AN AREA-TIME EFFICIENT FPGA IMPLEMENTATION OF ONLINE FINITE-SET MODEL BASED PREDICTIVE CONTROLLERS FOR FLYING CAPACITOR INVERTERS

T. Vyncke, S. Thielemans, J. Melkebeek

Electrical Energy Laboratory (EELAB), Department of Electrical Energy, Systems and Automation (EESA), Ghent University (UGent), Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium
e-mail : Thomas.Vyncke@UGent.be

Abstract - Recently there has been an increase in the use of model-based predictive control (MBPC) for power-electronic converters. Especially for flying-capacitor multilevel converters (FCC) this offers an interesting possibility to simultaneously control output current and the capacitor voltages. The computational burden however is very high and often restrictive for a good implementation. In this paper a time and resource efficient design methodology is presented for the FPGA implementation of FCC MBPC. The control is fully implemented in programmable digital logic. Due to a parallel processing for the three converter phases and a fully pipelined calculation of the prediction stage an area-time efficient implementation is realized. Furthermore, this is achieved by using a high-level design tool. The implementation aspects for 3, 4 and 5-level FC inverters are discussed, with a focus on the 4-level case.

Keyword - MBPC, predictive control, FPGA implementation, flying-capacitor inverters, programmable digital hardware

1 INTRODUCTION

Multilevel converters were developed to meet a growing need for higher power converters. The series connection of switches allows a higher voltage handling and thus higher power rating for these converters while the lower voltage switches have reduced switching losses and can switch at a higher frequency. These topologies furthermore can apply intermediate voltage levels resulting in an output voltage with lower harmonic distortion. Due to several advantages over other multilevel topologies, flying capacitor (FC) converters have attracted a lot of interest, [1, 2]. The capacitor voltages of the FC converter need to be regulated, either passively by using natural balancing or actively. It has been shown that the passive control fails in certain circumstances, [2], resulting in a trend towards active control. The simultaneous control of the FC inverter output current and flying capacitor voltages is done preferably with a true multivariable control. Furthermore the switch state of the FC converter is inherently discrete. Finite set model based predictive controllers (MBPC) are an interesting option in such a case. Recently the use of MBPC for power converters has increased tremendously [1, 2, 3, 4]. Although this increase was enabled by the availability of high processing power, design choices have to be made. Those choices and their effect on the implementation in an FPGA, as well as an area-time efficient design of MBPC are presented in this paper.

2 MBPC FOR FC INVERTERS

The implementation aspects and design method MBPC of for n-level converters with $n = 3, 4, 5$ are discussed later on in this paper. The topology and model-based predictive control are discussed here, however only for the 3-level converter as this is the least complicated case.

2.1 FLYING CAPACITOR INVERTER TOPOLOGY

The topology of a three-phase, three-level FC converter is depicted in figure 1. It uses 2 pairs of complementary controlled switches, $(S_{1x}, \overline{S_{1x}})$ and $(S_{2x}, \overline{S_{2x}})$ per phase x, where $x = a, b, c$. These switches make it possible to connect the flying capacitors C_{1x} in series with the load (an RL series connection). An overview of the possible switch states and their resulting output voltage is given in table I. When the upper switch of the switch pair i is closed, S_{ix} is 1, otherwise S_{ix} is zero.
The series connection of the flying capacitor produces an intermediate output voltage. Because the flying capacitor is connected in series with the load, the voltage of the capacitor changes as the load current flows through the capacitor. The voltage of the flying capacitor C_1 in a three-level converter should always be kept at $V_{\mathrm{DC}}/2$. This choice provides optimal voltage rating of the switches as this only has to be $V_{\mathrm{DC}}/2$. Each phase has two switch states which produce the intermediate output voltage. This makes it possible to perform a correction of the capacitor

I. Switch states and the corresponding output voltage.

#	S_{1x}	S_{2x}	V_{xn} (if $V_{C_x} = V_{\mathrm{DC}}/2$)
1	0	0	$-V_{\mathrm{DC}}/2$
2	1	0	0
3	0	1	0
4	1	1	$V_{\mathrm{DC}}/2$

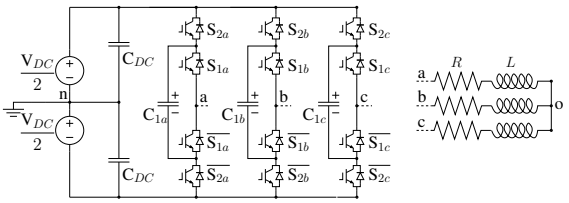

Fig. 1. 3-level flying capacitor converter topology.

voltage for every possible current direction and thus control the capacitor voltage. To avoid exposing the power switches to voltage overstress during start-up of the converter, a flying capacitor precharge technique should be applied. A self-precharge technique, as discussed in [5], is preferable.
The number of levels is increased by 1 by adding a pair of complementary controlled switches and a flying capacitor per phase.

2.2 MBPC ALGORITHM

The two control objectives for the model-based predictive control (MBPC) with multilevel converters are the tracking of the reference current and the balancing of the flying capacitor voltages. To this end the inputs for the MBPC algorithm are the reference values and the measurements of phase currents and flying capacitor voltages. The output of the algorithm is one of the possible switch states of the converter. As the switch state of the converter is maintained during an entire sample period, the output belongs to a finite set. The operation of the controller can be divided in: estimation, prediction and optimization.

Estimation
The controller operates in discrete time with a fixed sampling frequency. If a control ouput is applied at time instant k, the effects on the system states can be observed at time instant $k+1$. During the time interval $[k, k+1]$ the controller can no longer change the outcome of the states at $k+1$. Thus the control algorithm starts with calculating the system states at time instant $k+1$ based on the system model, the measurements (phase currents $\mathbf{i}_x^k$ with $x = a, b, c$ and the flying capacitor voltages $\mathbf{v}_{cx}^k$) and asserted control output (switch state $\mathbf{S}_{ix}^k$) at k.
The system model consists of the following expressions for the converter output voltages:

$$v_{xn}^{k+1} = \mathbf{S}_{2x}^k V_{DC} - (\mathbf{S}_{2x}^k - \mathbf{S}_{1x}^k)\mathbf{v}_{cx}^k \tag{1}$$

$$v_{on}^{k+1} = \frac{v_{an}^{k+1} + v_{bn}^{k+1} + v_{cn}^{k+1}}{3} \tag{2}$$

$$v_{xo}^{k+1} = v_{xn}^{k+1} + v_{on}^{k+1} \tag{3}$$

where bold variables denote measurements or actual states and the bus voltage V_{DC} is assumed known or measured. Also the output currents at $k+1$ have to be estimated. The expression for the current at $k+1$ consists of the free response and forced response:

$$i_x^{k+1} = e^{-\Delta\frac{R}{L}}\mathbf{i}_x^k + \frac{1 - e^{-\Delta\frac{R}{L}}}{R} v_{xo}^{k+1} \tag{4}$$

In equation (4) R and L are the resistive and inductive parts of the load respectively and Δ is the update period. The system model is finalized by the expressions for the flying capacitor voltages at the end of the k^{th} sampling period (i.e. at $k+1$). By using a trapezoidal discretization the following expression

$$v_{cx}^{k+1} = \mathbf{v}_{cx}^k + \frac{\Delta}{2C}(\mathbf{i}_x^k + i_x^{k+1})(\mathbf{S}_{2x}^k - \mathbf{S}_{1x}^k) \tag{5}$$

is obtained for a three-level converter, with C the capacitance of the flying capacitor.

Prediction
From $k+1$ on the controller can use any possible output during each sampling period to bring the controlled variables closer to their desired values. The controller thus calculates all possible control sequences over the time span from $k+1$ to $k+N_2$ based on the estimations at $k+1$. The prediction model essentially is the same as the estimation model, but the equations are evaluated one sample period later (for the first prediction step k and $k+1$ are augmented to $k+1$ and $k+2$ respectively). This results in the following set of equations to be evaluated for all possible switch states, for $i \in [1,\ N_2 - 1]$:

$$v_{xn}^{k+i+1} = S_{2x}^{k+i} V_{DC} - (S_{2x}^{k+i} - S_{1x}^{k+i}) v_{cx}^{k+i} \tag{6}$$

$$v_{on}^{k+i+1} = \frac{v_{an}^{k+i+1} + v_{bn}^{k+i+1} + v_{cn}^{k+i+1}}{3} \tag{7}$$

$$v_{xo}^{k+i+1} = v_{xn}^{k+i+1} + v_{on}^{k+i+1} \tag{8}$$

$$i_x^{k+i+1} = e^{-\Delta\frac{R}{L}} i_x^{k+i} + \frac{1 - e^{-\Delta\frac{R}{L}}}{R} v_{xo}^{k+i+1} \tag{9}$$

$$v_{cx}^{k+i+1} = v_{cx}^{k+i} + \frac{\Delta}{2C}(i_x^{k+i} + i_x^{k+i+1})(S_{2x}^{k+i} - S_{1x}^{k+i}) \tag{10}$$

As in the prediction phase all possible control actions have to be evaluated, the outcomes i_x^{k+2} to $i_x^{k+N_2}$ and v_{cx}^{k+2} to $v_{cx}^{k+N_2}$ of all possible switch state combinations are calculated and used during optimization.

Optimization
Once these possible control sequences have been calculated, the optimal sequence can be selected. The optimal sequence amongst the possible sequences is

found by evaluating all sequences in a cost function g which expresses the deviation of the controlled variables from their desired values. The sequence with minimal cost function is then selected and applied by the controller at time instant $k+1$. At this time $k+1$, the algorithm is started again, resulting in a so-called receding horizon.
When using a quadratic cost function (other cost functions can be used as well) the converter phase cost function g_x^k can be defined as

$$\begin{aligned} g_x^k &= W_{k+2}[(i_{x,r}^{k+2} - i_x^{k+2})^2 \\ &\quad + W_{v_c}(v_{c,r}^{k+2} - v_{cx}^{k+2})^2] \\ &\quad + \ldots \\ &\quad + W_{k+N_2-1}[(i_{x,r}^{k+N_2-1} - i_x^{k+N_2-1})^2 \\ &\quad + W_{v_c}(v_{c,r}^{k+N_2-1} - v_{cx}^{k+N_2-1})^2] \\ &\quad + W_{k+N_2}[(i_{x,r}^{k+N_2} - i_x^{k+N_2})^2 \\ &\quad + W_{v_c}(v_{c,r}^{k+N_2} - v_{cx}^{k+N_2})^2]. \end{aligned} \quad (11)$$

The weight factor W_{k+i} expresses the relative importance of the error in update period $k+i$ and W_{v_c} expresses the relative importance of an error in the flying capacitor voltage compared to an error in the output current. The weight factor values determine the control performance. The total cost function g^k is the sum of the converter phase cost functions.
When the prediction horizon is chosen to be 1 time-step, the converter phase cost function reduces to:

$$g_x^k = (i_{x,r}^{k+2} - i_x^{k+2})^2 + W_{v_c}(v_{c,r}^{k+2} - v_{cx}^{k+2})^2 \quad (12)$$

When the number of levels is increased by 1, which means the introduction of an extra flying capacitor, an additional term is added in (6), an equation similar to (10) is added to the set and the cost function is expanded with an extra weight factor and error term for the control of the extra flying capacitor voltage.

3 Simplified model versus model with coupled equations for FCC MBPC

The full model (equations (1)-(10)) of the flying capacitor converter (FCC) shows that the switch state of one phase influences the state variation in each phase through the load star point (equations (7)-(8)). As such the model for the three phases is fully coupled and for an m-phase, n-level converter $2^{m(n-1)}$ switch combinations have to be evaluated in the prediction step. To reduce the computational burden the model can be simplified when the interaction of the three phases through the load star point is ignored (assuming $v_{on} = 0$ in (8)), as is done in [1]. With this assumption the three phases are completely decoupled and only 2^{n-1} switch combinations have to be evaluated for each phase. With each phase cost function considered separately, this uncouples the switching decisions in the three phases. In order to asses whether this model simplification is desired, the impact on computational burden and control quality has to be evaluated.

3.1 Computational demands

In table II the number of prediction equation sets (the number of switch states to evaluate) are shown both for the coupled and uncoupled case, for $n = 3, 4, 5$. The comparison is made for two prediction steps as well, where the effect of the model simplification is very obvious. Clearly the benefits in computational demands are large, especially for higher n and higher prediction horizons (N_2).

II. Number of prediction equation sets

n	$N_2 - 1$	Coupled	Uncoupled
3	1	64	4
3	2	4096	16
4	1	512	8
4	2	262144	64
5	1	4096	16
5	2	16777216	256

3.2 Control quality difference

In [4] the effect of neglecting the star point voltage was investigated for a 3-level FCC. In figure 2 the mean square error (MSE) of the output current and capacitor voltage as well as the total error as a function of W_{v_c} are shown, both for the coupled and uncoupled case. Clearly a very low weight factor results in good current control quality but poorer FC control quality. The reverse is true for extremely large W_{vc}. A controller based on the full model outperforms a controller using the simplified model and has a much broader range of good values for W_{v_c} (i.e. a range where both current and voltage errors are low). As such it is, despite the computational advantages, not advisable to use the simplified model.
The expansion of the prediction horizon is also considered in [4]. It is shown that the combination of the simplified model and higher prediction horizons leads to a serious deterioration in the control quality. For the full model, the expansion of the prediction

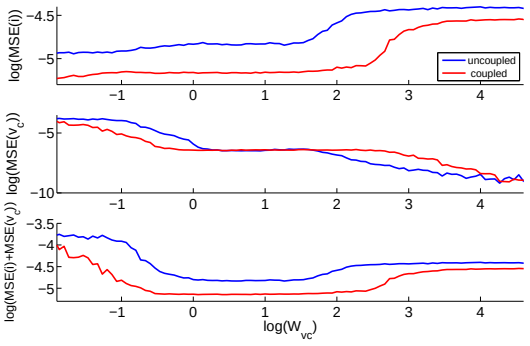

Fig. 2. Comparison of the control quality with simplified and full model for the current (top), capacitor voltage (middle) and the sum of both errors (bottom).

horizon can only be advisable if the cost function is expanded with control objectives other than current and voltage (e.g. related to EMC aspects of the converter). If no other objectives are added, there is a strong increase in calculations without control quality improvement.

3.3 SELECTED SCHEMES TO IMPLEMENT

As a result from the control quality analysis the most suitable candidate to implement is the full model (coupled case) with a prediction horizon of 1 update period. This means that between 64 ($n = 3$) and 4096 ($n = 5$) switch combinations have to be evaluated in the prediction phase. The prediction phase consists of the evaluation of the equations (6)-(10). In table III the number of calculations per switch combination evaluation (prediction and cost function) is shown, the operations are also listed by type. Clearly the execution of the prediction and optimization stage is a large computational burden, with a number of operations between 46 ($n = 3$) and 84 ($n = 5$) to evaluate for each switch state. The case of $n = 5$ has besides a considerably higher number of operations also much more switch states to evaluate. To compare the total computational burden, the last line of table III gives the total number of operations to be executed during the prediction and optimization phase.

III. Number of operations in the prediction and optimization step for each switch state possibility

	$n = 3$	$n = 4$	$n = 5$
addition	13	16	19
subtraction	15	24	33
constant mult.	11	14	17
multiplication	7	11	15
total per switch state	46	65	84
total all switch states	2944	33280	344064

The large number of calculations to perform clearly can be prohibitive for an implementation with reasonable update period on microcontrollers or DSPs, but with a proper design methodology it is very feasible on FPGA as discussed in the following section.

4 FPGA IMPLEMENTATION METHODOLOGY

During the implementation process of the coupled case MBPC for FCCs with $n = 3, 4, 5$ in the FPGA the aim is on the following goals:

- Obtained speed: to achieve an acceptable control quality, high update frequencies (short algorithm cycle times) are required. Here we use a 20 kHz update frequency (50μs cycle time).
- Used resources: in FPGAs the speed can be increased by paralleling processes, however the required resources (FPGA slices, multipliers) have to be available.
- Re-usability: a proper modular design allows for the re-use and straightforward adaptation of function blocks when increasing n.
- High-level configuration: to easily and quickly configure different versions for research.

The high-level configuration was done with the System Generator toolbox for Simulink/Matlab from Xilinx. It provides an environment to graphically build up the desired functionality in Simulink and to generate the FPGA bitstream, but allows also for the inclusion of low level VHDL code (e.g. for ADC-communication over SPI). Furthermore a modular build-up of the configuration comes naturally in this environment. In the FPGA implementation the operations are grouped in: estimation, prediction and optimization. This is shown in figure 3, where the entire block diagram is given. The other blocks are: measurements (for the ADC communication and scaling), reference generation (sinusoidal current reference and setpoint for capacitor voltages) and output (switch signal update and switch dead-time).

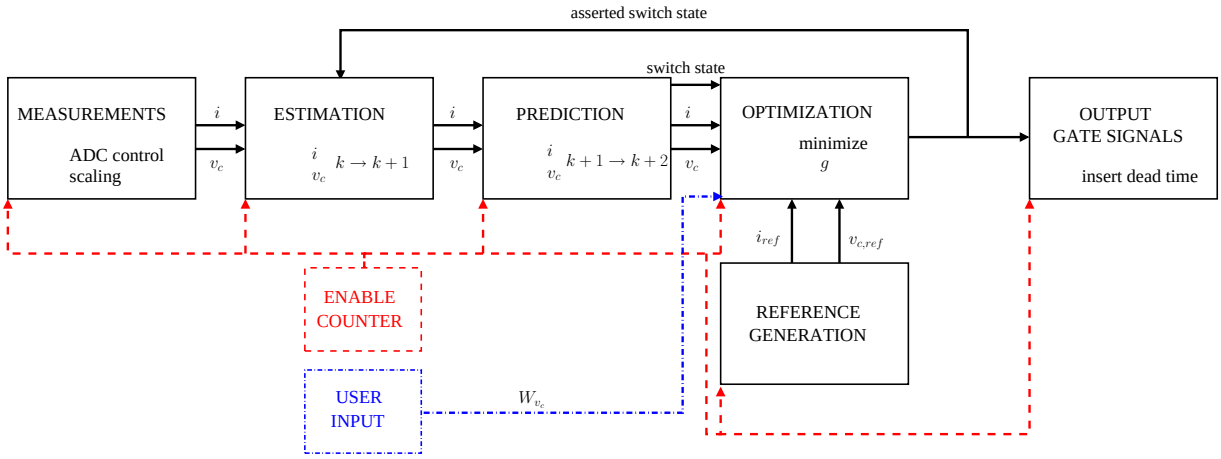

Fig. 3. Block scheme of MBPC with the constituting functions

Within each block signal latency is provided for correct timing of each operation. However each block is enabled separately, based on a central counter, as such the blocks are decoupled and reusable in other designs. This modular design makes the implementation very scalable to higher level FCCs, starting from $n = 3$ and expanding to $n = 4$ and $n = 5$.
As mentioned before the prediction and optimization blocks form the largest computational burden. As such the prediction and optimization are the main core of the implementation and need to be well designed, keeping calculation time and resources in check. This is done by exploiting the FPGA advantages. To improve speed the calculations are done in parallel for the three phases. Also the calculations for the cost function terms are done in parallel. The total latency to perform the calculations in the prediction phase in our design is 21 clock cycles. However paralleling the calculations for all switch combinations will not be possible, due to the limited FPGA resources, nor desirable. Thus, to find a good balance between speed and resources, the prediction block needs to calculate the results for all switch combinations sequentially. If for $n = 4$ the evaluation of the 512 switch combinations is performed with a calculation time of 21 clock cycles, the total calculation time would amount to 10752 clock cycles. This makes short cycle times impossible. For all switch combinations however the same equations need to be evaluated which allows, if a proper timing of the prediction block is provided, for a pipelined execution of the evaluation of all switch combinations. During the start-up phase of the pipeline no results are available at the output yet but a new switch state is fed into the pipeline at each clock cycle. During the steady-state of the pipeline a new switch state is fed into the pipeline and the results of the switch state loaded 21 clock cycles earlier becomes available at the output, for each clock cycle. During the last phase no new switch states are fed into the pipeline but a new result is still produced until the pipeline is 'empty'. With this fully pipelined prediction block all prediction equations for $n = 4$ take only 512+21=533 clock cycles. The optimization block is also fully pipelined and has a calculation latency of 2 clock cycles. This means that all prediction and optimization calculations are performed in 535 clock cycles for $n = 4$ (84 for $n = 3$ and 4120 for $n = 5$). Due to the fully pipelined design, scaling the prediction block to higher levels is easily done by duplicating the calculations for the flying capacitor and increasing the counter generating the switch combinations.

5 Design results: 4-level FCC MBPC

The FC converter is controlled with an Xilinx VirtexII-Pro FPGA (XUPV2P-30), clocked at 100 MHz. With a clock frequency of 100 MHz and an update period of 20 kHz there are 5000 clock cycles per update period. Figure 4 shows the timing diagram for the implemented 4-level FCC MBPC. The independent enabling can be seen. The values for the time intervals are given in table IV. The time needed for the measurements and reference generation t_{m+r} is 5 μs, although the ADCs allow a reduction to 1 μs. The time needed for the estimation t_{est} and output generation t_{out} is very small. Clearly the prediction and optimization phases use the most time (note however that the estimation and prediction phases use comparable amounts of resources). The time needed to calculate all 512 possible results t_{pred} and their cost function t_{opt} is only 535 clock cycles (5.35 μs), thus achieving about 6220 MOPS for the prediction core. Clearly the obtained speed is more than sufficient: only 21% of the cycle time is actually needed. This is mainly due to the parallel, pipelined prediction and optimization stage.

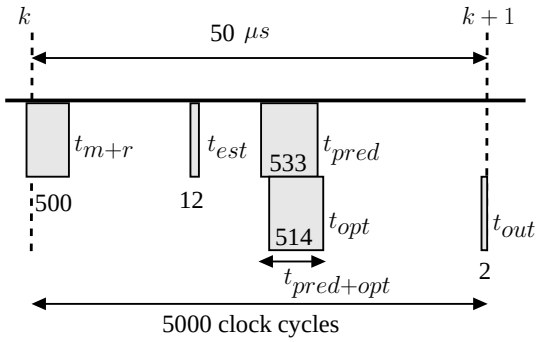

Fig. 4. Timing of 4-level FCC MBPC

IV. Timing for the 4-level FCC MBPC

	clk cycles	time (μs)	% cycle time
t_{m+r}	500	5.00	10.00%
t_{est}	12	0.12	0.24%
$t_{pred+opt}$	535	5.35	10.70%
t_{out}	2	0.02	0.04%
total	1049	10.49	20.98 %

In table V the used resources of the XUPV2P-30 are given. Only a fraction of the available resources are used. Even much cheaper FPGAs, such as the Spartan-3E-1200, can be used to implement this control (especially because System Generator allows to trade off between slices and multipliers). From the values for $n = 4$ it is clear that both time and resource constraints also allow the implementation of the $n = 5$ case. By using this design with the parallel and pipelined prediction core, actually a number of options are possible with either a higher level FCC or larger prediction horizon.

V. FPGA utilization for the 4-level FCC MBPC

	used/available	% used
Slices	5952/13696	43%
18x18 mults	38/136	27%
PowerPC	0/2	0%

6 EXPERIMENTAL RESULTS

The setup is a four-level flying capacitor converter constructed from in-house, half-bridge power electronic building blocks (PEBBs) as discussed in [6]. Each phase is equipped with a LEM LTS-25-NP current sensor to measure the output current. The appropriate signal conditioning and a 12bit ADC (National Semiconductor ADCS7476MSPS) provide a digitization on the PEBB. The flying capacitor voltages are measured in each phase leg with an instrumentation amplifier circuit and also digitized on the PEBB. The measured output currents and flying capacitor voltages are digitally transmitted to the FPGA, as such high resolution measurements are obtained. In the design of (FC) converters for model predictive control schemes the measurement quality is very important as it directly influences the obtained control quality. In certain applications it is desirable to avoid the measurements of the flying capacitor voltages and use an observer, [7].
In figure 5 a measurement is shown for the coupled control with $N_2 = 2$ for a 4-level FCC. The output current reference is a 50Hz sine with 2A peak value and the flying capacitor voltage references are chosen according a classical 3:2:1 ratio to the bus voltage V_{DC}. Clearly both the current and FC voltages are controlled close to their references. As the weight factor $W_{vc1} = 10$ is larger than the weight factor $W_{vc2} = 2.16$, v_{c1} stays closer to its reference.

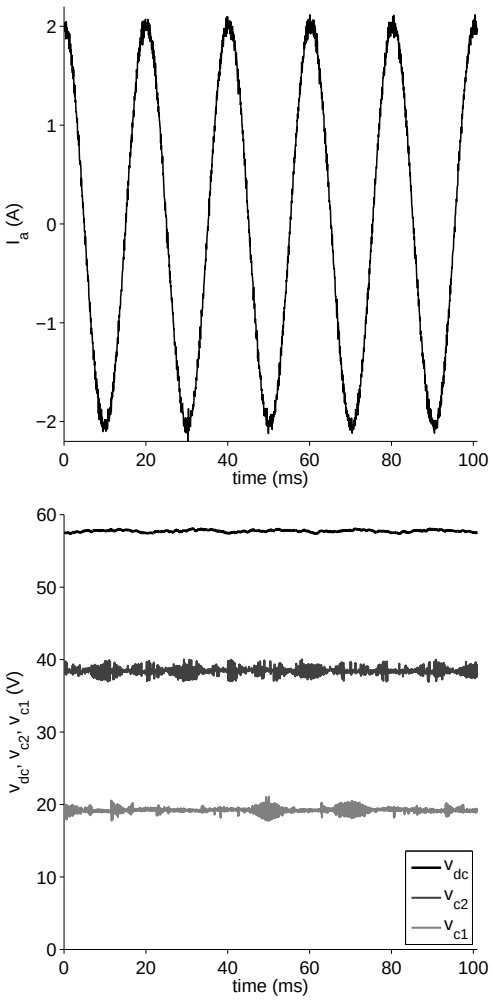

Fig. 5. Measurement of output current (top) and flying capacitor voltages (bottom) of one phase for 4-level FCC

7 CONCLUSIONS

In this paper a high level FPGA configuration design methodology for the model-based predictive control of flying-capacitor converters is discussed. Attention is paid to achieve high efficiency in time and resource utilization. By paralleling parts of the calculations and fully pipelining the prediction and optimization stages an area-time efficient design is obtained that is straightforward to scale to other numbers of levels or prediction horizons.

ACKNOWLEDGEMENT

This work was supported by the Fund of Scientific Research Flanders (FWO- Flanders) under Project G.0665.06 and by the Interuniversity Attraction Pole under Project (IUAP) P6/21. The work of T.J. Vyncke was supported by a Ph.D. fellowship from the FWO-Flanders.

REFERENCES

[1] P. Lezana, R. Aguilera, and D.E. Quevedo, "Model predictive control of an asymmetric flying capacitor converter", *IEEE Tran. on Ind. Electr.* 2009, Vol. 56 No. 6, pp. 1839–1846.

[2] F. Defay, A.M. Llor, and M. Fadel, "Direct Control Strategy for a Four-Level Three-Phase Flying-Capacitor Inverter", *IEEE Tran. on Ind. Electr.* 2010, Vol. 57 No. 7, pp. 2240-2248.

[3] S. Kouro, P. Cortés, R. Vargas, U. Ammann, and J. Rodríguez, "Model Predictive Control - A Simple and Powerful Method to Control Power Converters", *IEEE Tran. on Ind. Electr.* 2009, Vol. 56 No. 6, pp. 1826–1836.

[4] T. Vyncke, S. Thielemans, M. Jacxsens, and J. Melkebeek, "Analysis of design choices in model based predictive control of flying-capacitor inverters", accepted for publication in COMPEL

[5] S. Thielemans, A. Ruderman, and J. Melkebeek, "Self-precharge in single-leg flying capacitor converters", *35th Ann. conf. of the IEEE Industrial Electronics Society (IECON)* , Porto, Portugal, 03-05 Nov. 2009, pp. 809-814.

[6] K. De Gussemé, D. Van De Sype, J. Van Den Keybus, A. Van den Bossche, and J. Melkebeek, "Fully equipped half bridge building block for fast prototyping of switching power converters ", *IEEE 35th Ann. Power Electronics Specialists Conf. (PESC)*, Aachen, Germany, 20-25 Jun. 2004, pp. 1538-1544.

[7] G. Gateau, M. Fadel, P. Maussion, R. Bensaid, and T.A. Meynard "Multicell converters: active control and observation of flying-capacitor voltages", *IEEE Tran. on Ind. Electr.*, 2002, Vol. 49 No. 5, pp. 998-1008.