. Special care must be taken that to ensure linear modulation($|U_{io}| \leqslant 1$, $i \in \{u,v,w\}$), U_{no} must be selected within the range defined by (3) (4) and (5).

$$-U_{dc}/2 \leqslant U_{no} \leqslant U_{dc}/2 - max(U_{un}, U_{vn}, U_{wn}) \quad \text{if } min(U_{un}, U_{vn}, U_{wn}) > 0 \tag{3}$$

$$-U_{dc}/2 - min(U_{un}, U_{vn}, U_{wn}) \leqslant U_{no} \leqslant U_{dc}/2 \quad \text{if } max(U_{un}, U_{vn}, U_{wn}) < 0 \tag{4}$$

$$-U_{dc}/2 - min(U_{un}, U_{vn}, U_{wn}) \leqslant U_{no} \leqslant U_{dc}/2 - max(U_{un}, U_{vn}, U_{wn}) \quad \text{if } min(U_{un}, U_{vn}, U_{wn}) \leqslant 0 \text{ and } max(U_{un}, U_{vn}, U_{wn}) \geqslant 0 \tag{5}$$

The voltage references should in turn be normalized with regard to the dc-link voltages of each modulation cycle. Since every cycle is comprised of two rectifier-switching states, for both states the normalized voltage references are given below respectively.

$$U_{io}^{I} = 4U_{io}|U_1| / 3M^2$$
$$U_{io}^{II} = 4U_{io}|U_2| / 3M^2 \tag{6}$$

U_1 and U_2 are line-to-neutral voltage input that have the same polarities. M is the amplitude of the voltage input. Accordingly, the duty ratios of an inverter leg are obtained by shifting the normalized references into the range of [0, 1], as described in equation (7).

$$d_{inv}^{I} = (1 + U_{io}^{I}) / 2$$
$$d_{inv}^{II} = (1 + U_{io}^{II}) / 2 \tag{7}$$

III. Design of the Control Module

A. Overview

The whole control module was coded in VHDL to be synthesizable for FPGA. As demonstrated in Fig.2, a voltage-sampling device captures zero crossings of the voltage input to update the phase information for the Input State Machine (ISM). The ISM writes phase data to the PSM for duty-ratio generation. Commutation State Machine (CSM), which contains Rectifier State Machine (RSM) and Inverter Carrier Modulator (ICM), generates the switching patterns for CSB and VSB according to received duty ratios. The Global Counter (GC) defines the modulation cycle and synchronizes those blocks.

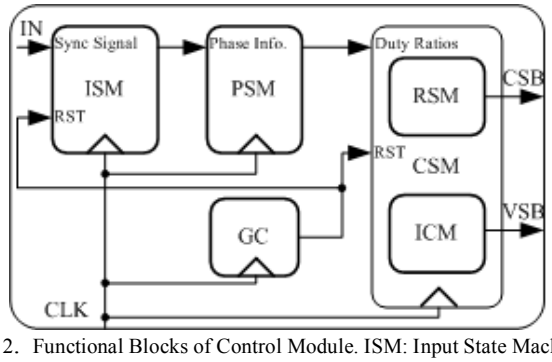

Fig. 2. Functional Blocks of Control Module. ISM: Input State Machine, PSM: Programmable State Machine; CSM: Commutation State Machine; GC: Global Counter; RSM: Rectifier State Machine; ICM: Inverter Carrier Modulator.

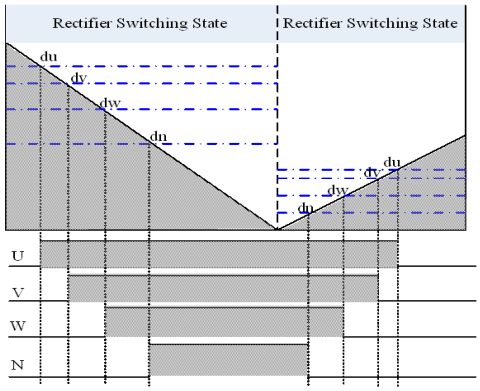

Fig. 3. A switching pattern realization for carrier PWM.

B. Commutation State Machine

In the RSM, the switching states of the CSB are determined by rectifier duty ratios and the global counter. While the ICM is made up of four comparators and loadable up-down counters functioning as carriers. Note that each inverter leg works independently, so packing the modulator for single leg in an entity and instantiating it four times in the VHDL source would save time and code space. A reference switching pattern realization is shown in Fig.3.

C. Programmable State Machine

The PSM (Fig.4) is a user-defined simplified RISC processor with its memory resources fully available on chip. It consumes less silicon resources than most commercial soft-core processors. As a processor with register architecture, the operational result of two operands (registers) is written back to

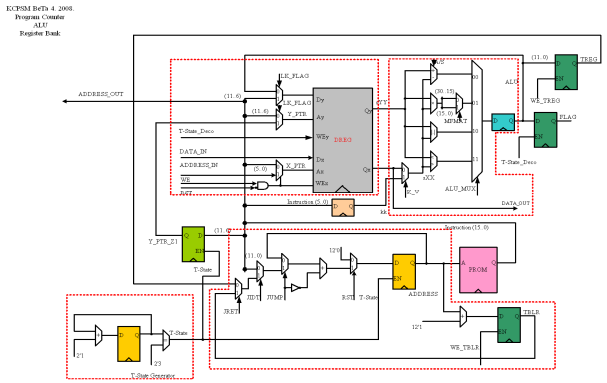

Fig. 4. Programmable State Machine RTL Diagram

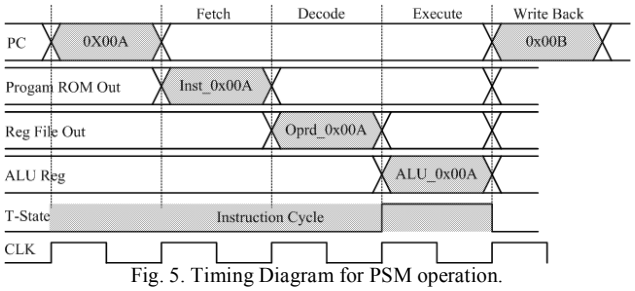

Fig. 5. Timing Diagram for PSM operation.

a third register in the register file. Hence, there are maximal three registers in a single instruction.

An analysis of the modulation algorithm in section II reveals that only 15 instructions are needed for data processing, listed in table I.

No pipeline structure is used for this PSM (Fig. 5). Every instruction cycle consists of four clock periods, fulfilling all the tasks: instruction fetch, instruction decode, execute and register write back. The pros include extremely simple RTL design, predictable execution time and no worry about data hazard.

For arithmetic instructions like addition, comparison, multiplication and absolute operations, FPGA vendors usually offer free IP cores for implementations. As for branching instructions, they have to refer to the flag register (containing the result of previous CMP instruction) to derive the next instruction address. Since the instruction width is 16 bit, to load a 16-bit constant in one instruction is impossible. Therefore, a decoding state machine (Fig.6) is designed to designate the

TABLE I
INSTRUCTION SET DESCRIPTION

	Instruction	Function	Operation(s)
Arith	ADD	16-bit addition	Reg1=Reg1+Reg2, PC++
	ADK	Add 6-bit constant	Reg1=Reg1+kk, PC++
	SUB	16-bit subtraction	Reg1=Reg1-Reg2, PC++
	SBK	Subtract Constant	Reg1=Reg1-kk, PC++
	MPL	Multiplication	Reg1=(Reg1 × Reg2)(30..15), PC++
	MPK	Multiply 6-bit Constant	Reg1=(Reg1 × Reg2)(15..0), PC++
	ABS	Absolute Value for 16-bit signed	Reg1=abs(Reg2), PC++
	CMP	Compare 16-bit signed values	Flag=ALU(1..0), PC++
Bran	JMP	Absolute Jump	PC=lkAA
	JLE	Jump if Flag⩽0	PC=lkAA if Flag ⩽0
	JEQ	Jump if Flag = 0	PC=lkAA if Flag = 0
I/O	OUT	Output Data	Out_enable_Flag=1, PC++
Tab Indx	LLK	Load 16-bit Constant	Reg1=lk, PC++
	TREG	Table Index	Treg=Reg1, PC++
	TBLR	Table Read	Reg1=*Treg, PC=Treg

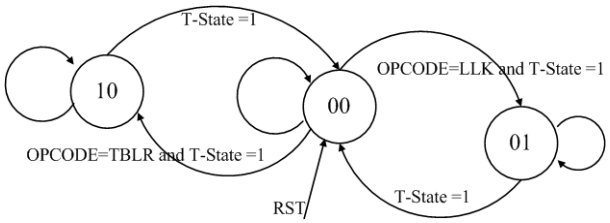

Fig. 6. Decoding State Machine

instruction right following the LLK instruction as a 16-bit constant rather than instruction with an opcode. This interlocked mechanism is applied on TBLR instruction as well, with the distinction that the element of a table is at remote address (the address of a table header plus an offset) rather than next to the current instruction.

The register file has 64 16-bit general-purpose registers, which are obtained as a dual-port RAM from on-chip distributed ram blocks. Any of the registers can be used as an accumulator, providing great flexibility for software coding. Actually, all variables have been stored in the registers and no scratchpad RAM was needed for this case.

D. Assembler Design

An assembler is also developed to help convert program into memory image files, and it mainly involves four steps. Firstly, a list of the variables is set up from the source program for indexing. The same procedure is applied again for the jumping addresses. Then the opcodes, variables and jumping addresses are all replaced with corresponding machine codes. Finally, the program image file is printed out by the rule of the specific memory initialization formats.

IV. VERIFICATION

Owing to the embedded multipliers and distributed RAM resources of the FPGA. The control module has been synthesized in QuartusII 5.1 consuming only 726 logic cells with a maximal frequency of 90MHz. The RTL description file has been verified in both ModelSim SE 6 and QuartusII 5.1 for functional and timing validity, as displayed in Fig.7 and Fig.8.

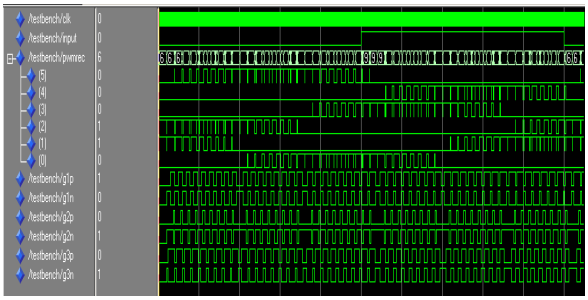

Fig. 7. Functional Verification For Control Module in ModelSim SE 6: input: signal from sampling board; pwmrec: gate signals for CSB IGBT drives; g1-g3: gate signals for VSB (signals for neutral leg not included).

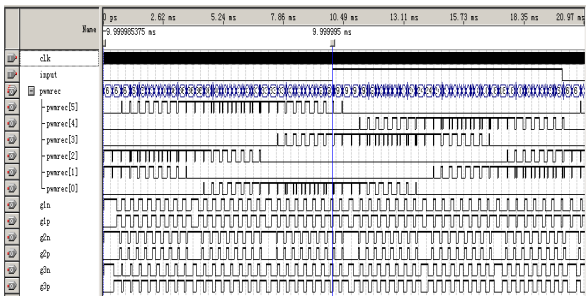

Fig. 8. Timing Verification For Control Module in QuartusII 5.1: input: input signal from sampling board; pwmrec: gate signals for CSB IGBT drives; g1-g3: gate signals for VSB IGBT drives(neutral leg not included).

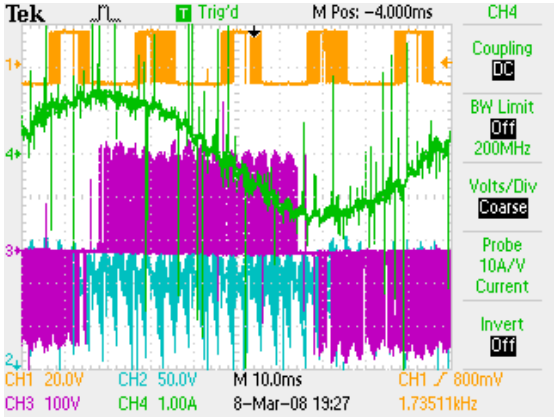

Fig. 9. Experiment Waveforms for 160V line-to-line voltage input(from top to bottom): drive-out signal for the CSB; phase current of the electric machine; output line-to-line voltage; dc-link voltage.

The experiment platform is a four-leg dual-bridge matrix converter driving a 2.2kW three-phase induction motor. The line-to-line voltage input is 160V/50Hz and the voltage transfer ratio 0.6. Modulation frequency is 2.5 kHz and the output frequency is 10Hz. The control module, downloaded to an EP2C8T144C8 experiment board clocked by a 40MHz oscillator, utilizes less than 10% of the total logic resources. The algorithm for the PSM costs only one eighth of the on-chip program space with an execution time of 22.6 us.

As the electric machine is three-phase symmetrical type, the voltage references are given as three balanced sinusoidal signals with the neutral reference zero. Therefore, the system operates in an open loop status with no dynamic compensator involved (Fig.9). A dynamic compensation algorithm and closed loop for output voltages are needed if nonlinear or imbalanced loads are present.

V. Conclusion

In this work, the author has demonstrated a novel carrier-PWM scheme for four-leg dual bridge matrix converter. The proposed algorithm is easy to implement and enables a simple commutation design. Meanwhile, the use of programmable state machine for system-on-chip design provides a cost-effective solution for power electronics applications.

Future work may include the realization of a dynamic compensation algorithm in the PSM for closed-loop operation when nonlinear and imbalanced loads are mounted.

Acknowledgment

The author would like to thank the National Natural Science Foundation (NNSF 60674065) for the financial support.

References

[1] Mason N.J., Wheeler P.W., Clare J.C., "Space Vector Modulation for a 4-Leg Matrix Converter, Power Electronics Specialists Conference", 2005. PESC apos;05. IEEE 36th, Volume , Issue , 11-14 Sept. 2005 Page(s): 31 – 38

[2] Yue Fan, Wheeler Patrick W. et al, "Indirect Space Vector Modulation for a 4-Leg Matrix Converter", Power Electronics Specialists Conference, 2007. PESC 2007. IEEE Volume , Issue , 17-21 June 2007 Page(s):639 – 645

[3] Yue Fan, Wheeler, Patrick W., Clare Jon C. "A Novel Four-leg Matrix Converter", IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference on, Volume , Issue , Nov. 2006 Page(s):2694 – 2699

[4] Mei SU,Lixun XIA,Yao SUN et al."Design and Realization of Carrier-Wave Modulation of Four-Leg Indirect Matrix Converter", Chinese Power Electonics, Vol42, No.1, pp35-37, Jan. 2008, in Chinese.

[5] Ying-Yu Tzou,Hau-Jean Hsu, "FPGA realization of space-vector PWM control IC for three-phase PWMinverters", Power Electronics, IEEE Transactions on, Vol. 12, Issue 6, Nov. 1997 Page(s):953 - 963

[6] R. Erickson, S. Angkititrakul, and K. Almazeedi, "A New Family of Multilevel Matrix Converters for Wind Power Applications", Subcontract Report NREL/SR-500-40051, Dec. 2006.

[7] Bingsen Wang, Giri Venkataramanan, "A Carrier Based PWM Algorithm for Indirect Matrix Converters", 37th IEEE Power Electronics Specialists Conference, June 18 - 22, 2006, Jeju, Korea

[8] J.W. Kolar, M. Baumann, F. Schafmeister, H.Ertl, "Novel Three Phase AC-DC-AC Sparse Matrix Converter", Proc. of APEC '02, Vol.2, pp. 777-791, 2002.

[9] Keliang Zhou and Danwei Wang, "Relationship Between Space-Vector Modulation and Three-Phase Carrier-Based PWM: A Comprehensive Analysis", IEEE Transactions on Industrial Electronics, Vol.49, No.1, Feb. 2002.

[10] Xilinx Co, "XAPP213: 8-bit Microcontroller for Virtex Devices", Feb. 2002.