Concurrent and Simple Digital Controller of an AC/DC Converter With Power Factor Correction Based on an FPGA

Angel de Castro, Pablo Zumel, Oscar García, *Member, IEEE*, Teresa Riesgo, *Member, IEEE*, and Javier Uceda, *Senior Member, IEEE*

***Abstract*—Nowadays, most digital controls for power converters are based on DSPs. This paper presents a field programmable gate array (FPGA) based digital control for a power factor correction (PFC) flyback ac/dc converter. The main difference from DSP-based solutions is that FPGAs allow concurrent operation (simultaneous execution of all control procedures), enabling high performance and novel control methods. The control algorithm has been developed using a hardware description language (VHDL), which provides great flexibility and technology independence. The controller has been designed as simple as possible while maintaining good accuracy and dynamic response. Simulations and experimental results show the feasibility of the method, opening interesting possibilities in power converters control.**

***Index Terms*—AC–DC power conversion, adaptive control, concurrency control, custom hardware, digital control, field programmable gate arrays, flyback converter, power factor correction.**

I. Introduction

THE CONTROL of power converters is usually based on analog commercial solutions. In the case of power factor correction (PFC), the control is more complex because, in the general case, two loops are involved. Anyway, there are several analog commercial ICs solving this control problem. These circuits perform the basic control and their main advantages are low price and ease of use.

The increasing performance and cost reduction of digital circuits has made possible their application for power converters control [1]–[3]. With a few exceptions [4], [5], they are usually digital signal processor (DSP) based controllers that exploit their mathematical oriented resources. These controllers implement complex algorithms with many arithmetic operations. However, DSPs are not very common in high switching frequency or low cost applications. PFC is no exception to this tendency and a few digital controllers have arisen for this application [6]–[8]. Again, they are based on DSPs, exploiting their arithmetic resources.

The main limitation of DSPs is their sequential operation, that is, instructions are executed one after the other. However, DSPs have been adapted for power electronics applications adding peripherals such as PWM modules, general purpose timers and event interruption modules. These peripherals allow some concurrent operation, that is, several control tasks are performed simultaneously. Anyway, the simultaneous tasks must be very simple (PWM operation, timing $\cdots$), and they are not enough for a general concurrent operation structure like the one proposed here.

Following this tendency to use concurrent hardware for control purposes, we propose using a custom hardware solution, implemented in a field programmable gate array (FPGA) instead of a DSP, in order to exploit its concurrent operation [9]. All the internal logic elements of the FPGA, and therefore all the control procedures, are executed continuously and simultaneously. This method allows using high speed demanding algorithms, like the digital charge control proposed later in this paper for the current loop. This method would not be possible using a DSP.

The control algorithms have been developed in VHSIC hardware description language (VHDL) [10], [11]. This method is as flexible as any software solution, like developing the control algorithms in C-language for a DSP. Another important advantage of VHDL is that it is technology independent. The same algorithm can be synthesized into any FPGA and even has a possible direct path to a custom chip. In this way, the FPGA could be substituted by an Application Specific Integrated Circuit (ASIC), opening interesting possibilities in power systems in terms of performance and cost.

VHDL has also been used for modeling purposes. The power converter and the A/D converters (ADC) have been modeled in VHDL in order to simulate the whole system. These models have been developed as simple as possible in order to run long simulations in a reasonable time. In this way, both control loops, which differ greatly in their characteristic time, can be simulated simultaneously.

An FPGA-based solution changes the design point of view. Arithmetic operations should be kept to the minimum to optimize the required logic resources (silicon area). However, conditional execution (translated to *if* statements in VHDL) should be exploited because of its hardware oriented nature. As a result, an adaptive control is proposed for the voltage loop. Conditional execution is also specially suitable for implementing protections because they only actuate under specific conditions.

A disadvantage of FPGAs is their cost. They are slightly more expensive than DSPs and they have no A/D converters integrated, so their cost must be added. However, the high-speed

Manuscript received January 23, 2002; revised October 8, 2002. This work was presented in part at the Applied Power Electronics Conference (APEC'02), Dallas, TX, March 10–14, 2002. Recommended by Associate Editor S. B. Leeb.

The authors are with the E.T.S.I. Industriales, División de Ingeniería Electrónica (DIE), Universidad Politécnica de Madrid (UPM), Madrid 28006, Spain (e-mail: acastro@upmdie.upm.es; pzumel@upmdie.upm.es; oscar@upmdie.upm.es; teresa@upmdie.upm.es; uceda@upmdie.upm.es).

Digital Object Identifier 10.1109/TPEL.2002.807106

A/D converter used in the proposed control can not be substituted by those usually integrated in the DSPs. Anyway, the cost disadvantage would disappear substituting the FPGA by an ASIC, suitable for mass production.

As a conclusion, substituting the common DSP solutions by FPGA-based ones means a trade-off between the DSP capacity for arithmetic operations and the FPGA concurrency. In order to exploit the FPGA concurrency new control algorithms must be developed, because adapting the DSP ones to FPGAs would mean no special advantage. These new algorithms can be quite simple, like the digital charge control proposed, but they must be designed from the concurrency point of view.

The rest of the paper is organized as follows. Next section draws a general scheme of the application and its control. Sections III and IV explain in detail both implemented control loops, while Sections V and VI explain the digitally implemented protections and the modeling method. Finally, there are some experimental results and conclusions.

II. General Scheme

The objective of a PFC converter is to take energy from the ac source reducing the harmonic currents as much as possible, ideally working as a resistor emulator (sinusoidal current). Therefore, a loop must control the input current (I_{in}) making it proportional to the input voltage (V_{in}) through a previously defined conductance (G_{in}). This is the first loop: the current loop, which controls I_{in} according to V_{in}. However, this loop is not enough because the output voltage (V_{out}) has also to be controlled. Thus, the second loop controls V_{out} changing G_{in}, used in the first loop. So the second loop controls G_{in} according to V_{out} (voltage loop). An important difficulty for the voltage loop is that V_{out} has a 100 or 120 Hz ripple (twice the AC mains frequency). Traditional analog controls filter that ripple, reducing consequently the maximum voltage-loop bandwidth to about 20 Hz. Digital controls usually avoid that filter and the bandwidth reduction through some algorithm, like calculating the ripple and subtracting it from the measured V_{out} signal. We propose a much simpler method based on measuring the peak V_{out} values. This is explained in the "Voltage-Loop Controller" section.

Most PFC controls must observe at least three signals (V_{in}, I_{in}, V_{out}), as reflected in Fig. 1. In digital controllers, these signals are sampled through A/D converters. The only output of the controller is the signal that drives the switch in the power converter.

The proposed control also implements some protections that open the switch under specific circumstances, like overcurrent or overvoltage, or that avoid some malfunctions. These protections are implemented in the digital controller, so no external resources are necessary for the protections. This is explained in further detail in the "Protection Issues" section.

The block diagram of the complete system is outlined in Fig. 2, including the flyback converter and a scheme of the control. As it can be seen, the voltage-loop calculates the equivalent input conductance (G_{in}) that is used in the second part of the control, the current-loop. This second loop calculates the control signal sent to the switch in the power converter.

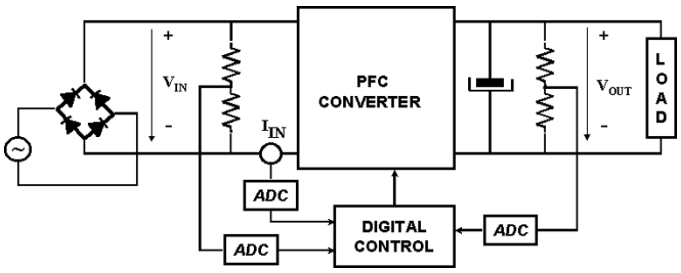

Fig. 1. General scheme of a PFC converter.

The current-loop controls the mean input current (I_{in}) making it proportional to the input voltage (V_{in}) through G_{in}, and therefore it achieves a high Power Factor (PF). The calculated control signal is not a duty cycle, but an "On/Off" signal that goes directly to the switch driver, avoiding a PWM block. Anyway, the switching frequency is kept constant. The digital protections are not shown for figure clearness. They actuate over the control signal before it is sent to the switch.

III. Current-Loop Controller

This loop controls the input current (I_{in}) in order to meet the PFC goal. Its purpose is to keep I_{in} proportional to V_{in}, in this case through the equivalent input conductance (G_{in}, the inverse of the equivalent input resistance R_{in}), calculated by the voltage-loop. The goal is to set the mean input current (over a switching cycle) according to the following formula:

$$I_{in-mean} = V_{in} \cdot G_{in}. \tag{1}$$

The proposed current-loop is the digital version of a charge control. The original charge control for PFC applications was proposed in [12], and it had not previously been implemented in a digital version because it is not feasible using a DSP. The working principle is to integrate I_{in} during a switching cycle (in the digital version, the integrator is substituted by a simple adder) until it reaches the target value, defined by the product of V_{in} by G_{in}, as reflected in (1). The digital version of the mean input current is the sum of the input current samples divided by the number of samples in a switching cycle ($N = 400$ in this case)

$$I_{in-mean} = \frac{\sum I_{in}}{N}. \tag{2}$$

This loop keeps the switch closed until the mean input current reaches the value defined by (1). Substituting (2) in (1), the condition for opening the switch becomes

$$\sum I_{in} \geq V_{in} \cdot G_{in} \cdot N. \tag{3}$$

In order to avoid one of the two multipliers in (3), G_{in} is scaled by the factor N (number of I_{in} samples in a switching cycle). Therefore, the variable which is in fact used is

$$G'_{in} = G_{in} \cdot N. \tag{4}$$

Finally, the algorithm implemented for the current-loop controller is

$$\sum I_{in} \geq V_{in} \cdot G'_{in}. \tag{5}$$

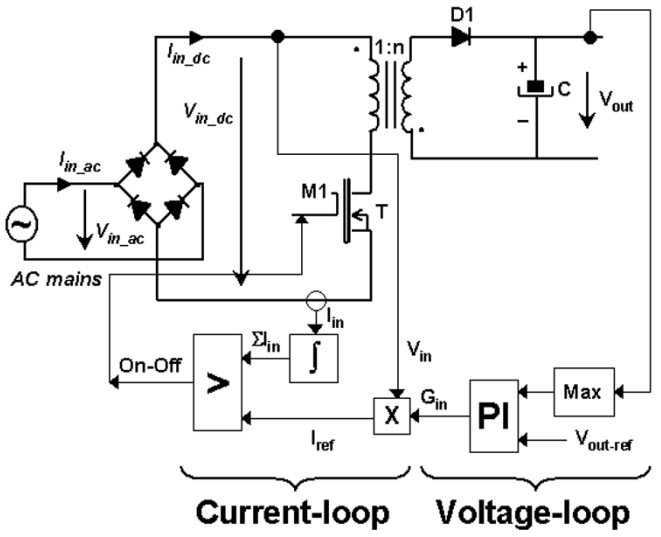

Fig. 2. Block diagram of the complete system.

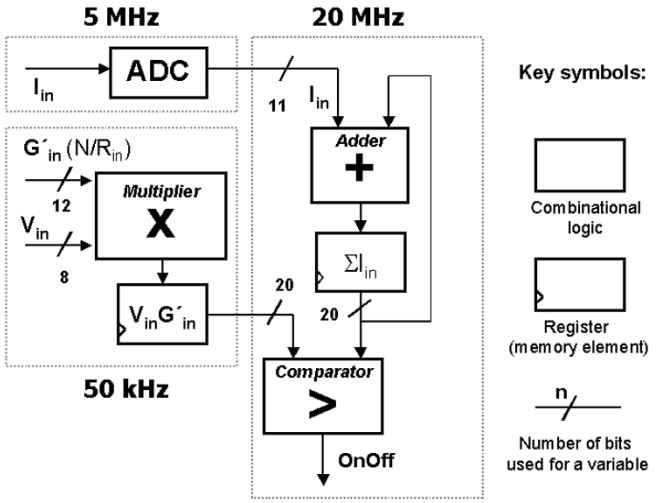

Fig. 3. Current-loop controller scheme.

The above equations are implemented as shown in Fig. 3. All the variables are represented in fixed-point format with the number of bits indicated in the figure. The sampled variables (I_{in}, V_{in}) representation depend on the A/D converter scale and the resistor divider, which can be chosen for an adequate representation. The accumulator (ΣI_{in}) uses more bits than I_{in}, assuring that overflow never takes place. The G'_{in} format is such that $V_{in} \cdot G'_{in}$ can be directly compared to ΣI_{in}.

The number of possible duty cycle values (equivalent to the PWM resolution) is the result of dividing the comparator frequency by the switching frequency (50 kHz). That is the reason for using a higher frequency for the adder and comparator (20 MHz) than for the A/D converter (5 MHz). In this way, the duty cycle resolution is 400 instead of 100, which would be the result if the comparator worked at the ADC frequency. This is done in order to avoid high frequency I_{in} oscillations [5], [13], as reflected in Fig. 4(a). Another solution would be to use a 20 MHz A/D converter, but the proposed solution obtains almost the same result with a slower, and therefore cheaper, ADC. The same I_{in} sample is added up to four times, but the error associated with this operation is small, as I_{in} has a slow evolution during a switching cycle as shown in Fig. 4(b). The mean input current is measured with about a 1.6% error, which has a negligible effect on the power factor.

The multiplier can work at the switching frequency (50 kHz) because its inputs are low frequency signals. The low multiplier frequency allows to implement it using fewer resources.

The proposed digital charge control would be difficult to use in a DSP because of the DSP sequential nature. At least, two instructions would be necessary after every sample (addition and comparison), making the process too slow for obtaining an acceptable duty cycle resolution. That is why DSP-controllers use more complex algorithms for the current loop, which are usually based on PID algorithms [6]–[8]. The main difference of the proposed current-loop controller is that no duty cycle is calculated but the switch is controlled directly, so the PWM module is embedded inherently in the proposed algorithm (no extra PWM is necessary). This means that the mean input current is controlled more accurately, as it is controlled at 20 MHz (every 50 ns) instead of at 50 kHz (every 20 μs, when a new duty cycle

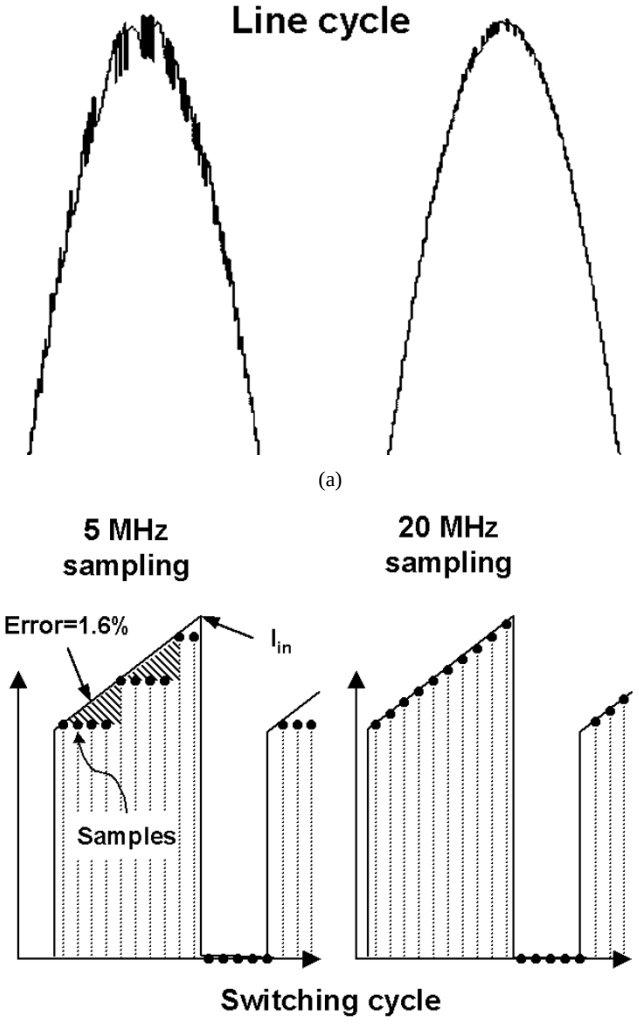

Fig. 4. (a) Mean input current for 100 (left) and 400 (right) possible "d" values. (b) Input current sampled at 5/20 MHz.

is calculated). The advantage of doing so is a higher PF. However, the PF obtained by the PID algorithms is high enough, so the actual advantage is a simpler algorithm for the same or even higher accuracy.

This loop, implemented as a digital charge control, is a good example of the FPGA advantages. It uses a high-speed algorithm in which all the resources are executed simultaneously. The hardware resources required for the algorithm implementation are quite small (just an adder, a comparator and a multiplier) when compared with the traditional PID algorithms. The main point, which makes it as accurate as the traditional algorithms, is concurrency: not many resources, but executed simultaneously.

Other advantages of this current-loop are valid operation for both CCM and DCM and no need for a converter model. This method is valid independently of the inductor (L) value.

IV. Voltage-Loop Controller

The previous loop makes I_{in} proportional to V_{in} through G'_{in}, therefore achieving power factor correction. However, that loop alone would leave the output voltage (V_{out}) uncontrolled. Therefore, the voltage-loop calculates G'_{in} in order to control V_{out}, and consequently the input power (P_{in}).

The whole control has only one output: the control signal sent to the switch in the power converter. This signal is calculated by the current-loop as explained before. The second loop (the voltage-loop) actuates changing the G'_{in} value that is sent to the current-loop. So one loop changes a parameter used in the other one. That is the way in which two different signals (I_{in} and V_{out}) are controlled with just one control signal.

A. Voltage-Loop Design

For clearness purposes, the following calculus are made with G_{in} instead of G'_{in}, which is the variable used. The only change is a scaling by the factor N as reflected in (4).

The control formula has been calculated without any transfer function. Furthermore, the controller has been calculated directly as a discrete controller, not as the digital redesign of an analog algorithm. The principle is just equaling the input and output power so V_{out} remains unchanged, based on the capacitor value, C. The power balance equation is the following one, where V_{out1} is the output voltage at the beginning of a rectified V_{in} cycle and V_{out2} at the end of the same cycle

$$\Delta E = (P_{in} - P_{out})T_{Vin} = \tfrac{1}{2}\, C \left(V_{out2}^2 - V_{out1}^2\right). \quad (6)$$

The input power depends on the previous value of the input conductance (G_{in1}), and the goal is to calculate the new G_{in} value (G_{in2}) so the input power is equal to the output power, restoring the balance

$$P_{in} = V_{in-rms}^2 \cdot G_{in1} \quad (7)$$

$$P_{out} = V_{in-rms}^2 \cdot G_{in2} \quad (8)$$

$$\Delta G_{in} = G_{in2} - G_{in1} = \frac{C\left(V_{out1}^2 - V_{out2}^2\right)}{2 \cdot V_{in-rms}^2 \cdot T_{vin}}. \quad (9)$$

The result is a second order equation, but the implemented algorithm uses the first order equivalent equation (which is the linearization around the working point)

$$\Delta G_{in} = \frac{C \cdot V_{out-ref}}{V_{in-rms}^2 \cdot T_{vin}} \cdot (V_{out-ref} - V_{out}). \quad (10)$$

T_{Vin} is the period of rectified V_{in} (10 ms), $V_{out-ref}$ is the reference value for V_{out} (48 V) and V_{in-rms} is also considered a constant (110 V), so V_{out} is the only variable. This algorithm is a discrete regulator that makes the difference between the new and the previous G_{in} values proportional to the error. This error is the difference between the V_{out} reference and measured values

$$E = V_{out-ref} - V_{out}. \quad (11)$$

B. Voltage-Loop Analysis

As stated before, the voltage-loop algorithm has been designed without calculating any transfer function, just equaling the input and output power. However, in order to analyze the control algorithm, its transfer function is calculated. As it is a digital regulator, the z-transform (discrete) transfer function is used. The corresponding z-transform formula is deduced as

$$[G_{in}]_n - [G_{in}]_{n-1} = K \cdot [E]_n \quad (12)$$

being constant K equal to

$$K = \frac{C \cdot V_{out-ref}}{V_{in-rms}^2 \cdot T_{Vin}}. \quad (13)$$

Therefore, the transfer function is

$$\frac{G_{in}(z)}{E(z)} = K \cdot \frac{z}{z-1}. \quad (14)$$

As the variable used is G'_{in} instead of G_{in}, the constant K includes the factor N in the actual algorithm.

Its physical implementation is reflected in Fig. 5, according to (10). This algorithm, if transformed to a continuous equation, would correspond to a PI control. Therefore, it has no steady-state error. It has also shown a good dynamic response for controlling V_{out}, recovering steady state within 3–4 cycles as shown in the experimental results.

The implemented loop is quite simple. All the variables are represented in fixed-point format using the number of bits indicated in Fig. 5. There is a subtractor for the difference between the reference and measured V_{out}. The result of the subtraction has to be multiplied by a constant (K). However, this constant can always be adjusted to a power of two (changing the internal representation of some signal), so the multiplication can be substituted by a shifting operation, which is much simpler. Finally, the value calculated by (10) represents the change from the previous value, so an adder is also necessary. Therefore, the only necessary resources are a subtractor, a shifter and an adder. In spite of this loop simplicity, it has worked fine attaining a good dynamic response and accuracy (as shown later in the "Experimental Results" section). This allows keeping the whole controller very simple, according to the design methodology here proposed.

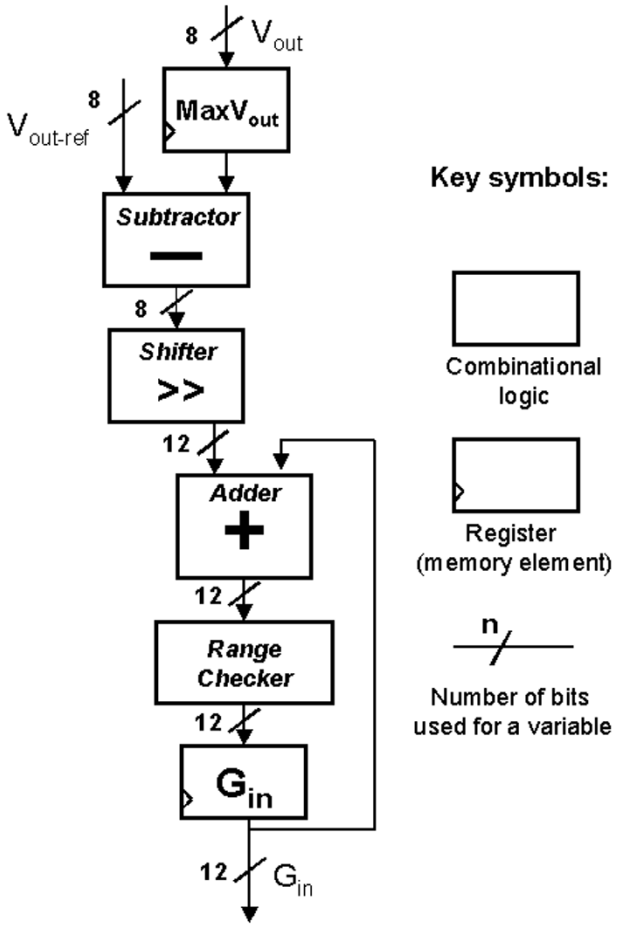

Fig. 5. Voltage-loop controller scheme.

In a power factor correction application, V_{out} has a 100 or 120 Hz ripple (twice the AC mains frequency). An advantage of using a digital control is avoiding the low-pass filter used for V_{out}, which analog controls use for canceling that ripple. This filter decreases dramatically the bandwidth, and therefore the dynamics. Some controls [7], [14] overcome this problem calculating the ripple and subtracting it from the measured V_{out}. However, this calculus is not a trivial one. We propose a simpler method that consists in using the maximum value of V_{out} in each rectified V_{in} cycle ($V_{out-\max}$). In this way, V_{out} does not need to be filtered and no further calculation is necessary. The main drawback of this method is that the changes in G_{in} happen only once every rectified V_{in} cycle (at 100 or 120 Hz), when a new $V_{out-\max}$ value is measured. In spite of it, a good dynamic response is achieved. Other disadvantage is that the mean V_{out} is not controlled, but the ripple peak value. The ripple amplitude is proportional to the load value, so the mean V_{out} changes slightly as the load changes (higher as the load decreases). This could be avoided by controlling the maximum and the minimum simultaneously, which would also improve the loop dynamics.

C. Adaptive Voltage-Loop Algorithm

The proposed voltage-loop is a PI control that has been calculated simplifying the power balance equation. As the algorithm has been calculated to recover steady state in a single V_{in} cycle,

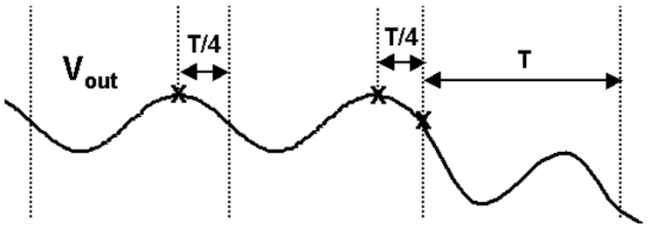

Fig. 6. V_{out} sampling delay due to the maximum technique.

its dynamics are quite demanding. Therefore, stability problems have to be analyzed.

In order to do so, a discrete root locus analysis (in the z-plane) is presented. For this analysis, the complete open loop transfer function is necessary, so the transfer function between G_{in} and V_{out} is shown (representing the output capacitor behavior based on the mean current balance during a rectified V_{in} cycle)

$$G(s) = \frac{V_{out}(s)}{G_{in}(s)} = \frac{(V_{in}^2/C)\cdot V_{out}}{s + (2/R)\cdot C} = \frac{B}{s+a}. \quad (15)$$

Equation (15) is the continuous transfer function, but for the discrete analysis its z-transform is used. In fact, it has been converted using the modified z-transform so we can take into account the delay introduced by the technique of sampling $V_{out-\max}$ during a rectified V_{in} cycle. The parameter "m" represents this delay, being 1 for no delay and 0 for a full period delay ($T = 10$ ms for 50 Hz mains). The delay is usually one fourth of the period, as shown in the left part of Fig. 6, so "m" is usually 0.75.

The modified z-transform of (15) is

$$G(z, m) = \frac{B}{a}\cdot\frac{(1-e^{-mTa})z - (e^{-aT} - e^{-mTa})}{z^2 - e^{-aT}z}. \quad (16)$$

The parameter "a" is highly dependent on the load value—R in (15)—so different transfer functions have to be analyzed for different loads. The root locus is based on the open transfer function, which is the result of multiplying (14) by (16). Fig. 7 shows the discrete root locus for nominal conditions (nominal load and $m = 0.75$). The poles of the closed-loop system, marked with crosses, are quite inside the unit circle, which is the condition for stability. However, simulations have shown some instability problems for low loads (below 25% of the nominal load) under sudden changes in the load value. These problems are due to two reasons. The first reason is the delay introduced by the control algorithm. When a positive step load takes place, the output voltage decreases and its maximum can appear at the beginning of the period, as shown in the right part of Fig. 6. Therefore, the parameter "m" changes temporally to 0, worsening stability. The second reason is that the control has been calculated around a working point equal to the nominal load, but the transfer function changes for different loads. The discrete root locus for 25% of the nominal load and $m = 0$ is shown in Fig. 8. The poles of the original closed-loop system, marked with crosses, are out of the unit circle, so the system is unstable.

An easy solution to avoid instability would be to decrease the constant used in the control—K in (14). However, this solution would decrease the dynamic response of the voltage-loop.

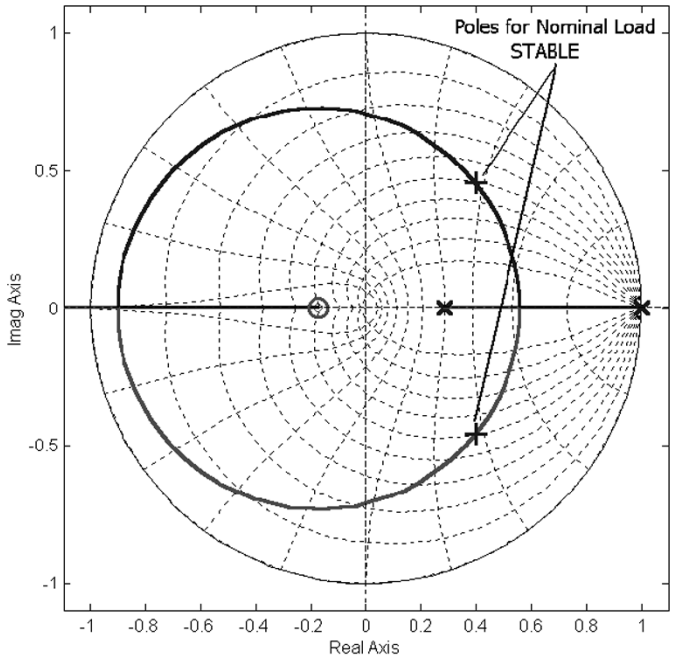

Fig. 7. Discrete root locus for nominal load and $m = 0.75$.

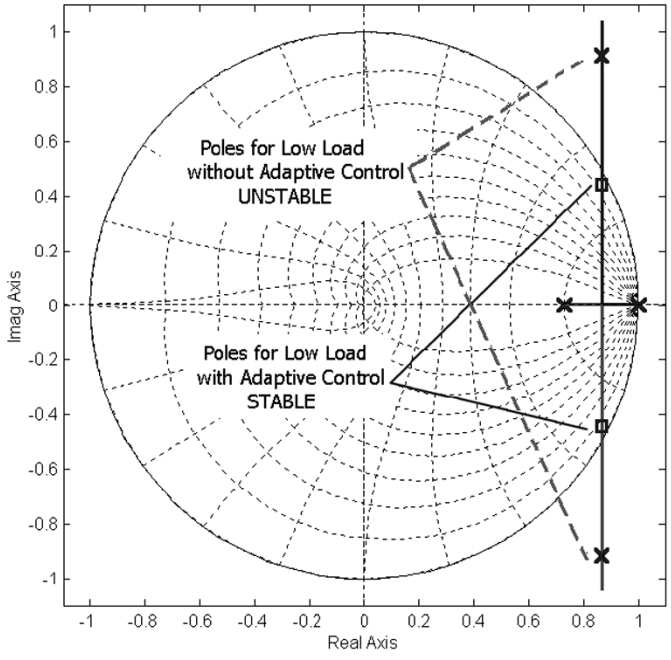

Fig. 8. Discrete root locus for 25% nominal load and $m = 0$.

The proposed and implemented solution is to use an adaptive algorithm for the voltage-loop. It consists in using two different constants in the voltage-loop algorithm, depending on the load value. The original constant is used for loads over 50% of the nominal load, and the second constant for lower loads. This second constant is 1/4 of the original one, and its corresponding poles are marked with squares in Fig. 8. A power of two is used because in this way the multiplier is substituted by a shifter whichever the constant being used. The new closed-loop poles are inside the unit circle even under the worst delay conditions $(m = 0)$, assuring therefore stability.

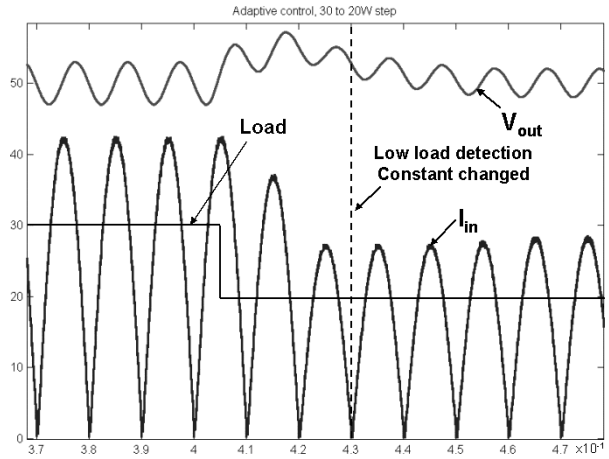

Fig. 9. Step from 60 to 40% nominal load (simulation).

The last problem is to measure the load value in order to change the constant used in the adaptive control. The goal is to avoid any further measurements, because that would mean using more ADCs. The proposed solution is to use the previous values calculated for G_{in}, which in fact represent the output load because the input and output powers are the same in steady state. The result is an adaptive control in which the control algorithm is changed by the controller when the load changes from above to under 50% of the nominal load, or vice versa. A simulation with a change from 60% to 40% of the nominal load is show in Fig. 9.

D. Possible Improvements

Some improvements in the voltage-loop considered for the future are as follows.

1) Maximum and minimum control. The V_{out} ripple effect is cancelled measuring the maximum V_{out} value each cycle. However, that means that the mean V_{out} value can change slightly depending on the load. This problem can be overcome measuring the minimum and the maximum values, and trying to control both simultaneously.
2) The G_{in} value is changed only once every rectified V_{in} cycle (10 ms for 50 Hz mains). This is enough for the voltage-loop because it is a slow one. For improving its dynamics, the G_{in} value should be refreshed with a higher frequency. However, this would worsen the Power Factor Correction. A possible solution would be to refresh G_{in} with a higher frequency only during transitions, when good dynamics are necessary.

V. Protection Issues

An important advantage of using an FPGA for the controller is that some protections can be added with no additional resources and almost no drawback in performance. The FPGA concurrency allows to execute the logic dedicated to the main control (the two loops explained before) and any additional logic devoted to protections simultaneously. In this way, there is almost no drawback in the controller performance because the control logic is executed as if there were no protections. Even more, the protections are executed continuously, instead of the

periodic execution in a DSP. DSP solutions must keep the protections resources at a minimum because the main control is stopped while a protection is verified.

Digital controllers can implement protections much easier than their analog counterparts. Analog ICs used for power converters control do not include protections. Therefore, adding a protection in an analog controller means adding hardware outside the IC controller. That is an advantage of digital controllers, in which no additional components are necessary for adding new algorithms. In fact, any algorithm can be added while there are available resources (memory program in DSPs or empty macrocells in FPGAs). Besides, the same measured signals (V_{in}, I_{in} and V_{out}) used for the main control are also used as protections input information.

Four protections have been implemented in the proposed controller.

1) Overvoltage: a limit to the output voltage. Whenever this limit is reached the switch is kept off in order to avoid possible damages.
2) Overcurrent: a limit to the input current. This is not a limit to the mean input current, but to the instant input current. Again, the switch is kept off when this limit is reached. It avoids the transformer saturation or any other physical damages.
3) Maximum duty cycle: an upper limit to the duty cycle. This is an usual protection that assures constant switching frequency and that avoids possible physical damages.
4) Maximum duty cycle change: the duty cycle can not change beyond a limit from switching cycle to switching cycle. This avoids sub-harmonic oscillations that can take place if the duty cycle is above 0.5. However, the control dynamics could be affected by this limit, so it has been calculated in order to let the dynamics unaffected.

VI. Modeling Method

The proposed control has been modeled in VHDL language and implemented in an FPGA [15]. This modeling method has two main advantages.

1) Rapid changes and error correction. If a control mistake is detected, the only handmade job is to change the VHDL code accordingly. Synthesis and reprogramming jobs are made automatically by tools, shortening considerably the time spent in the correction. The same process is used for adding any new functionality. No hardware change is necessary for changing the control algorithm. This feature has proven to be very useful, specially in prototypes like the one presented here.
2) Technology independence. The same VHDL code can be easily synthesized into any other FPGA or even an ASIC. This allows using different technologies for the prototype and the final implementation.

A problem of developing a digital control for a power converter using VHDL is simulating the whole system (digital control and power converter together). There are some simulators for mixed (analog–digital) signal models, like PowerSim, but they handle C code and not VHDL. Other option is using Matlab-Simulink, which is valid for the power converter simulation and can also generate VHDL code from Simulink models through specific plug-ins like Xilinx System Generator. However, there is not a direct way to generate generalized VHDL RTL code from Simulink and, anyway, that would change the control design flow adopted (starting from VHDL). A third option is using VHDL-AMS, a version of VHDL that includes analog functions [16]. However, a major goal is to obtain a model ready for fast simulation. This is due to the need of simulating both control loops, which requires simulating thousands of switching cycles (millions of system clock cycles). The solution adopted is to create VHDL functional models for the rest of the system, that is, a model for the power converter (flyback) and other one for the ADCs. This is reflected in Fig. 10. The flyback model is just a first order model representing ideal components. Its first order equations model the behavior of the inductor and capacitor under both conditions: switch on and off. No switching phenomena can be studied in this model but it works for controlling the signals evolution and, therefore, testing the control. The ADCs model includes their delays, which have to be taken into account for a detailed analysis of the controller. In this way, the control was exhaustively tested under all conditions and it did work at the first time in the actual system.

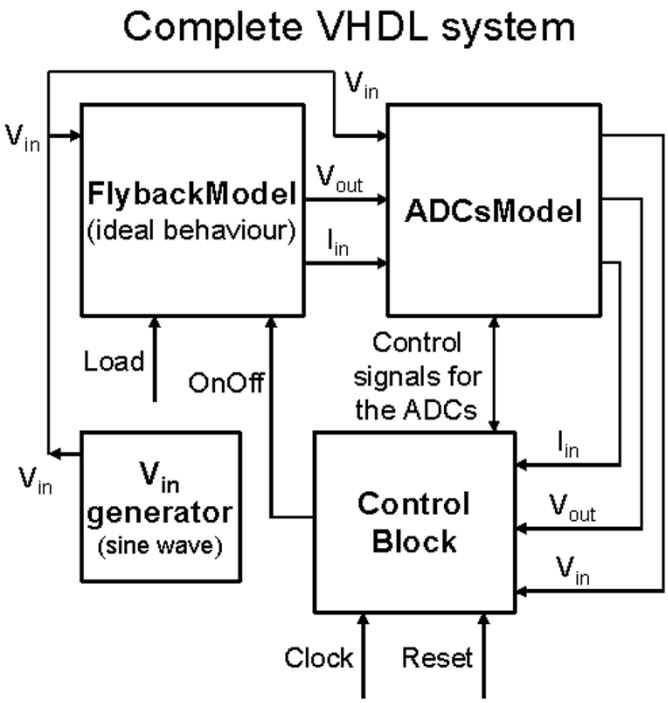

Fig. 10. Complete VHDL model for simulation.

VII. Experimental Results

An actual power converter prototype has been designed in order to test the proposed control strategy. A 50 W flyback converter operating in CCM, with 110 V in the ac input and 48 V in the dc output has been built. The chosen switching frequency is 50 kHz. Probably, DCM is preferred for such a low power. However, CCM is more interesting as a test bench because the duty cycle can not be kept constant for power factor correction.

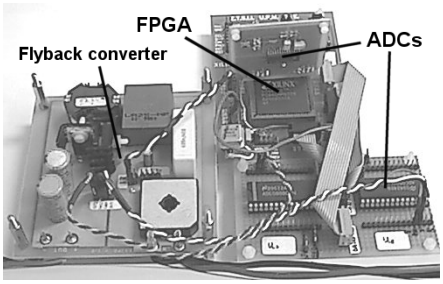

Fig. 11. Actual prototype.

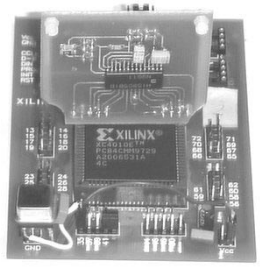

Fig. 12. Digital controller implemented in an FPGA.

Anyway, the designed control works in both CCM and DCM. This flyback converter is shown in Fig. 11 with the control circuitry. The prototype has only testing purposes. Size could be easily reduced with a redesign of the control layout.

The proposed controller has been implemented in a Xilinx FPGA, model XC4010. Due to the control simplicity, it has occupied only 5200 equivalent gates (even less than an 8-b microprocessor). The resulting digital design is fully synchronous, using a 20 MHz system clock. A picture of the chosen FPGA with the input current ADC is shown in Fig. 12. For prototyping purposes FPGAs are well indicated, but for a possible series production ASIC implementation could be very interesting from the cost point of view.

The A/D converters are a HI5805 (12-b, 5 MSPS) for the input current and two ADC0808 (8-b, 10 kSPS) for the output and input voltage. The ADC for the current loop is critical in terms of speed, while the ADCs involved in both voltages sampling can be low speed converters or even can be substituted by a multi-channel ADC. Although the input current ADC has a 12-b resolution, the experimental results have shown that 8 b would be enough for any of the ADCs involved in this application.

Static and dynamic tests have been performed with the prototype. Both tests are reflected in Figs. 13 and 14 respectively. *Steady state operation* is quite satisfactory. Input current is almost sinusoidal, obtaining a PF higher than 0.99 even in transient operation (input current waveform keeps a good shape in load transitions). This is due to the high accuracy of the proposed current-loop algorithm, a digital implementation of a charge control. The output voltage ripple is inherent to the PFC operation of the converter, since it works as a resistor emulator and the output bulk capacitor must absorb the oscillations on the input power.

Dynamic response, shown in Fig. 14, shows a sudden step from 40% to 60% of the nominal load (resistive load is used). Input current reaches the new amplitude in only one line cycle.

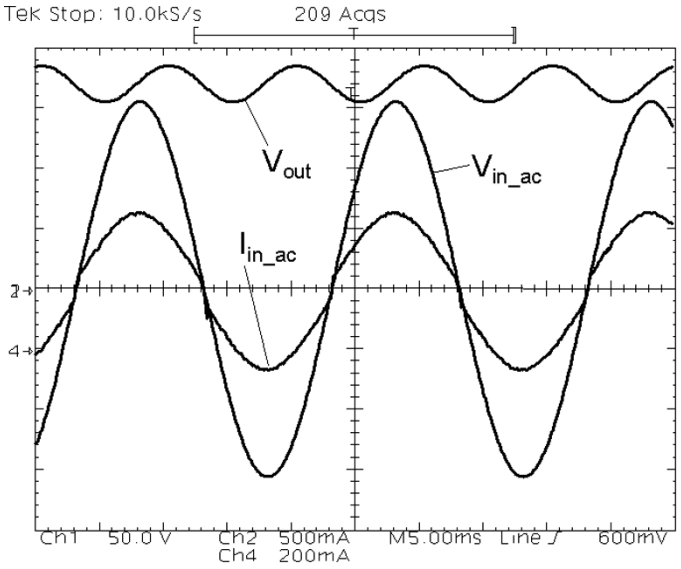

Fig. 13. Steady-state response (experimental).

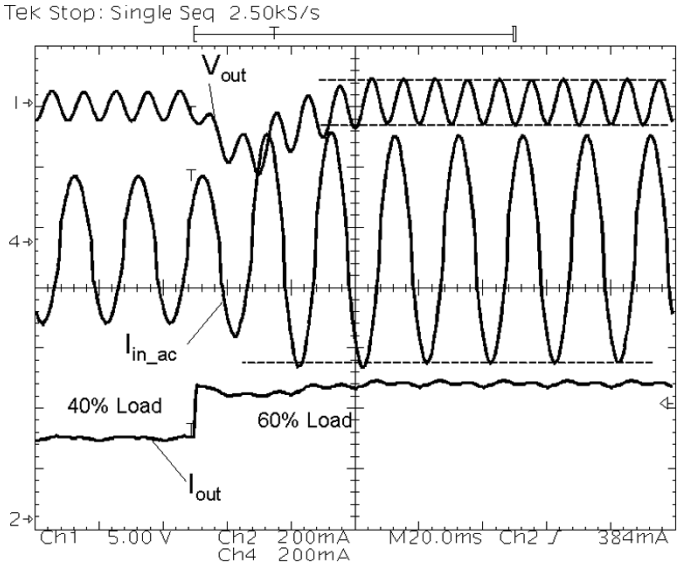

Fig. 14. Dynamic response (experimental).

It shows that the voltage loop is fast enough to change the input conductance in very few cycles.

Output voltage transition is slower due to the delay of the voltage loop and to the dynamic of the capacitor (in the actual prototype the output bulk capacitor is 440 μF). However, the output voltage does not fall dramatically in the transitions. The output bulk capacitor stores a lot of energy and it is able to keep the voltage within a reasonable interval when a variation of the load occurs, in spite of the inherent output voltage ripple in the PFC circuits.

An analog control based on a UC3854 circuit has been designed and simulated, in order to compare both solutions (digital and analog). The analog control includes a *feed-forward* loop, a current loop (averaged current mode control) and a PI voltage loop. In order to compare it with the presented digital control, the designed analog one has a similar oscillation in the output voltage.

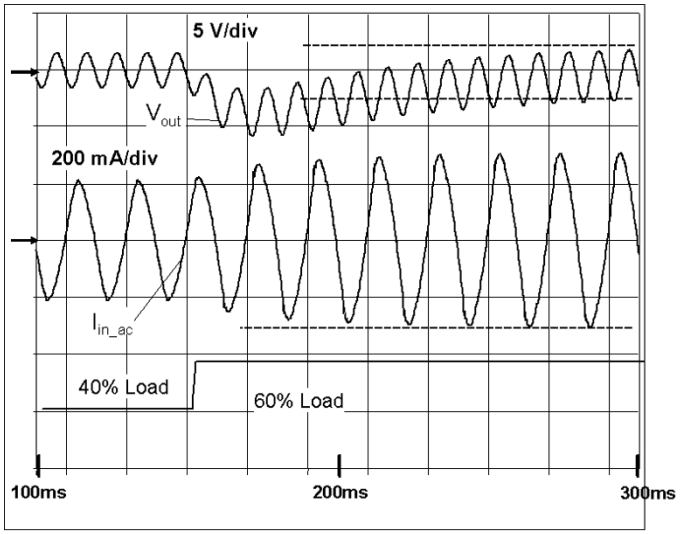

Fig. 15. Analog control dynamic response (simulation).

The analog control has been simulated under the same conditions used for the digital control (load step from 40% to 60%) in order to make a proper comparison. Reference lines have been added in Figs. 14 and 15 to compare the dynamic response. In the case of the proposed digital control, the input current (I_{in_ac}) reaches its new value in only one line cycle, while the analog control takes three line cycles. The output voltage (V_{out}) takes less time in the digital solution (2.5 line cycles) that in the analog control (five line cycles).

The simulated analog control has a worse dynamic response for both the input current and the output voltage. This is due to the filter used for avoiding the output voltage ripple, which is necessary for avoiding perturbations on the current loop. This filter turns the voltage loop slower, and then the input current establishment time is higher. In conventional analog controls a trade-off must be reached between dynamic response and input current distortion. Relaxing the output voltage filter improves the dynamic response, but worsens the input current distortion (PF). The proposed solution avoids the output voltage filter, so the dynamic response can be improved without affecting the input current distortion.

VIII. Conclusion

A full digital controller for a power converter has been proposed. The most important difference from previously proposed digital controllers is that it is based on specific hardware (an FPGA in this case) instead of the common DSP solutions. The main advantage of this method is that all the logic is executed continuously and simultaneously (*concurrent* operation). New high-speed algorithms can be used in this way. However, complex arithmetic operations should be avoided whenever possible, because they need much resources (silicon area). This is the other main characteristic of the proposed controller: *simplicity*.

The target system of this work is a flyback converter working as a Power Factor Corrector, although it can be easily adapted to any other topology or application. This task (PFC) makes the control a bit more complicated, involving two control loops. One of them, the current loop, has been designed as a digital version of a charge control, which is independent of the conduction mode. This has been possible thanks to the FPGA concurrency, showing a high accuracy in spite of its simplicity. A good dynamic response has also been achieved, and protections have been added without any external resources.

Static and dynamic responses have been evaluated experimentally. The proposed digital control has been compared with a conventional analog solution. Dynamic response improves with the proposed control (one line cycle of establishment time for the input current), in spite of its simplicity. Static performance is very satisfactory (PF $>$ 0.99).

Implementing the control algorithms in a hardware description language (as VHDL) allows high flexibility and technology independence. The same controller can be directly synthesized into any other FPGA or even in an ASIC, and it can also be added to other logic blocks forming a more complex multi-task system in a single chip.

Solutions based on specific hardware, that allows high concurrency, are suitable to be used in other power electronics applications, like interleaved converters, power integrated circuits, dynamic voltage scaling, etc.

Acknowledgment

The authors thank A. Soto and J. A. Oliver for their support in control analysis and comparison with traditional control algorithms.

References

[1] Y. Duan and H. Jin, "Digital controller design for switchmode power converters," in *Proc. IEEE Appl. Power Electron. Conf. (APEC)*, vol. 2, 1999, pp. 967–973.

[2] J. Xiao, A. V. Peterchev, and S. R. Sanders, "Architecture and IC implementation of a digital VRM controller," in *Proc. IEEE Power Electron. Spec. Conf. (PESC)*, vol. 1, 2001, pp. 38–47.

[3] A. Prodic, D. Maksimovic, and R. W. Erickson, "Design and implementation of a digital PWM controller for a high-frequency switching dc–dc power converter," in *Proc. IEEE Ind. Electron. Soc. Conf. (IECON)*, vol. 2, 2001, pp. 893–898.

[4] B. J. Patella, A. Prodic, A. Zirger, and D. Maksimovic, "High-frequency digital controller IC for DC/DC converters," in *Proc. IEEE Appl. Power Electron. Conf. (APEC)*, vol. 1, 2002, pp. 374–380.

[5] A. M. Wu, J. Xiao, D. Markovic, and S. R. Sanders, "Digital PWM control: Application in voltage regulation modules," in *Proc. IEEE Power Electron. Spec. Conf. (PESC)*, vol. 1, 1999, pp. 77–83.

[6] M. Fu and Q. Chen, "A DSP based controller for Power Factor Correction (PFC) in a rectifier circuit," in *Proc. IEEE Appl. Power Electron. Conf. (APEC)*, vol. 1, 2001, pp. 144–149.

[7] Y. T. Feng, G. L. Tsai, and Y. Y. Tzou, "Digital control of a single-stage single-switch flyback PFC AC/DC converter with fast dynamic response," in *Proc. IEEE Power Electron. Spec. Conf. (PESC)*, vol. 2, 2001, pp. 1251–1256.

[8] A. Prodic, J. Chen, R. W. Erickson, and D. Maksimovic, "Digitally controlled low-harmonic rectifier having fast dynamic response," in *Proc. IEEE Appl. Power Electron. Conf. (APEC)*, vol. 1, 2002, pp. 476–782.

[9] S. Hauk, "The roles of FPGAs in reprogrammable systems," *Proc. IEEE*, vol. 86, pp. 615–638, Apr. 1998.

[10] *IEEE Standard VHDL Language Reference Manual*, IEEE Std 1076-1993.

[11] A. Rushton, Ed., *VHDL for Logic Synthesis*. New York: Wiley, 1998.

[12] W. Tang, Y. Jiang, G. C. Hua, F. C. Lee, and I. Conen, "Power factor correction with flyback converter employing charge control," in *Proc. IEEE Appl. Power Electron. Conf. (APEC)*, 1993, pp. 293–298.

[13] A. V. Peterchev and S. R. Sanders, "Quantization resolution and limit cycling in digitally controlled PWM converters," in *Proc. IEEE Power Electron. Spec. Conf. (PESC)*, vol. 2, 2001, pp. 465–471.

[14] G. Spiazzi, P. Mattavelli, and L. Rossetto, "Power factor preregulators with improved dynamic response," in *Proc. IEEE Power Electron. Spec. Conf. (PESC)*, vol. 1, 1995, pp. 150–156.
[15] T. Riesgo, Y. Torroja, and E. de la Torre, "Design methodologies based on hardware description languages," *IEEE Trans. Ind. Electron.*, vol. 46, pp. 3–12, Feb. 1999.
[16] *IEEE Standard VHDL Analog and Mixed-Signal Extensions*, IEEE Std 1076.1-1999.

Angel de Castro was born in Madrid, Spain, in 1975. He received the M.Sc. degree in electrical engineering from the Universidad Politécnica de Madrid, Spain, in 1999, where he is currently pursuing the Ph.D. degree.

His research interests include digital control of switching mode power supplies, digital circuits design, and smart transducers.

Pablo Zumel was born in Burgos, Spain, in 1973. He received the B.Sc. degree in electrical engineering from the Universidad de Burgos, Spain, in 1995, the M.Sc. degree in electrical engineering from both the Universidad Politécnica de Madrid and the Ecole Centrale Paris, in 1999, and is currently pursuing the Ph.D. degree at the Universidad Politécnica de Madrid.

His research interests include interleaved converters, digital control for power converters, and integrated magnetics.

Oscar García (M'99) was born in Madrid, Spain, in 1968. He received the M.Sc. and Ph.D. degrees in electrical engineering from the Universidad Politécnica de Madrid, Spain, in 1992 and 1999, respectively.

He is an Associate Professor at the Universidad Politécnica de Madrid. He has been involved in more than 15 research projects and he has published nearly 60 papers in IEEE conferences. His research interests are switching mode power supplies, power factor correction, power architectures, and digital control applied to power electronics.

Teresa Riesgo (M'96) was born in Madrid, Spain, in 1965. She received the M.Sc. and Ph.D. degrees in electrical engineering from the Universidad Politécnica de Madrid, Spain, in 1989 and 1996, respectively.

Since 1990, she has been working at the Electronic Engineering Division, Universidad Politécnica de Madrid, as Assistant Professor, and since 1998 as Associate Professor. Her main research interests are ASIC design, VLSI test techniques, and embedded systems.

Javier Uceda (M'83–SM'91) was born in Madrid, Spain, in 1954. He received the M.Sc. and Ph.D. degrees in electrical engineering from the Universidad Politécnica de Madrid, Spain, in 1976 and 1979, respectively.

Since 1986, he has been a Professor at the Universidad Politécnica de Madrid. His research interests include high-frequency high-density power converters, high-power-factor rectifiers, and modeling of magnetic components.

Dr. Uceda is a member of the Editorial Board of the *European Power Electronics and Drives Journal* and of the Steering Committee of the European Power Electronics and Drives Association. He has been Associate Editor of the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS. He was also the Vice President of the Technical Activities of the IEEE Industrial Electronics Society for 1991. He was Technical Program Committee Chairman of the IEEE Power Electronics Specialists Conference in 1992 and General Chairman of the European Conference on Power Electronics and Applications in 1995. He is senior AdCom member of the Industrial Electronics Society.