Verilog-A behavioral modeling of power converters

Giorgio Spiazzi, Simone Buso

Department of Electronics and Informatics

University of Padova
Via Gradenigo 6/a   35131 Padova – Italy
e-mail: *giorgio.spiazzi@dei.unipd.it*
simone.buso@dei.unipd.it

Donato Tagliavia

**ST Microelectronics

Via Stradale Primosole 50, 95121 Catania - ITALY
Phone: +39-095-7401111 Fax: +39-095-7406006
E-Mail: donato.tagliavia@st.com

Abstract - **The paper presents the implementation of a Verilog-A based, behavioral macro model for a flyback power converter to be used in a Spice based simulator (SpectreS by Cadence©). The paper discusses the effectiveness of this simulation strategy in reducing the complexity of a simulation problem, namely the analysis of a new smart-power integrated circuit (IC). The IC consists in a flyback PFC controller with integrated power switch. The problem of simulating the quite complex control circuitry at the transistor level, in the presence of a power switching converter, for a sufficiently long time to appreciate the line side behavior of the PFC, is efficiently solved by substituting the power converter with an input-output behavioral macro model developed in the Verilog-A language. The paper describes the effectiveness of the Verilog-A language in creating macro-models of power converters generalizing the results obtained for the specific flyback case. The description of the converter's operation, also considering the discontinuous conduction mode (DCM) of operation is shown to be particularly straightforward. The paper also describes the basic controller structure and presents some results of the extensive simulations performed to check the functionality of its different component units with the power converter's behavioral model.**

I. INTRODUCTION

Spice based simulations of electronic circuits with very different time constants always exhibits severe convergence problems which, forcing to reduce the integration step size, compel to very long simulation times. The problem is particularly difficult to solve when the system to be simulated includes a power converter (slow time constants) and a control circuit which is described at the transistor level (very fast time constants). The behavioral macro modeling is a simulation strategy that appears to be particularly effective in reducing the complexity of a simulation schematic substituting parts of the electrical circuit with a input-output equivalent model. This model can be generated with different tools, all being essentially high-level programming languages with specific instruction libraries, allowing the programmer to manipulate run-time simulation variables like electrical signals or the *time* variable. The Spice simulator then treats these user developed models as conventional sub-circuits. Being the whole set of equations describing the part of the circuit substituted by the macro-model reduced to a very small sub-set, the complexity of the simulation is significantly reduced. The effectiveness of the approach is more evident when the macro-model takes the place of a

power device, whose switching process and related phenomena (recovery of diodes, turn-on transients etc.) are this way drastically simplified. Consequently, the simulation can be focused on the remaining parts of the circuit, which are still electrically described at the transistor level and so with maximum detail, without being slowed or even blocked by the power converter's dynamic behavior.

The paper presents the implementation of a Verilog-A [1] based, behavioral macro-model for a flyback power converter to be used in a Spice based simulator (SpectreS by Cadence©). The paper discusses the effectiveness of this simulation strategy in reducing the complexity of a simulation problem, namely the analysis of a new smart-power integrated circuit (IC). The IC is designed to provide, in a single chip, the power switch (of the "emitter switching" type [2]) and the control circuitry to implement a flyback power factor corrector (PFC) operating in continuous conduction mode (CCM) [3]. The control strategy is specifically developed to minimize the IC complexity, its pin count and cost, while maintaining a sufficiently high performance level to guarantee compliance with EMC low frequency standards, namely EN 61000-3-2.

The problem of simulating the control circuitry at the transistor level, in the presence of a power switching converter, for a sufficiently long time to appreciate the line side behavior of the PFC, is efficiently solved by substituting the power converter with an input-output behavioral macro model developed in the Verilog-A language. The paper describes the effectiveness of the Verilog-A language in creating accurate macro-models of power converters, thus generalizing the results obtained for the specific flyback case. The paper also describes the internal structure of the control circuit and its integrated circuit implementation with a ST Microelectronics smart power proprietary technology, known as VIPower M3© [4]. Some of the results of the extensive simulations of the IC main control functions with the power converter macro-model are presented.

II. SMART POWER INTEGRATED PFC CONTROLLER

The basic topology of the considered power converter is shown in Fig. 1. The dotted box includes all the parts of the circuit that have been substituted by the Verilog-A
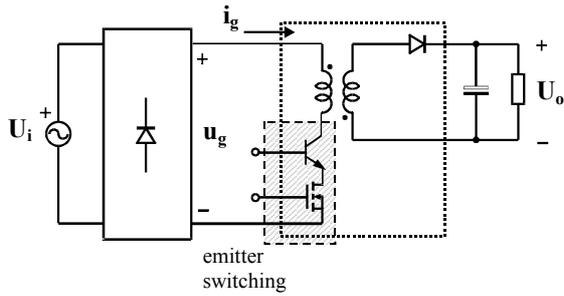
Fig. 1 – Converter's basic topology (the shaded area includes the equivalent circuit of the emitter switching device)
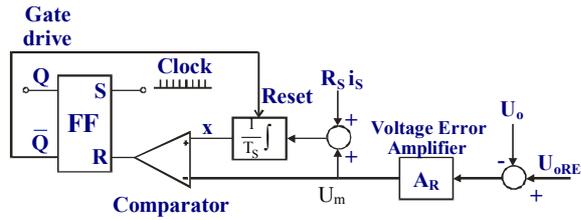
Fig. 2 – Controller's simplified schematic

based, behavioral macro-model, described in the following section.

Fig. 2, instead, shows a simplified schematic of the PFC integrated controller. As documented in [3], the considered technique is a modified non-linear carrier control, which allows to achieve an almost sinusoidal line current with low complexity. It basically requires the integration of the switch current and comparison with a suitable, internally generated, carrier signal. A more detailed block diagram of the IC under development is shown in Fig. 3, which includes the main required control and supply functions and the power switch. Note that, for the sake of simplicity,

only the control related connections are shown. As already stated, the main goal of the controller design is to minimize the pin count and so the IC cost, being the largest part of it determined by the IC package. This is the reason why the control technique mentioned above was selected. As can be seen, it only requires 8 pins for the IC: in addition to the supply (V_{DD}), ground (Gnd) and high voltage collector (HV_{CC}) pins, the IC presents feedback (FB) and error amplifier output (O_{ERR}) pins for compensation purposes, oscillator pin (Osc) to set the clock frequency (i.e. the switching frequency), and an additional pin (Cap) used to connect an external capacitor that, together with the voltage to current converters, implements the integrator function of Fig. 2. Reset of the integrator is provided by the integrated transistor shown in Fig. 3, which is controlled by the negative output of the flip-flop. Despite its reduced pin count, the IC includes all the typical supervising functions, namely a start-up circuit, an under-voltage protection and a thermal and over-current protection circuit. The development of this integrated control circuit has already undergone the first steps, that are the design of the sub-circuits implementing each of the basic functions shown in Fig. 3, their layout definition and first prototyping as separate component blocks or cells. The experimental test of the component cells, in stand-alone configuration, is currently in progress. Also the emitter switching device, which will be used as the integrated power switch, is already available as a discrete component and has already been characterized and tested in a converter prototype [5]. It is worth adding here that the employed technology (VIPower M3) basically consists of a modified 2 μm CMOS process, allowing the generation of CMOS, Bipolar and vertical high voltage (up to 25 V) MOS devices, together with a particular power switch structure, named emitter switching [2], rated for a 6 A maximum current and
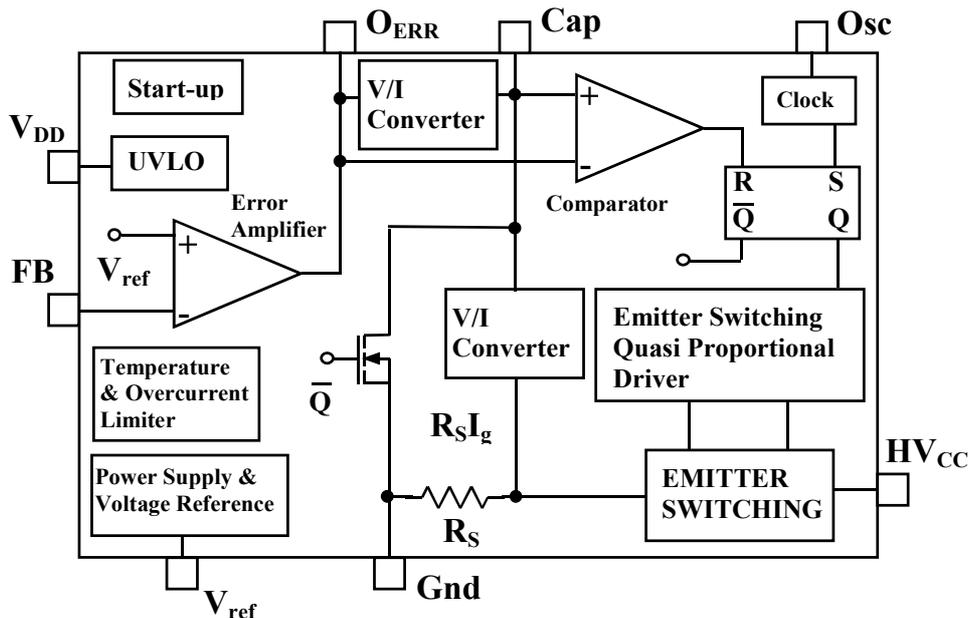
Fig. 3 – Block diagram of the PFC integrated controller

up to 1500 V maximum voltage.

Before the final IC can be manufactured, the functionality of the complete control circuit has to be checked. In order to do that, simulation of the basic controller cells together with the power converter has to be performed. The aim of this activity is to verify the behavior of the component cells for all the typical converter's operating conditions. In particular, the behavior of the driver circuit for the emitter switching, of the flip-flop and comparator block, and of the two voltage to current converters required by the controller implementation, is critical. This has to be within the specifications for all the operating conditions.

The problem of simulating at the transistor level all the controller's fundamental cells together with a power converter, whose operation is periodical at twice the line frequency, is very serious. In practice, detailed simulation of both the power converter and the controller generates severe convergence problems and/or calls for unacceptably long simulation times, even on a quite powerful workstation. Since the focus of the analysis is on the control circuit operation, there is no loss of significant information if the power converter is substituted by an input-output equivalent, or behavioral, model, also called macro-model. Being the complexity of the behavioral macro-model very low with respect to the electrical power converter model, the complexity of the simulation is significantly reduced. Consequently, convergence problems are minimized and simulation times become acceptable.

III. POWER CONVERTERS MODELING WITH VERILOG-A

The behavioral model of any power converter topology can be easily developed with the Verilog-A language. The philosophy underlying this language is to provide the user with functions allowing to manipulate run-time simulation quantities, both global, like time and external electrical signals, and locally defined inside the Verilog-A module itself, as program variables. The manipulation may include logical and conditional operations, simple arithmetic, but also integral and differential operations. The user's program is then compiled and transformed into a simulation block of the sub-circuit type, with input and output pins, like the one shown in Fig. 4, which includes a user configurable parameter list. The particular graphical aspect of the model clearly depends on the underlying Spice simulator. In our case, the SpectreS simulator, included in the Cadence Analog Artist© integrated circuit design tool was used. The key of the modeling program is the definition of the integral equations referring to the converter's state variables. As in any numerical integration program, also for the Spice based simulator state variable equations are better written in the integral form, which gives higher numerical stability. In the case of the converter model shown in Fig. 4, the only state variable taken into account is the transformer magnetizing current, since the output capacitor is not included in the converter macro-model. Based on the switch and diode states and on the input voltage, the program implementing the
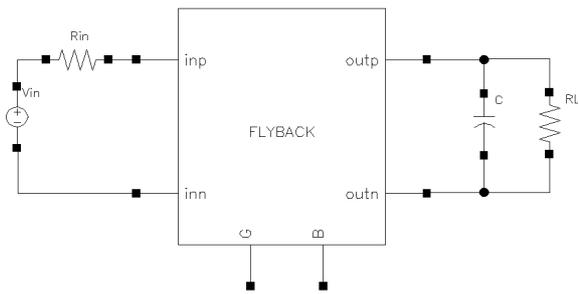
Fig. 4 – Flyback converter macro-model.

macro-model basically computes the voltage across the magnetizing inductance and integrates it to get the magnetizing current. As can be seen in Fig. 4, the output of the macro-model is the diode current, which is calculated from the magnetizing current, taking into account the transformer turns ratio. The switch state depends on the control signals G and B. These represent the gate voltage and base current of the emitter switching power device, which are generated by the driver sub-circuit. A positive voltage on pin G and a positive inbound current in the B terminal, both higher than suitable threshold values, correspond to the switch on-state. The diode state is derived from the switch state and continuous conduction mode (CCM) is continuously (i.e. at any integration step) checked. If the converter enters the DCM, the diode current is set to zero, and so is the voltage across the magnetizing inductance, until the next switch turn-on. At the same time, the value of the integral of the magnetizing inductance voltage is set to zero. This allows the DCM to be dealt with accurately. It is worth noting that the approach considered here directly applies to any switching power converter topology, the only difference being the set of equations describing the state variable evolution. The Verilog-A program we developed is given in Appendix A. As can be seen, its structure is extremely simple, consisting of basically two parts. The initial part declares the module, its variables and parameters. A module is declared in a way very similar to a function declaration in high level programming language (e.g. C language). The module parameters are all the signals the module processes as inputs and/or outputs. Each of these corresponds to a pin of the Spice sub-circuit, as shown in Fig. 4. Input and output pins are also given a type or *discipline* (in this case *electrical*) to allow the user to simulate complex systems where different types of physical quantities (e.g. mechanical or thermal quantities) are considered and need to be kept logically separate. The pin function (input, output or, as in this case, input and output or *inout*) must be declared by the user at this point in the program. The electrical state (in terms of voltage or current) of each pin must be defined within the program, so as to be found by the simulator at any simulation step, otherwise the simulation fails to converge. In our case, as it was previously explained, this requires the calculation of input and output currents. As can be seen, the initial part of the
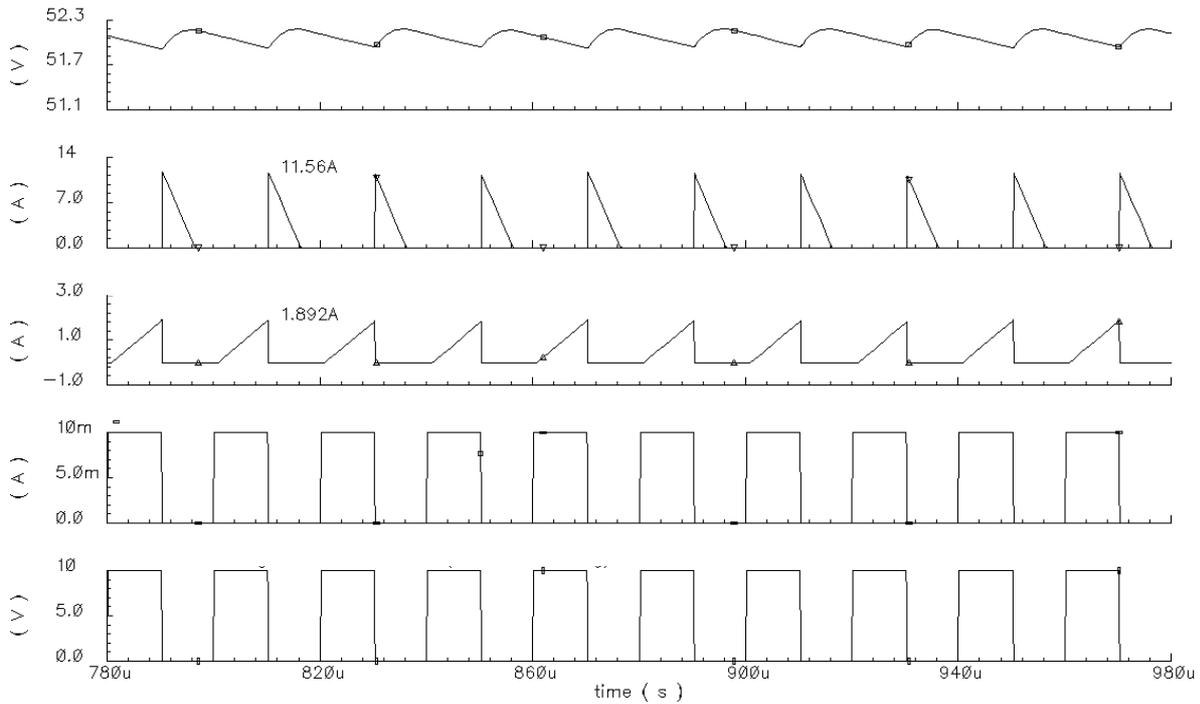
Fig. 5 – Converter's signals in DCM. From top to bottom: load voltage, diode current, switch current, switch base current and gate voltage.

program also includes the declaration of constant parameters and internal variables, which can be real or integer quantities.

The second part of the program implements the actual converter model. As can be seen, Verilog-A syntax is close to that of any high level programming language. As a consequence, the readability of the code is very good. It is worth noting the presence of keywords, like *initial_step*, allowing the user to perform initialization operations only at the beginning of the simulation. The rest of the program is instead executed at any integration step. The availability of the function *idt*, in general, allows the user to define integral differential equations describing the dynamical system under simulation. It is important to note that reset of the integral function is also allowed. In our case, the *idt* function is used to compute the transformer magnetizing current, integrating the magnetizing inductance voltage, as it was previously explained. Simple *if* statements allow to determine the state of the switch and so to compute the voltage across the inductance and, at the same time, to assign input and output current values. As can be seen, the inductance voltage is defined to be equal to the input voltage when the switch is on, and to the output voltage reflected to the primary side, when the switch is off and the inductance current is positive. If this gets to zero, i.e. when the converter enters the DCM, the voltage is set to zero and so is the integral function value.

As previously said, when writing a Verilog-A program, it is important to verify that the electrical state of the input and output pins is defined in any condition. In case the module is not able to provide the electrical state in a particular integration step, for example because of a logic error in the program, a run-time error is generated during the simulation, which fails to converge.

In our case, being the model quite simple, only a short debugging time was required to obtain a stable simulator operation. In case of a more complicated problem, the
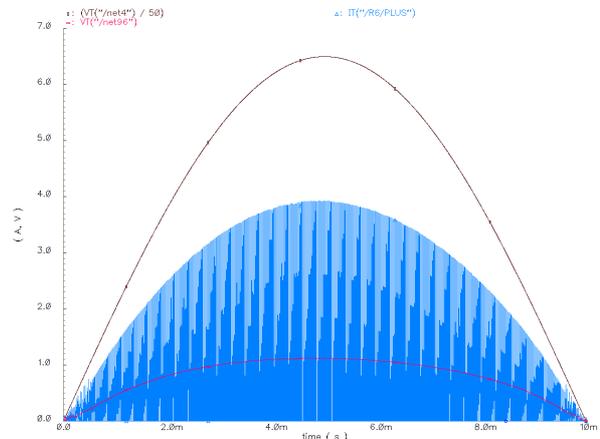
Fig. 6 – Converter's operation in a line half-period. From top to bottom: line voltage (50 V pu), input current and average input current (1 A pu).
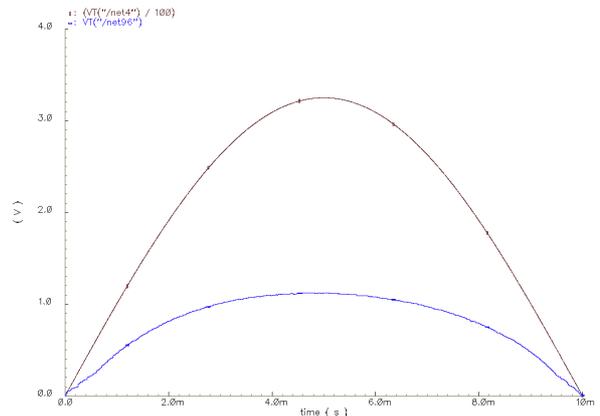
Fig. 7 – Converter's operation in a line half-period. From top to bottom: line voltage (100 V pu), average input current (1 A pu).

Verilog-A compiler included in our CAD software also provides a debugger unit, with step by step execution capability, which simplifies the elimination of logic errors.

IV. CONTROLLER SIMULATION

Firstly, the power converter macro-model has been tested to verify its correctness and eliminate possible bugs. Considering a constant duty-cycle (equal to 0.5) and input voltage (equal to 100 V), the critical signals have been checked, both in CCM and DCM. A typical result, for the DCM case, is given in Fig. 5. As can be seen, the expected waveforms are obtained with sufficient accuracy.

Since the fundamental controller cells had already been tested and validated one by one, the converter's behaviour as a PFC could be checked, at first in the nominal conditions. We therefore connected the main controller blocks, excluding the various protection and supervision blocks indicated in Fig. 3, and the converter's macro-model to get a quite accurate system model. We then simulated the PFC operation. The basic results are given in Fig. 6 and Fig. 7. As can be seen, the average input current waveform is close to sinusoidal, as expected. In addition, Fig. 8 shows a detail of the main control circuit waveforms. It is possible to conclude that the basic controller cells considered in Fig 3, and included in the system model, are operating correctly. To complete the validation of the controller cells' design, extensive simulations referring to other possible operating conditions, (e.g. lower output power, different input voltage) have been performed, also revealing a substantially correct operation of the control circuitry.

It is worth underlining that this work allowed us to verify in different conditions the internal operation, at the transistor level, of the basic analog integrated circuits making up the controller. This means that we have been able to monitor the effect of second order phenomena like propagation delays, operational amplifier gain and bandwidth limitations, V/I converter non-linearity and so on. We therefore made sure that in the typical, variable operating conditions, these undesirable non-ideal characteristics of the analog circuitry could not cause malfunctioning of the control system. We also made sure that the design specifications for the single cells, which were given with suitable tolerances, did not combine in such a way to determine an unacceptable behaviour at the system level.

However, it is necessary to underline that other aspects of the operation, for instance those related to the auto-compatibility of the IC (i.e. robustness of the control circuit to disturbances induced by the switch commutations), could not be taken into account, mainly because of the lack of complete technological models for the power switch and the control circuitry to account for substrate effects. This makes our analysis somewhat partial and requires experimental verification of the controller operation with a set of prototypal integrated controllers. Nevertheless, the confidence on the quality of the design has been improved.

V. CONCLUSIONS

The paper presents the implementation of a Verilog-A behavioral macro-model for a flyback power converter to
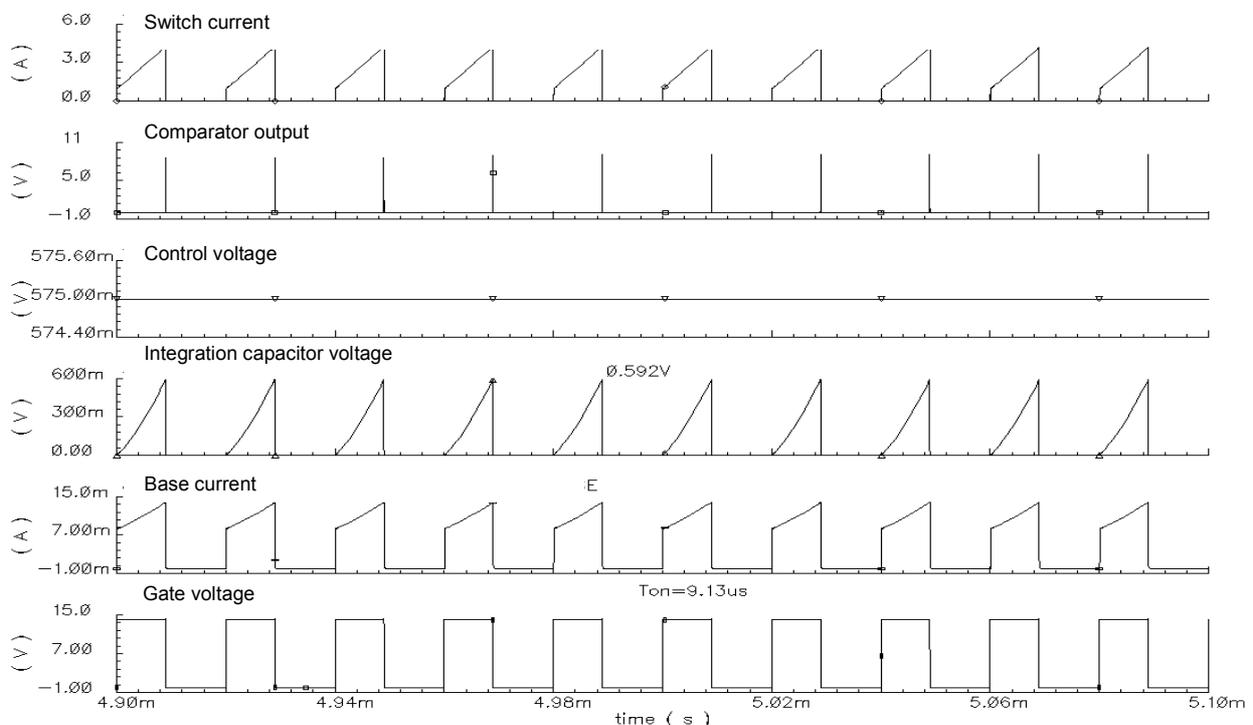
Fig. 8 – Control's signals. From top to bottom: sensed switch current, comparator output, voltage control reference, integration capacitor voltage, switch base current and gate voltage.

be used in a Spice based simulator (SpectreS by Cadence©). The paper discusses the effectiveness of this simulation strategy in reducing the complexity and time expense of the verification of a new smart-power integrated circuit's operation. This is designed to provide, in a single chip, the power switch and the control circuitry implementing a flyback power factor corrector operating in continuous conduction mode and, even if simple, is made up of several analog circuits (e.g. comparators, operational amplifiers). Simulating this control circuitry at the transistor level, in the presence of a power switching converter, for a sufficiently long time to appreciate the line side behavior of the PFC represents a serious problem, because numerical instabilities and unacceptably long simulation times, even in the case an ideal switch model is used, are encountered. This problem is efficiently solved by substituting the power converter with an input-output behavioral macro model developed in the Verilog-A language. The paper describes the effectiveness of the Verilog-A language in creating accurate macro-models of power converters, thus generalizing the results obtained for the specific flyback case. The paper also describes the internal structure of the control circuit and its integrated circuit implementation with a ST Microelectronics smart power proprietary technology, known as VIPower M3©. Some of the results of the extensive simulations of the IC main control functions with the power converter macro-model are presented.

VI. ACKNOWLEDGEMENT

The authors would like to thank Genesia Bertelle for taking care of the simulation activity and data collection.

REFERENCES

[1] D. Fitzpatrick, I. Miller, Analog Behavioral Modeling with the Verilog-A Language, Boston, USA, Kluwer Academic Publishers, 1998.
[2] M. Melito G. Belverde, A. Galluzzo, S. Palara, "Bipolar-MOS Monolithic Cascode Switch in VIPower Technology," IEEE IAS Annual Conference Proc., Denver, 1994, pp.1322-1325.
[3] S. Buso, G. Spiazzi, D. Tagliavia, "Simplified Control Technique for High-Power-Factor Flyback Cuk and Sepic Rectifiers Operating in CCM," IEEE Transactions on Industry Applications, Vol.36, No. 5, September/October 2000, pp. 1413-1418.
[4] S. Sueri, "VIPower M3: A New Smart Technology for High Power, High Speed Applications," Proc of PCIM, 1997, pp.19-27.
[5] G. Spiazzi, S. Buso, D. Tagliavia: "A Low-Loss High-Power-Factor Flyback Rectifier Suitable for Smart Power Integration", Power Electronics Specialist Conference (PESC) Proc., Galway, Ireland, June 18-23, 2000, pp. 805-810.

APPENDIX A

VERILOG-A PROGRAM IMPLEMENTING THE FLYBACK MACROMODEL

```
`include "constants.h"
`include "discipline.h"
module FLYBACK(inp, inn, outp, outn, B, G);
inout inp, inn, outp, outn, B, G;
electrical inp, inn, outp, outn, B, G;
parameter real ind_mag=0.001;
parameter real n=-6.06;
parameter real vgate_thres=0.2;
parameter real ibase_thres=0.001;
parameter real state_ing=0;
real il0, Vl, il;
integer reset;

analog begin
@(initial_step)
begin
   il0=state_ing*ind_mag;
   Vl=n*V(outp, outn);
   reset=0;
end
il=idt(Vl,il0,reset)/ind_mag;
if (il<=0)
 begin
   il=0;
   reset=1;
   il0=0;
 end
if ((V(G)>vgate_thres)&&(I(B)>ibase_thres))
begin
   Vl=V(inp, inn);
   I(outp, outn) <+ 0;
   I(inp,inn) <+ il;
   reset=0;
end else
begin
   Vl=n*V(outp, outn);
   I(inp, inn) <+ 0;
   I(outp, outn) <+ il*n;
   reset=0;
   if (il==0)
    begin
     Vl=0;
     reset=1;
     il0=0;
    end
end
end
endmodule
```