



HAL
open science

Implementation of a Peak-Current-Control Algorithm Within a Field-Programmable Gate Array

Martin Aimé, Guillaume Gateau, Thierry A. Meynard

► **To cite this version:**

Martin Aimé, Guillaume Gateau, Thierry A. Meynard. Implementation of a Peak-Current-Control Algorithm Within a Field-Programmable Gate Array. IEEE Transactions on Power Electronics, 2007, 54 (1), pp.406-418. <10.1109/ISIE.2004.1571940>. <hal-03796399>

HAL Id: hal-03796399

<https://hal.science/hal-03796399v1>

Submitted on 8 Feb 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC 4.0 - Attribution - Non-commercial use - International License

Implementation of a Peak Current Control Algorithm within a Field Programmable Gate Array

Martin AIMÉ, Guillaume GATEAU, Thierry MEYNARD.

Laboratoire d'Electrotechnique et d'Electronique Industrielle

Unité Mixte de Recherche INPT-ENSEEIH / CNRS

BP 7122 - 2 rue Camichel - 31071 TOULOUSE Cedex 7 - France

e-mail : leei@leei.enseeiht.fr

Abstract—The Peak Current Control is a strategy widely used to control devices such as forward converters or two-level choppers. This strategy provides an imposed switching frequency of the output voltage and good dynamic performances. This paper presents the application of this strategy to a multilevel voltage source converter. An algorithm controls the level of the output voltage in such a way that the output current follows its reference. The principle of operation is introduced, and the implementation of the algorithm within a FPGA is demonstrated. Finally, the experimental results are presented.

Index Terms—Peak Current Control, multi-cell converters, FPGA.

I. INTRODUCTION

A large number of applications require controlled alternative current sources, such as the harmonic current compensation or the supply of AC motors. For those applications, a voltage source converter can be connected to an inductance, and its output current must be controlled. Many control strategies have been developed to ensure the control of the output current. The first one can be the hysteresis control which imposes the load current ripple, but produces a non-controlled switching frequency. Many alternatives can be found with this strategy [1,2,5]. Some of them allow to set the switching frequency, by modulating the hysteresis bandwidth. Some others have been developed for controlling multilevel converters [2]. The second strategy is the Peak Current Control (P.C.C.), that enables a fixed switching frequency, but generates a static error. This strategy was only designed for two-level converters. In this paper, we generalize this strategy to multilevel flying capacitor converters (choppers or inverters). An algorithm selects the right level of the output voltage of the multilevel converter. An other one chooses the cell which has to commute, and generates the switching orders. The second part introduces these two algorithms. The third part shows the implementation of the algorithms within a Field Programmable Gate Array (FPGA). The fourth part presents the experimental results.

II. PRINCIPLE OF THE P.C.C STRATEGY APPLIED TO MULTI-CELL CONVERTERS

The multilevel control strategy presented in this paper is inspired by the peak current control in the case of a two-level chopper. In this case, the converter is composed of a power semiconductor which can be turned on or turned off.

A free wheel diode ensures the continuity of the load current when the controlled switch is off. The switching frequency is set by a fixed clock. Each rising edge of the clock signal turns on the power switch. The output voltage becomes high, therefore the load current (noted i_{load}) increases. When i_{load} reaches the reference current (i_{ref}), the power switch is turned off, and the load current decreases. A compensating ramp is added to i_{ref} to avoid instability of the output current for duty cycle values greater than 1/2 [3].

The Peak Current Control strategy can be generalized to multilevel converters, and in particular to multi-cell converters. In this case, two compensating ramps are required, as shown in Fig. 1.

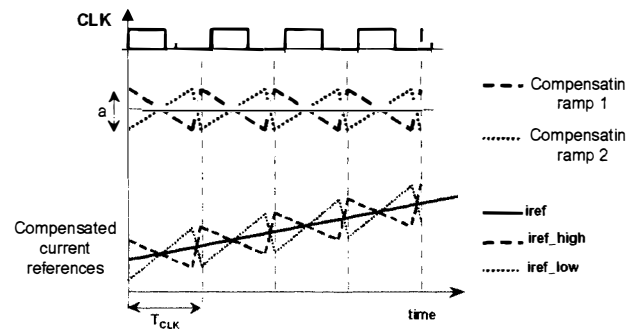


Fig. 1: Two compensating ramps are used when controlling a multilevel converter.

The two saw-teeth are synchronised by the same clock signal, and they are the opposite one of the other. Those ramps are intended to avoid limit cycles on the output voltage. The first one, added to the reference current (i_{ref}) gives the higher current reference (i_{ref_high}), whereas the second one gives the lower current reference (i_{ref_low}). In the case of a three-cell converter, the clock frequency is chosen equal to three times the desired switching frequency of the power semiconductors.

The peak current control strategy of the multilevel converter is divided into two blocks, as shown in Fig. 2. Within the first block, an algorithm determines the level of the output voltage needed to make i_{load} follow i_{ref} . Then, the second block generates the switching orders in such a way that V_{out} becomes exactly the same as its reference given by the first block. The two following paragraphs present the algorithms implemented within the two blocks.

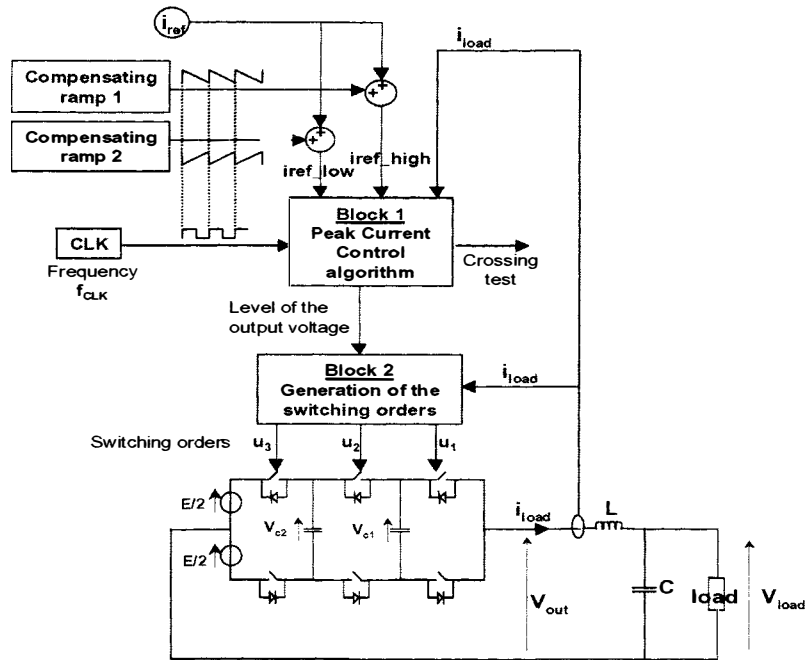


Fig. 2: Global view of the PCC strategy applied to a three-level flying capacitor inverter.

A. First block: Control algorithm of the output voltage

The algorithm within the first block controls the level of the output voltage. Its input signals are the clock, the measured load current and the two current references (i_{ref_low} and i_{ref_high}), (see Fig. 2). This algorithm can change the level of the output voltage either on a rising edge of the clock signal (CLK), or when i_{load} crosses a current reference (i_{ref_low} or i_{ref_high}). The beginning of each clock period is defined by the rising edge of CLK. At this time, three cases can occur:

- If i_{load} is lower than i_{ref_low} , and if the output voltage is not maximum, then the algorithm requests a one level increase of the output voltage.
- If i_{load} is higher than i_{ref_high} , and if the output voltage is not minimum, then the algorithm requests a one level decrease of the output voltage.
- If i_{load} is higher than i_{ref_low} and lower than i_{ref_high} , then the change of the output voltage depends on the previous edge of V_{out} . If this previous edge was falling, then a one level increase is applied upon V_{out} . If the previous edge was rising, then a one level decrease is applied.

When i_{load} crosses a current reference (i_{ref_low} or i_{ref_high}), the level of the output voltage can also be changed. However, some conditions are required. In particular, the PCC algorithm generates a signal called "crossing_test". This signal avoids multiple switches of V_{out} per clock period. It is set to zero on the beginning of each clock period. As long as crossing_test remains equal to zero, the level of V_{out} can be changed when i_{load} crosses a current reference. On the first change of V_{out} , the flag crossing_test is set to one. After that, the level of

the output voltage remains constant until the next rising edge of the clock signal.

When i_{load} crosses i_{ref_high} , three conditions are required for a one level decrease of the output voltage.

- The output voltage must not be at its minimum level.
- i_{load} has to cross i_{ref_high} upwards. Just before the crossing, the slope of i_{load} must be greater than the slope of i_{ref_high} .
- The flag crossing_test must be equal to zero.

When i_{load} crosses i_{ref_low} , three conditions are required for a one level increase of the output voltage.

- The output voltage must not be at its maximum level.
- i_{load} has to cross i_{ref_low} downwards. Just before the crossing, the slope of i_{load} must be lower than the slope of i_{ref_low} .
- The flag crossing_test must be equal to zero.

B. Second block: balancing of the floating voltages

The multi-cell structure is made of floating capacitors whose voltages have to be balanced. Fig. 3 shows the example of a three-cell chopper. In this example, the two floating voltages must be maintained at their correct values: V_{c1} should be equal to $V_{supply}/3$ and V_{c2} to $2*V_{supply}/3$. The algorithm within the second block of Fig. 2 generates the switching orders in such a way that V_{out} is exactly the same as its reference given by the first block. When a change of level is requested by block 1, the second algorithm selects among the three cells the one which has to switch. This choice is made according to the value of V_{c1} and V_{c2} . This algorithm intended to balance the floating voltages was introduced in [7].

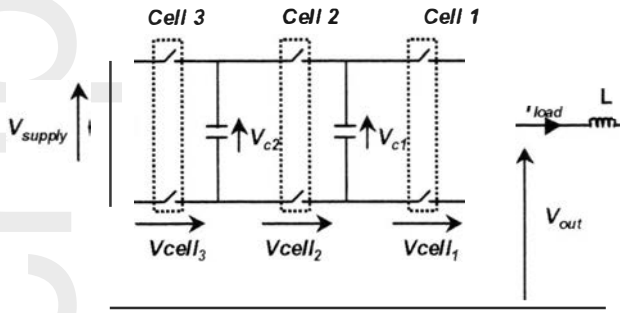


Fig. 3: Example of a multi-cell converter: the three-cell chopper.

We define each cell voltage as the difference between two consecutive floating voltages:

$$V_{cellk} = V_{ck} - V_{ck-1} \quad (\text{for } k=1, 2, 3.) \quad (1)$$

with $V_{c0} = 0$ and $V_{c3} = V_{supply}$

Balancing the floating voltages is equivalent to balance the cell voltages. In the case of a three-cell converter, we should have:

$$V_{cell1} = V_{cell2} = V_{cell3} = V_{supply}/3. \quad (2)$$

The table presented in Fig. 4 shows the selection criterion of the cell by the second algorithm [7].

Change on the level of V_{out}	Sign of the output current	Select among the cells...	...the one whose cell voltage is the ...
Increase (+1)	> 0	turned off	highest
Increase (+1)	< 0	turned off	lowest
Decrease (-1)	> 0	turned on	highest
Decrease (-1)	< 0	turned on	lowest

Fig. 4: Selection of the right cell when a change on the output voltage is required.

Thanks to the second algorithm, each cell voltage is rightly balanced at $V_{supply}/3$. Moreover, when the steady state is reached, the switches of V_{out} are shared among the three cells. As a consequence, the switching frequency of each cell is equal to the third of that of V_{out} .

C. Theoretical waveforms

The three-cell converter controlled by the PCC strategy was simulated. It appears that the output current can be guided either by i_{ref_high} or i_{ref_low} . When i_{load} is guided by i_{ref_high} , each clock period begins with a rising edge of the output voltage. Symmetrically, when i_{load} is guided by i_{ref_low} , each clock period begins with a falling edge of V_{out} . None of the two cases is favoured when controlling a multilevel converter. In that way, one can observe alternatively phases of i_{load} being guided by i_{ref_low} and phases of i_{load} being guided by i_{ref_high} . Of course, transitions between those two phases are possible, and they do occur when the load voltage (V_{load}) crosses the intermediate values of V_{out} . These transitions generate jumps on the load current, as shown on Fig. 5.

However, these jumps can be cancelled by adding another external loop to the structure shown on Fig. 2. Fig. 6 shows the simulated response of the controlled device to a step of the current reference. The response time depends on the amplitude of the step. In the simulation result presented on Fig. 6, this amplitude is great, and the response time is maximum. However, even in this case, the response time is lower than six clock periods. The theoretical waveforms shown on Fig. 6 demonstrate the good dynamic performances of the multicell converter controlled by the peak current control strategy. The experimental tests will confirm these good dynamic performances.

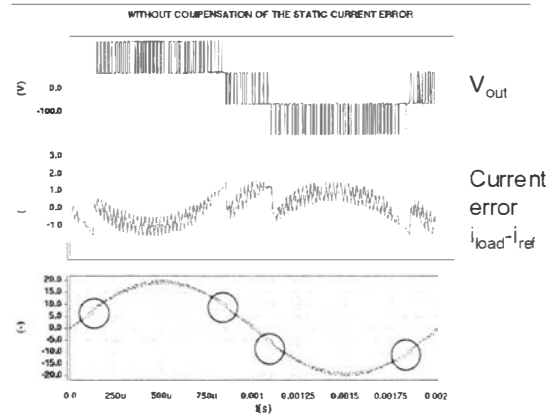


Fig. 5: Output voltage and load current obtained with a sinusoidal reference current. The jumps of the load current are located by circles.

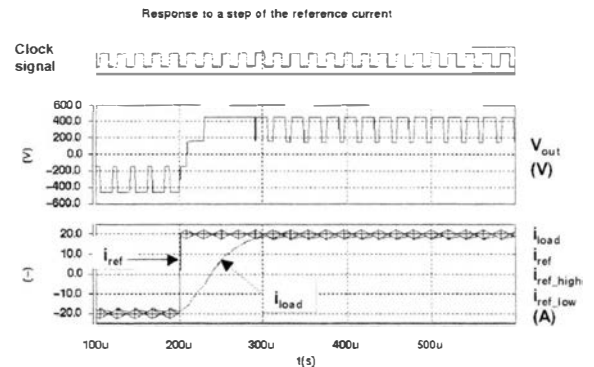


Fig. 6: Response to a step of the reference current.

III. IMPLEMENTATION OF THE CONTROL ALGORITHMS WITHIN THE FPGA

The PCC strategy was experimentally achieved, with a three-cell single phase chopper similar to the one presented on Fig. 3. Fig. 7 shows a global view of the device. The load current is measured by a current sensor, afterwards it is digitalized by a fast Analog to Digital Converter (named ADC on Fig. 7). The current reference is digitally generated by the Digital Signal Processor (DSP). A quartz oscillator at 16 MHz generates the clock signal both for the ADC and the FPGA.

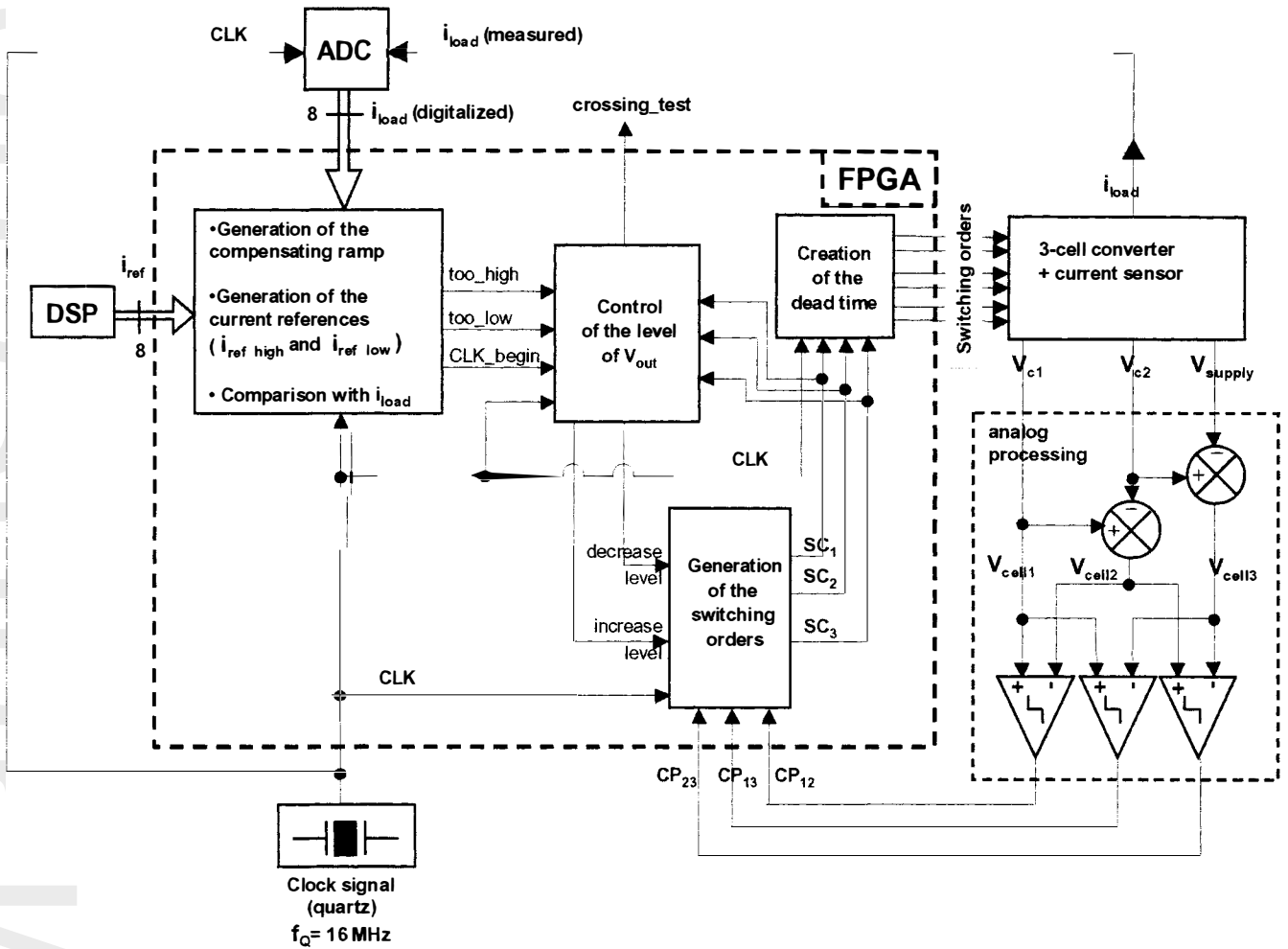


Fig. 7: Implementation of the Peak Current Control within a FPGA.

Three voltage sensors measure the supply voltage and the two floating voltages (V_{c1} and V_{c2}). The calculus of the three cell voltages (V_{cell1} , V_{cell2} and V_{cell3}) is achieved by two analog subtractors. The analog comparison of V_{cell1} , V_{cell2} and V_{cell3} gives the three logic signals CP_{12} , CP_{13} and CP_{23} .

For i and j from 1 to 3 (and $i \neq j$),

- If $V_{celli} > V_{cellj}$ then CP_{ij} is high.
- If $V_{celli} < V_{cellj}$ then CP_{ij} is low.

The signals CP_{12} , CP_{13} and CP_{23} are inputs of the FPGA, as well as the digitalized load current. The organization of the FPGA is divided into four blocks, as shown on Fig. 7.

The first block generates the compensating ramp and the current references i_{ref_low} and i_{ref_high} . It also compares the load current to i_{ref_low} and i_{ref_high} . The logical signals too_low and too_high are the result of this comparison. Fig. 8 shows in detail this first block. An up-down counter generates the compensating ramp signal, coded on 5 bits. This signal is added to the digitalized reference current to obtain i_{ref_high} . It is also subtracted from i_{ref} to obtain i_{ref_low} . To avoid any overflow during the addition and the subtraction, the ramp signal and the current reference must

both be converted into 9 bits signals. Consequently, i_{ref_high} and i_{ref_low} are also coded on 9 bits. The digitalized load current is then compared to i_{ref_high} and i_{ref_low} , giving the logical signals too_high and too_low . too_high is equal to zero so long as i_{load} is lower than i_{ref_high} , and too_high is equal to one otherwise. In the same way, too_low is equal to zero so long as i_{load} is greater than i_{ref_low} , and too_low is equal to one otherwise.

The first block also generates a pulse signal named 'CLK_begin'. This signal sets the switching frequency of the output voltage. CLK_begin is set to one each time the ramp signal reaches its maximum value (+15). All the rest of the time, CLK_begin is equal to zero. The period of CLK_begin is 256 times that of the quartz clock signal (that means the frequency of CLK_begin is 62.5 kHz).

The second block named 'control of the level of V_{out} ' on Fig. 7 contains the Peak Current Control algorithm presented in the second paragraph. Its input signals are the comparison signals too_high and too_low , the pulse signal CLK_begin, and the signals SC_1 , SC_2 and SC_3 which control the state of the 3 commutation cells of the converter. The second block is synchronized by the clock signal at 16 MHz. It generates two impulse signals noted

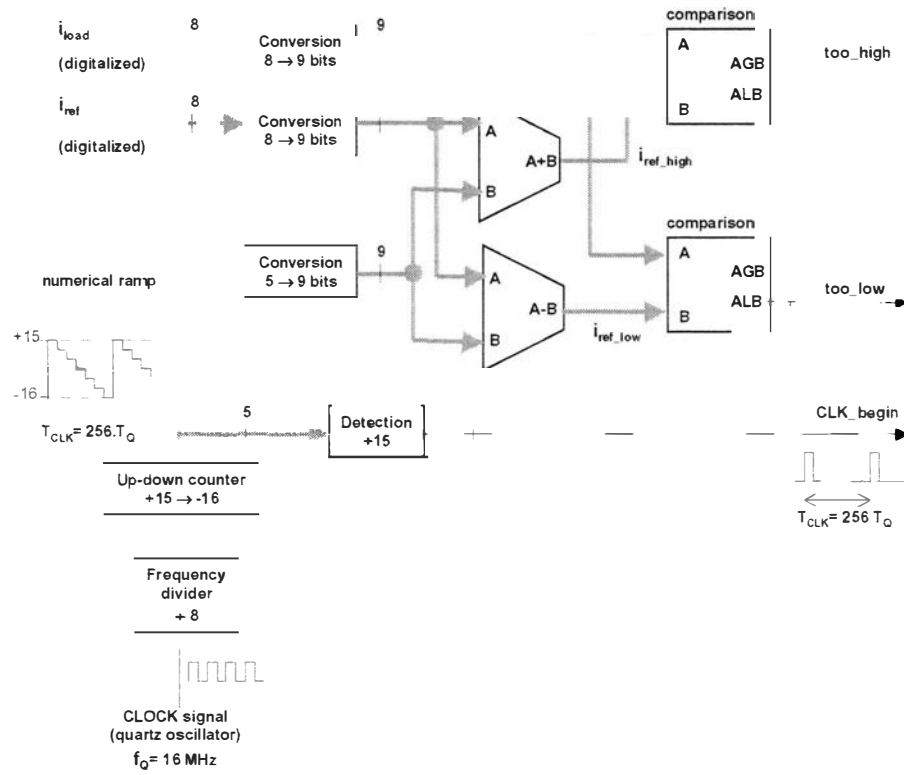


Fig. 8- Detail of the first block: generation of the current references i_{ref_low} and i_{ref_high} .

increase level and decrease level (see Fig. 7). These two signals are intended to control the changes of the output voltage level. Each pulse on increase level should produce a one level increase on the output voltage. On the other side, each pulse on decrease_level should cause a one level decrease of V_{out} . The duration of each pulse is equal to one period of the clock signal at 16 MHz. The second block also generates the signal crossing test introduced in paragraph II. Thanks to a Set-Reset flip-flop, each pulse on the signal CLK_begin resets $crossing_test$ to zero. When the conditions for changing the level are met, $crossing_test$ is set to one. At the same time, an impulse appears on one of the two signals $increase_level$ or $decrease_level$. Once the output level is changed, $increase_level$ and $decrease_level$ are reset to zero until the next pulse on CLK_begin . The conditions for changing the level of the output voltage are given by the algorithm presented in paragraph II.A. The detection of the rising edges of the logical signals too_high and too_low gives the instants at which the load current crosses i_{ref_high} and i_{ref_low} . Moreover, the PCC algorithm needs to know whether the output voltage level is maximum or minimum. This information is given by the three inputs SC_1 , SC_2 and SC_3 . Indeed, for $k=1,2,3$, SC_k controls the state of the k^{th} commutation cell (see Fig. 7). If $SC_k=0$, then the top switch of the k^{th} cell is turned off, and the bottom switch is turned on. On the contrary, if $SC_k=1$, then the top switch is 'on', and the bottom switch is 'off'. As a consequence, the minimum level of V_{out} is detected when the three signals SC_1 , SC_2

and SC_3 are equal to zero. In the same way, the maximum level of V_{out} is detected when SC_1 , SC_2 and SC_3 are equal to one.

The third block presented on Fig. 7 generates the switching orders for the three commutations cells (SC_1 , SC_2 and SC_3). When an impulse occurs on one of the two signals $increase_level$ or $decrease_level$, the third block selects the cell which has to commutate, according to the selection table introduced in Fig. 4. An impulse on $increase_level$ means that V_{out} has to be increased. So the algorithm chooses a cell whose state signal (SC_k) is equal to zero. On the other hand, on an impulse on $decrease_level$, the algorithm changes the state of a cell whose SC_k is equal to one. The selection table is entirely programmed using logical gates and RS flip-flops, but the implementation is too complex to be detailed here.

The fourth block named 'creation of the dead times' generates the switching orders for the multi-cell converter. Each commutation cell needs two signals to control the state of the two switches. The signal which controls the top switch should be complementary to the one which controls the bottom switch, except during the dead times. Indeed, the dead times are necessary to let the power switches commutate without creating any short-circuit. During the dead times, the two signals must be set to zero. To achieve this operation, the rising edges and the falling edges of each signal SC_k are detected within the fourth block. An up-down counter begins a countdown on each edge of SC_k . Two flip-flops generate the two control signals for each

commutation cell. The enable input of the two flip-flops is inhibited during the countdown, so the output signals are set to zero. When the countdown is finished, the two output signals become again complementary. We can choose the duration of the dead times by setting the initial value of the countdown.

IV. EXPERIMENTAL RESULTS

The control algorithms developed for the PCC strategy have been validated when operating the device. The experimental results have revealed several points:

- The active balancing of the floating voltages is very efficient.
- The PCC strategy generates jumps on the load current and a static error.
- The dynamic performances obtained with the PCC strategy are good.

Fig. 9 shows a waveform experimentally obtained with a low-frequency sinusoidal reference current, and Fig. 10 presents the response to a step of i_{ref} .

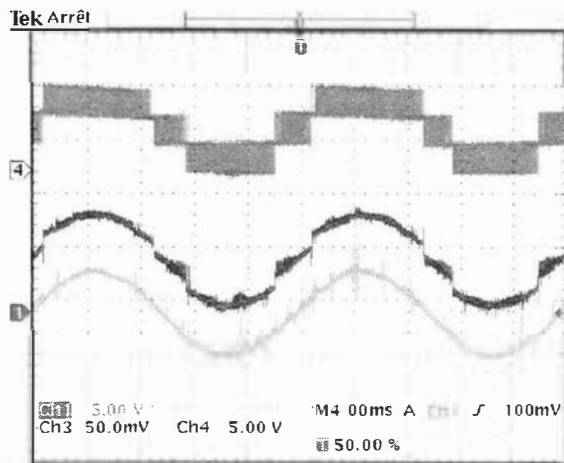


Fig. 9: Output voltage and load current obtained with a low-frequency sinusoidal reference current.

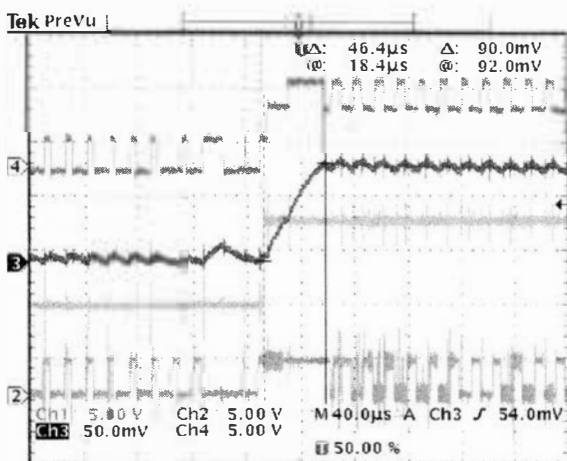


Fig. 10: Response to a step of the reference current.

The experimental results obtained with the PCC strategy are in conformity with the theoretical forecasts. The jumps

of the load current and the static error clearly appear on the waveforms presented on Fig. 9, as well as on the waveforms of Fig. 5. The response to a step of i_{ref} is very similar on Fig. 10 (experimental waveform) and on Fig. 5 (theoretical waveform). The only difference comes from the different values of the output inductance. The response time is shorter on Fig. 10 (3 clock periods) than on Fig. 6 (6 clock periods), because of a lower value of the inductance.

V. CONCLUSION

A new Peak Current Control strategy was tested by simulation and experimentally achieved. This strategy is dedicated to multilevel and multi-cell converters. The experimental achievement of the PCC strategy has shown that it is possible to implement complex control algorithms within a FPGA. In order to minimize the delay introduced by the sampling of i_{load} , such a control strategy has to be achieved using a very fast Analog to Digital Converter. As a consequence, the frequency of the quartz oscillator has to be chosen much higher than the switching frequency of V_{out} . Then we obtain a real time control of the load current and good dynamic performances.

A further development could consist in canceling the jumps of i_{load} and the static error introduced by the PCC strategy. Two external control loops could be added to that already implemented within the FPGA. So we could finally obtain a control strategy with an imposed switching frequency, and with no static error nor jumps of the load current.

VI. REFERENCES

- [1] T.A. Meynard, H. Foch, "Multi level conversion: high voltage choppers and voltage source inverters", PESC Conference, Toledo (Spain), 1992.
- [2] G.H. Bode and D.G. Holmes, "Load Independent Hysteresis Current Control of a Three Level Single Phase Inverter with Constant Switching Frequency", in proceedings of the IEEE conference, PESC 2001, 32nd Annual Power Electronic Specialist Conference, Vancouver (Canada), 17-21 June 2001.
- [3] M. Marchesoni, "High Performance Current Control Techniques for Applications to Multilevel High-Power Voltage Source Inverters", IEEE Trans. on Power Elect., vol 7, N°1, January 1992.
- [4] F. Dong Tan and R.D. Middlebrook, "A Unified Model for Current-Programmed Converters", IEEE Trans. on Power Elect., vol 10, N°4, July 1995.
- [5] A. Ras and F. Guinjoan, "Ramp-Synchronized, Sliding Mode Hybrid Control of Buck Converter", in proceedings of the IEEE conference, EPE'99, Lausanne (Switzerland), 7-9 September 1999.
- [6] G.H. Bode, D.N. Zmood, P.C. Loh and D.G. Holmes, "A Novel Hysteresis Current Controller for Multilevel Single Phase Voltage Source Inverters", in proceedings of the IEEE conference, PESC 2001, 32nd Annual Power Electronic Specialist Conference, Vancouver (Canada), 17-21 June 2001.
- [7] C.A. Martins, X. Roboam, T.A. Meynard, A.S. Carvalho, "DiCoIF: a DTC Strategy With Imposed Switching Frequency for Multi-Cell VSI fed Induction Motor Control", EPE'99, Lausanne (Switzerland), 7-9 September 1999.
- [8] M. Aimé, G. Gateau, T.A. Meynard, "A New Strategy based on Peak Current Control and Fixed Switching Frequency for a Multilevel Voltage Source Converter", EPE'03, Toulouse (France), 2-4 September 2003.
- [9] A. V. Anunciada, M.M. Silva, "A New Current Mode Control Process and Applications", IEEE Trans. on Power Electronics., vol 6, N°4, October 1991.