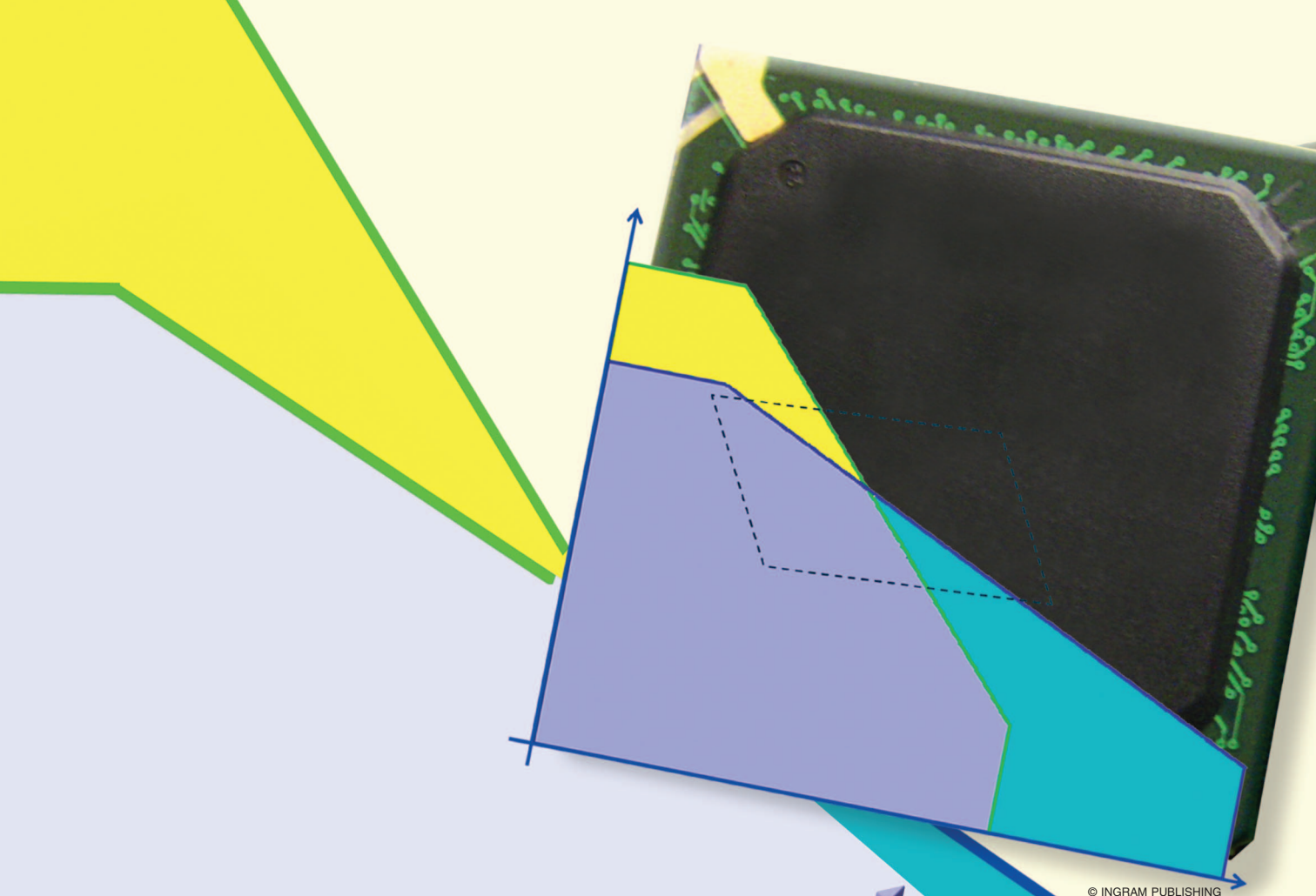


© INGRAM PUBLISHING

FPGA-Based Controllers

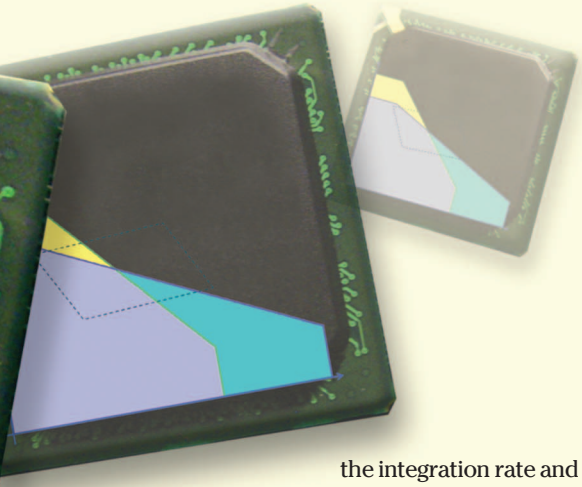
Different Perspectives of Power Electronics and Drive Applications

This article presents the benefits of using field-programmable gate array (FPGA)-based controllers for power electronics and drive applications. For this purpose, an algorithm perspective is first proposed, where it is stated that, depending on the intrinsic parallelism properties as well as level of complexity, it makes sense to implement each control algorithm on a specific hardware and/or software architecture to get the best performances in terms of execution time or the best ratio performance versus cost. Then, an application perspective is proposed where the constraints specifically linked to the control of power converters are discussed.

The ceaseless interest during the last 30 years for the digital control of power electronics and drive applications is mainly due to the great advantages that this technology offers compared to the analogous one. The flexibility, reduced design time, and possibility to implement very complex treatment in real time are among them. However, even if

Digital Object Identifier 10.1109/MIE.2011.940250
Date of publication: 25 March 2011

1932-4529/11/$26.00©2011IEEE



ERIC MONMASSON,
LAHOUCINE IDKHAJINE, and
MOHAMED WISSEM NAOUAR

the integration rate and timing performances of the current digital signal processors (DSPs) are very high [1], one cannot achieve the bandwidth of a simple analog controller. The main reasons are the necessity to add additional components like analog-to-digital (A/D) converters (ADCs) and zero holder to correctly interface the processor with its analog environment. Besides, computing time also needs to be taken into account, since it introduces a delay that can significantly reduce the bandwidth of the closed-loop system. Finally, the quantization that is inherent to digital implementation has to be considered, as it introduces additional nonlinearities. Although important, this last aspect will not be discussed in this article.

Keeping in mind the advantages of both analog and digital controllers, it is obvious that a digital system that could execute quasi-instantaneously a control algorithm should be of great interest by cumulating the advantages of both worlds. This leads to a third category of controllers: the quasi-analog controllers by digital means. An FPGA is a good candidate for this new category of controllers. But first, we shall give a short description of FPGAs and recall the main steps of an FPGA-based controller design. Then, it will be shown how FPGAs can significantly accelerate the processing time of an algorithm, thanks to their ability to implement a specific architecture that fits with the salient features of the algorithm to implement. This part is called an algorithm perspective.

In the last part of the article, a systematic analysis of the benefits due to the rapidity of an FPGA-based controller for power electronics and drive applications is given. Two cases are derived: the high-demanding applications case where the use of FPGA-based controllers with custom hardware architecture is mandatory and the constrained switching-frequency case where a relevant exploitation of the remaining time within each sampling period brings a value-added solution compared to standard software solution. Finally, the conclusions and future trends are given.

A Technological Perspective

Short FPGA Presentation

FPGAs belong to a wide family of programmable logic components. An FPGA is defined as a matrix of configurable logic blocks (CLBs; combinatorial and/or sequential), linked to each other by an interconnection network, which is entirely reprogrammable [2], [3]. The memory cells control the logic blocks as well as the connections so that the component can fulfill the required application specifications. Several configurable technologies exist. Among them, only those that are reprogrammable [Flash, static random access memory (SRAM)] are of interest here, since they allow the same flexibility as that of a standard software solution. Therefore, the rest of the article will discuss only the SRAM-based FPGA technology [4], [5], which is by far the most widespread technology. However, the flash-based technology [6] is of interest for some stringent niche applications such as space and aircraft industries. Indeed, the flash technology preserves the configuration of an FPGA when the power is off, and, as a consequence, the device is ready to operate as soon as it is powered up. Furthermore, the flash-based FPGAs guarantee the configuration against the single-event upset (SEU) radiations.

The most recent FPGAs are produced using a 40-nm copper process. Their density can reach more than 800,000 logic elements per component (roughly speaking, a logic element is also called a logic cell

An FPGA is defined as a matrix of configurable logic blocks, linked to each other by an interconnection network, which is entirely reprogrammable.

and is a combination of a 4–6-b look-up table and a flip-flop), with the available internal clocking resources of up to 1 GHz [4].

As can be seen in Figure 1, the generic architecture of an SRAM-based FPGA is composed of a matrix of CLBs, which consist of a cluster of logic cells (2–16, depending on the type of FPGA). This matrix of CLBs core is bordered by a ring of configurable input/output blocks (IOBs), whose number can reach 1,200 user IOBs. Finally, all these resources communicate among themselves through a programmable interconnection network.

More recently, a trend for a coarse-grained architecture has been observed, with the introduction of some dedicated blocks such as block random access memory (RAM), DSP accelerator units (hardwired multipliers with corresponding accumulators, high-speed clock management circuitry, serial transceivers), embedded hard-processor cores such as PowerPC or advanced reduced instruction set computing (RISC) machine (ARM) [4], [6], and soft-processor cores such as NiosII [5] or Microblaze [4], [7]. Moreover, an interesting feature of control applications is the recent integration of an ADC in the fusion component from Actel [8]. Thus, the original architecture based on a CLB matrix is now enriched by efficient blocks [DSP, memories, processor, digital clock manager (DCM), and ADC], making an FPGA a true system-on-chip (SoC) solution.

FPGAs are frequently used to implement complex functions. As already explained, this evolution has its origin in the recent advances in very large scale integration (VLSI), but it is also due to the development of appropriate design tools and methods, which were initially reserved to the world of the application-specific integrated circuits (ASICs). These tools are mostly based on hardware description languages (HDLs) such as very high-speed integrated circuits (VHSICs) HDL (VHDL) [9], [10] or Verilog [11]. The existence of IEEE standards [12] has spread the use of HDLs and has allowed the creation and development of high-performance computer-aided design (CAD) tools in the field of microelectronics. Thus, the designer can take advantage of HDLs to build his or her own circuit by using a hierarchical and modular approach defined at different levels of abstraction using the top–down methodology [13], [14]. The corresponding design flow is partitioned into the following four steps:

- *system level*, where the specifications of the circuit are given
- *behavior level*, which consists of the algorithmic description of the circuit
- *register transfer level (RTL)*, where the circuit is described in terms of its components
- *physical level*, where the circuit is physically described by taking into account the target hardware characteristics.

At each level of abstraction, the future integrated circuit is described in HDL, such as behavioral VHDL or synthesized VHDL. This last description gives an exact representation of the operators and variables of the final architecture.

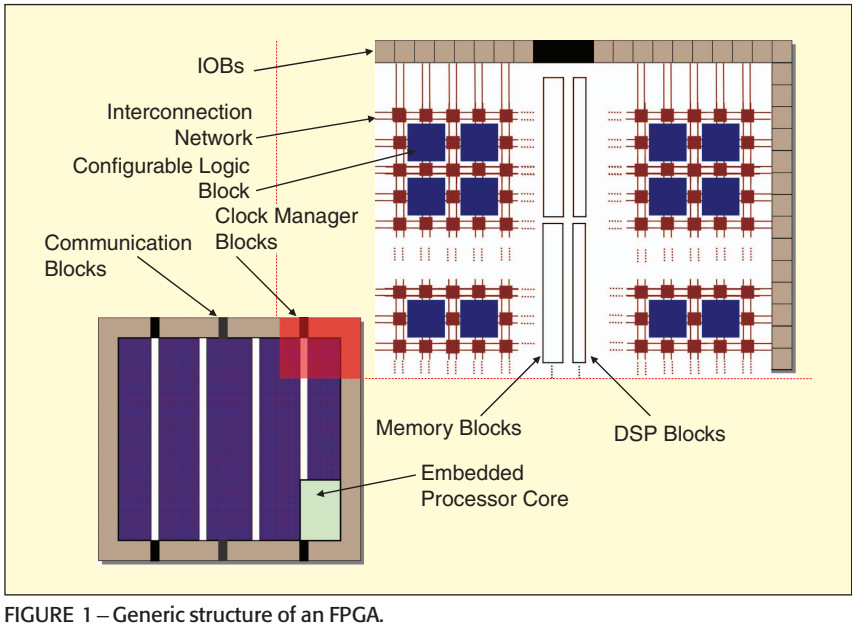


FIGURE 1 – Generic structure of an FPGA.

FPGA-Based Controller Design Consideration

The proposed design methodology is summarized in Figure 2. It is a balanced solution between two opposite needs: 1) a friendly method that is perfectly adapted to a control engineer who is not an expert in electronic digital design and 2) the consideration of good control performance requirements that necessarily leads to substantial efforts during the design of the hardware architecture. The main steps of this design methodology are now recalled and further details can be found in [15]. The first three steps [Figure 2(a)] are independent of the target:

- The modular partitioning step consists of partitioning the algorithm in reusable blocks that make sense from a functional point of view (e.g., proportional integral (PI) regulators). Doing so, the designer is able to capitalize his work along

the different projects he has to manage.

- The functional simulation step, where the synthesis of the controller is mostly validated in a time-continuous mode via the MATLAB-Simulink friendly environment.
- The digital redesign step that is crucial, since it includes the realization of the digital filter, the choice of the sampling period, and the choice of the fixed-point formats for coefficients and variables. In [16], an efficient design methodology is proposed to minimize the finite word length (FWL) effects on the behavior of the controller. It is shown that, when the sampling period is very short, a realization based on the delta operator [17] is recommended, since it is less sensitive to quantization effects and consumes less resources.

The fourth step is linked to the final target. In case of an FPGA-based implementation, it consists of the following:

- The data flow graph optimization, where the designer is modeling the data flow graph of the algorithm to get the best balance in terms of time/area performances. Methods of graph folding such as A^3 methodology [18] are perfectly adapted to this goal.
- The HDL coding of the obtained architecture is then done. The HDL code is the direct transcription of the resulting data flow graph.
- The following step is an FPGA implementation in itself. This is an automatic procedure to map, place, and route the design and analyze its static timing performances.
- Finally, experimental validation is made either on the final system or using a hardware-in-the-loop (HIL) procedure [19], [20].

Finally, it mentions that an HDL automatic code generation is also directly possible from Simulink when a rapid prototyping procedure is desired [21]. It corresponds to the blue arrow in Figure 2. This procedure is faster than the proposed methodology; however, several important optimization steps such as graph folding are omitted. This may alter the performance of the final architecture in terms of area.

Flash technology preserves the configuration of the FPGA when the power is off and, as a consequence, the device is ready to operate as soon as it is powered up.

An Algorithm Perspective

To accelerate the processing time of a control algorithm, a designer has two options when using a DSP controller: reducing the system clock period (this approach encounters technological limits) or programming in assembly code (this approach is time consuming). But this approach

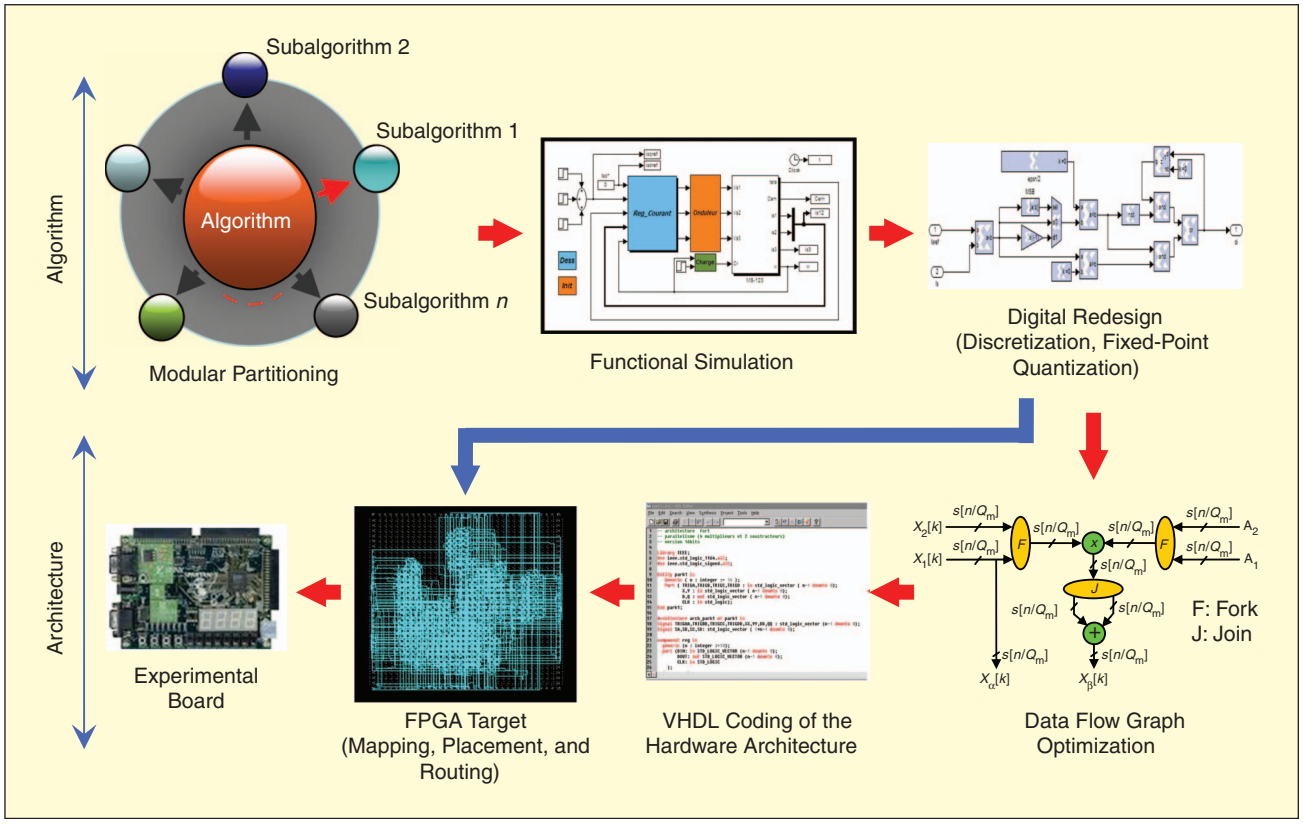


FIGURE 2 – Design methodology.

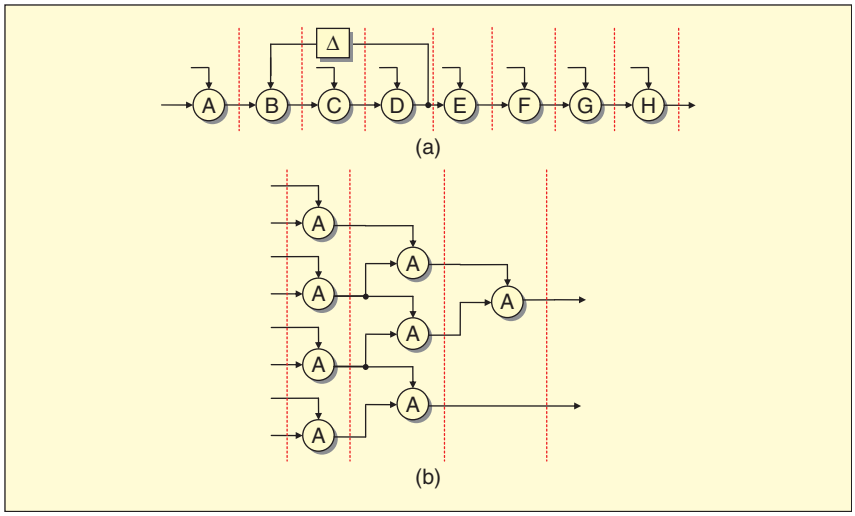


FIGURE 3 – Two opposite structures of algorithms. (a) First case. (b) Second case.

encounters technological limits or programming in assembly code which is quite time consuming.

Another way of seeing things is considering the possibility of designing a specific hardware architecture that matches the requirements of the algorithm to implement. In our opinion, this approach can significantly reduce the execution time of the control algorithm, if based on the former design methodology, without increasing too much design efforts.

Figure 3 illustrates the interest of this approach by considering two examples that have the same number of operations but opposite characteristics. In Figure 3(a), the proposed algorithm is fully serial (one has to wait for the result of a given operation before undertaking the next one). The relative complexity of this algorithm despite its low number of operations should be noted. Indeed, this algorithm integrates a loop, which generates a latency, and it includes a high level of heterogeneity (all the operations are different).

Figure 3(b) corresponds to a much more straightforward algorithm, with several possibilities of parallelism that, if they are integrated in the final hardware architecture of the controller, will dramatically accelerate the processing time. The last feature concerns the extreme regularity of this algorithm, since all of the eight operations are identical.

The direct confrontation of these two opposite cases clearly shows the relevancy to consider for implementation of a specific architecture for each algorithm: a) is serial and complex; therefore, it is well adapted to a general-purpose processor implementation and b) its parallelism and regularity makes this algorithm a good candidate for an FPGA-based implementation.

Figure 4 presents a synthesis of the preceding observations by putting in perspective the characteristics of the control algorithms and the ability of DSP and FPGA technologies to support their implementation in real time.

The *x* axis of the graph in Figure 4 represents the data-dependency level of the algorithm. The higher the dependency, the more sequential the algorithm [Figure 3(a)]. It is obvious that a software solution (DSP) is preferable in this case. On the other hand, when the considered algorithm includes a significant amount of concurrency between its operations [Figure 3(b)], it is clear that hardware solutions like FPGAs become most interesting.

However, as seen earlier, data dependencies are not enough to fully characterize a given algorithm. Its complexity also has to be considered. For this purpose, it is reported on the *y* axis of Figure 4. Algorithm complexity is evaluated in two different but complementary ways: the number of operations and the level of regularity of these operations. For example, the arithmetic logic unit (ALU) of a DSP integrates a multiply-and-accumulate (MAC) operator, since this kind of operation is very common in the digital filter algorithm. This makes DSP a powerful component for implementing algorithms that include a large number of MAC operations.

We shall end this description by recalling that the FPGA's ever-increasing density makes it possible to easily integrate one or two 32-b RISC

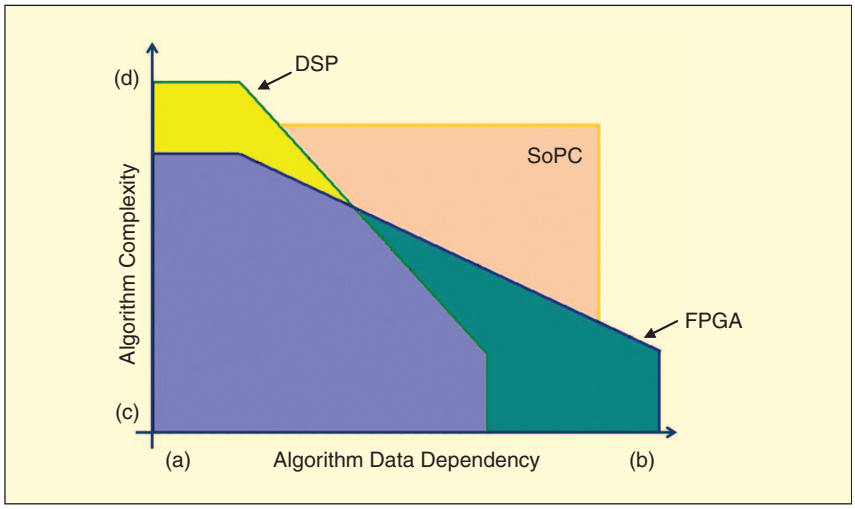


FIGURE 4 – DSP and FPGA domain of use for the digital control of power electronics and drive applications. (a) High data dependency. (b) High level of parallelism of the algorithm. (c) Few functions and/or homogeneous functions. (d) Lot of functions and/or heterogeneous functions.

microprocessors, leading to a complete system-on-programmable-chip solution. As can be seen in Figure 4, this approach is able to extend the ability of an FPGA to implement a larger diversity of control algorithms.

As shown in Figure 4, most of the control algorithms in the fields of power electronics and drive control applications belong to the purple zone, that is to say they can be implemented both in DSPs or FPGAs. This is mainly due to the moderate complexity (less than 1,000 basic arithmetic operations, medium range in terms of regularity) and because they are integrating a couple of parallelism possibilities (three-phase systems, *d-q* axis regulations).

Thus, the choice of the technology for implementing a control algorithm will be based on additional considerations. Cost is of course one of them, as well as the experience of the designer in software programming and/or digital electronics. However, despite their influences, we would like to focus on the control performances of the final application. Indeed, the constraints imposed by the application are also significant. Among them, the timing constraints imposed by the power converter losses are of prime importance. This matter will be thoroughly discussed in the next section.

An Application Perspective

Figure 5 presents a generic digital-controlled power system. It is generally composed of a source, load, power converter, and digital controller along with the corresponding interface modules [sensors, ADCs, pulsewidth modulation (PWM) timers]. For instance, in typical drive applications, the source is the main grid, the power converter is a back-to-back PWM inverter, and the load is a synchronous or induction motor. As for the digital controller, it is implemented in a DSP controller and/or an FPGA.

Main limitations in terms of dynamics (bandwidth) are, on one hand, because of the delays encountered in the digital loop [ADC time conversion, digital PWM (DPWM), and processing time] and on the other hand because of the limitation of the power converter switching frequency due to switching losses.

Algorithm complexity is evaluated in two different but complementary ways: the number of operations and the level of regularity of these operations.

Today, among all these limiting factors, the more important is the switching frequency of the static power converter. Indeed, the rapidity of digital controllers (DSP, FPGA) is now so high that the processing time is no longer the main limiting factor for the choice of the sampling period of current loops. Thus, the choice of the sampling frequency is most of the time conditioned by the maximum available switching frequency. Based on this analysis, two different cases can be identified:

- high-demanding applications where the switching frequency is above 100 kHz
- constrained switching-frequency applications where the switching frequency is below 100 kHz.

In the next two sections, the interest in using FPGA-based controllers is presented for both cases.

High-Demanding Applications

The first group, called high-demanding applications, consists of applications where timing constraints are so stringent that it is the digital controller that represents the main limitation of the whole control loop. As mentioned earlier, the choice of a high switching frequency (>100 kHz) for the power converter can be one of the reasons why the digital controller becomes the limiting factor of loop. This is because the sampling period used in the controller should be at least equal to the PWM switching period or to its half. But this obvious reason is not the only one. Let us check the different possibilities. The high-demanding application group can be classified into two subtypes.

The first subtype concerns the control of static converters where power is segmented to reduce the stress of the power switches. In these cases, concurrency is high, since several power channels have to be driven in parallel, and dedicated hardware architectures are much more efficient for implementing the corresponding digital controllers. These topologies of converters imply to concurrently

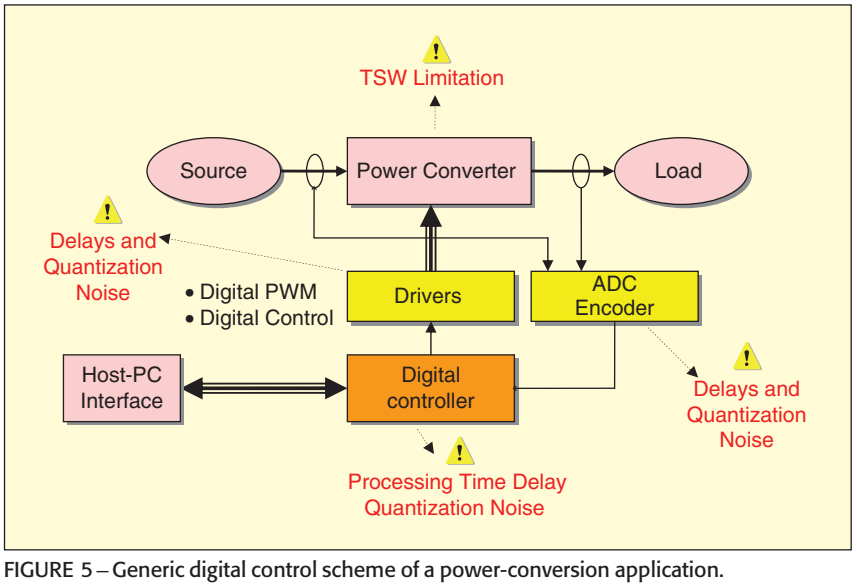


FIGURE 5 – Generic digital control scheme of a power-conversion application.

The choice of the sampling frequency is most of the time conditioned by the maximum available switching frequency.

control multiple power channels of the process [see Figure 6(a)]. The first presented example is an interleaved chopper for automotive application [22]. In this application, an FPGA-based controller has been designed to control two different 1-kW dc-dc buck converters of 16 and 36 phases, respectively, where the phase switching frequency is equal 150 kHz. The proposed controller can work at frequencies above 50 MHz, allowing a high duty cycle resolution, which in turn positively impacts the passive current sharing in a continuous conduction mode.

Another good example of highly parallel operations can be found in [23], where a new combined multiphase, multilevel (five phases and five levels) space vector PWM (SVPWM) strategy was presented and implemented with success in a low-cost FPGA.

The second subtype of high-demanding applications concerns applications where the sampling frequency is very high (at least equal or above 100 kHz), such as low-voltage, low-power switch-mode power supplies (SMPS). For such applications, the used switching frequency is often fixed above 1 MHz. In these cases, there is no option but using a hardware architecture to implement the control algorithm [see Figure 6(b)]. Direct digital synthesis for the output voltage controller is preferred [24], and great care is taken to enhance the timing resolution of the DPWM (here 2 ns) [25] to avoid limit cycles in steady state [26].

Very high switching frequency is no longer limited to low-power SMPS. Recently, Hartmann et al. [27] have presented an FPGA-based current controller for a 1-MHz, 10-kW three-phase Vienna rectifier. This application is interesting since it reveals the trend for more integration in embedded applications such as space and aircraft, leading to higher switching frequency while keeping a significant level of power.

Finally, we shall end this section devoted to high-demanding applications by another very challenging field. It concerns hardware and power-HIL applications. One of the most representative examples can be found in [28]. In this work, the authors present an FPGA-based real-time digital simulator, a three-level, 12-pulse voltage source inverter (VSI)-fed induction machine drive. It is worth mentioning that the VSI model is computed at a fixed time step of only 12.5 ns, allowing a realistic representation of the insulated-gate bipolar transistor (IGBT) nonlinear switching characteristics and power losses. All the models have been implemented using VHDL.

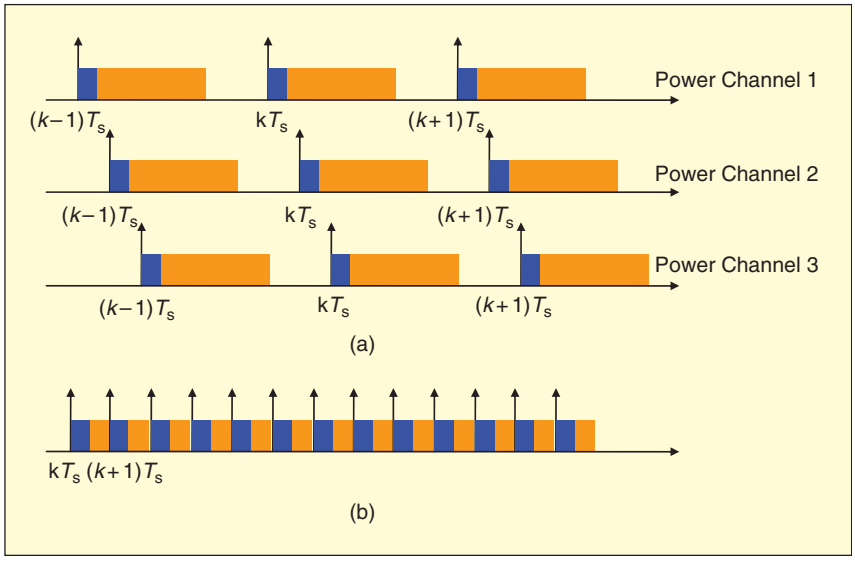


FIGURE 6 – High-demanding applications: (a) multipower channel controllers and (b) high-frequency applications.

Constrained Switching-Frequency Applications

The second group, called constrained switching-frequency applications, consists of applications where sampling is not critical due to switching frequency limitation (acceptable level of switching losses in the power converter). For this kind of application, a software implementation is possible. However, even in this case, using a dedicated hardware architecture can be of great interest, since the control processing time can be drastically reduced up to a fraction of the sampling period [see Figure 7(a)]. This velocity has an immediate influence on the quality of the control performances, especially, when direct control is chosen. In [15], Naouar et al. clearly showed the influence of processing time on the respect of hysteretic band for a current control of the ac drive (see Figure 8). These results demonstrate that the behavior of the proposed controller is very close to its analog counterpart. Besides, because of the ever-increasing density of the FPGA components, one can now implement in real time a significantly complex algorithm within a few microseconds. Thus, in [29], a fourth-order extended Kalman filter (EKF) for ac drives was proposed (this algorithm is one of the most demanding in this field in terms of computational resources: more than 1,000 basic arithmetic operations). For a Virtex II pro at 50-MHz clock frequency, the execution time of the whole algorithm does not exceed 6 μs (see Figure 9).

The quasi-analog behavior of digital controllers is also very much expected

when the power systems are controlled via a resonant controller. Resonant controllers are based on the use of generalized integrators [30]. Generalized integrators supply infinite gain at the resonant frequency, leading to a zero steady-state error when regulating *abc* or $\alpha\beta$ stationary-frame sinusoidal signals. In [31], the authors have shown that this type of controller allows obtaining similar performances compared with standard *dq* synchronous frame PI controllers but without the need of any *dq*/*abc* transformation.

Active filters are of course good candidates for this type of controller, especially, when they are controlled with harmonic selection-compensation techniques [32], [33]. More recently there appears another important set of applications that take advantage of the use of generalized integrator-based controllers. It consists in the control of the grid-side power converter of distributed power generation systems (DPGS) based on renewable energies. In these applications, the control tasks devoted to the grid-side power converter are mainly the dc-link voltage control to achieve the power balance between the power source and grid, the quality of the generated power (control of the grid side currents), and the synchronization with the grid voltage [30]. As shown in [34], the synchronization task can be quite challenging when the positive- and negative-sequence components of the power signal at the fundamental frequency and its harmonics are simultaneously tracked.

For both types of applications (active filtering and grid-connected DPGS), one can note that the trends are the use of more and more generalized integrator-based controllers put in parallel. From an implementation point of view, it has been shown in [35] and [36] that this kind of controller is very sensitive to FWL effects on the coefficients and variables of the controller. One of the major aspects to consider here is that the sampling frequency is very high compared with the resonant frequency of the controller. In this context, the designer

The quasi-analog behavior of digital controllers is also very much expected when the power systems are controlled via a resonant controller.

is invited to use the delta operator instead of the classical shift operator to implement generalized ac integrator-based controllers. In doing so, much more well-conditioned implementations are obtained. The features of the implemented digital controller are quasi-equivalent to the ones of the original analog-generalized integrator-based controller, using only a reasonable number of bits for coding coefficients and variables [35].

Up to now, most of the proposed implementations were made on a DSP-controller target. Typical results with this technology are a compensation of up to the 15th harmonic with a processing time of approximately 30 μs. No doubt that the use of dedicated FPGA-based hardware architecture

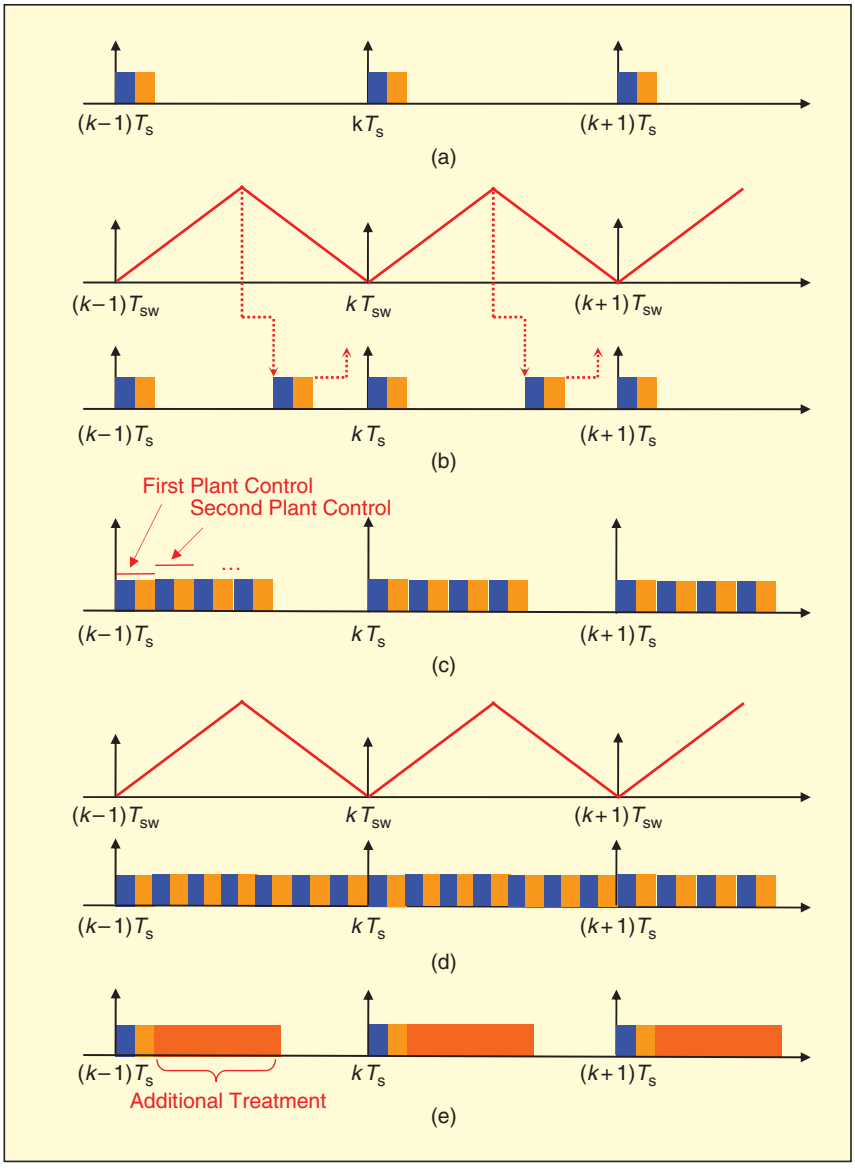


FIGURE 7 – Constrained switching-frequency applications: (a) quasi-analog behavior, (b) synchronization between the measurement unit and PWM unit, (c) shared control resources, (d) oversampling, and (e) value-added additional treatment.

An interesting feature of control applications is the integration of an ADC in the fusion component from Actel.

could significantly reduce the processing time and help to track more harmonics. Finally, another interesting alternative is the use of repetitive-based control [37]. But as shown in [38], the processing time in this case is significantly higher than the case of a set of generalized integrator-based controllers. This is due to the computing of a discrete cosine-transform filter. Once again, an FPGA-based controller could be advantageously used for implementing a repetitive-based controller.

But, the reduction of processing time due to FPGA-based implementation Figure 7(a), beyond its immediate improvement of the closed-loop bandwidth (quasi-analog behavior), is also interesting for at least five other reasons.

1) The first reason is the possibility to simplify the complexity of the control algorithm by integrating direct feed-forward compensation of measured disturbances given in [39], where the 100-Hz oscillations due to the rectified grid voltage are significantly reduced by using this technique. This example is also interesting since it includes a division. Indeed, another advantage of an FPGA-based controller over a standard software solution can be underlined here, since it consists of the implementation of a hardware divider.

   The rapidity of the controller can also help to reduce its complexity. In [40], this concept is applied to the design of a fault-tolerant controller. Authors present a simple and efficient FPGA-based real-time power converter failure diagnosis for wind-energy conversion systems. It is based on the combination of a time and magnitude threshold, which allows to significantly reduce the time for default detection (less than 10 μs), where standard solutions need at least one quarter of the fundamental period.
2) The second reason is that the significant reduction of processing time obtained by the use of FPGA-based controllers can allow an accurate synchronization process between the voltage and current sample measurement unit and the PWM carrier [see Figure 7(b)]. Thus, in [41], Blaabjerg et al. present an FPGA-based implementation of a switched reluctance motor (SRM) current control without antialiasing filters. Indeed, by choosing to sample current in the center of a symmetrical modulation, an exact measure of the average current is obtained without any additional filter. Dynamics of the current loop is then increased.

   Synchronization of the current sample measurement unit and PWM unit can also be advantageously used in sensorless control

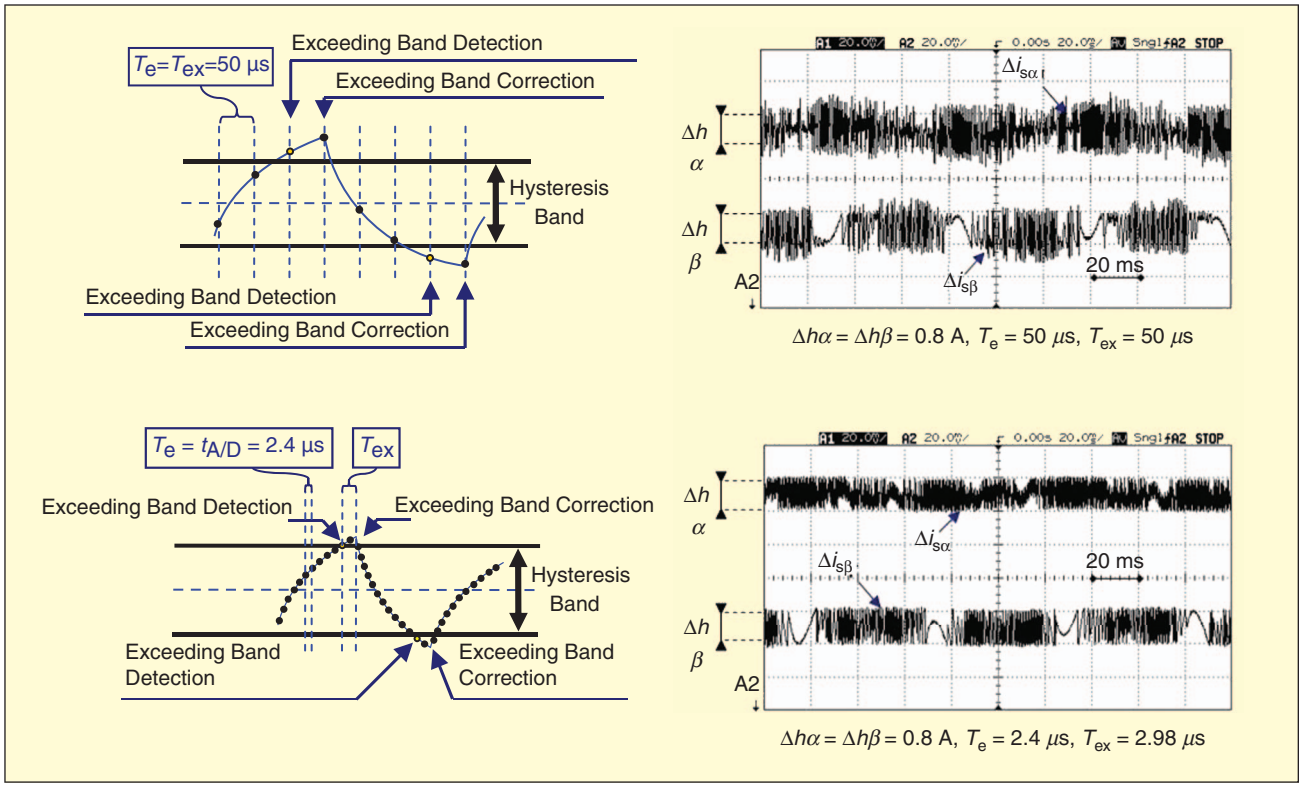
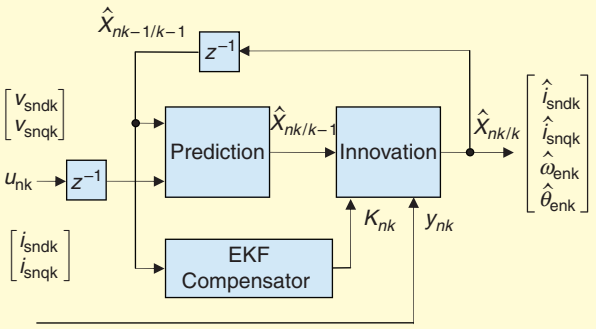
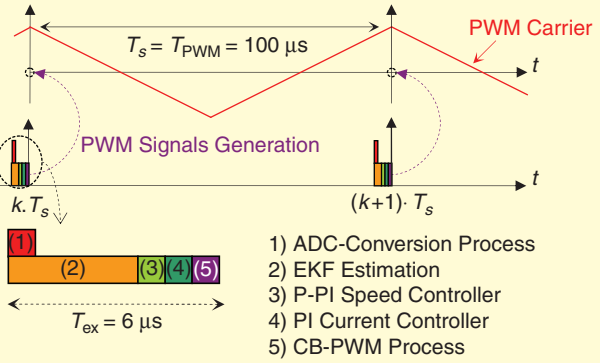


FIGURE 8 – Digital quasi-analog hysteretic controller for synchronous motor drives.



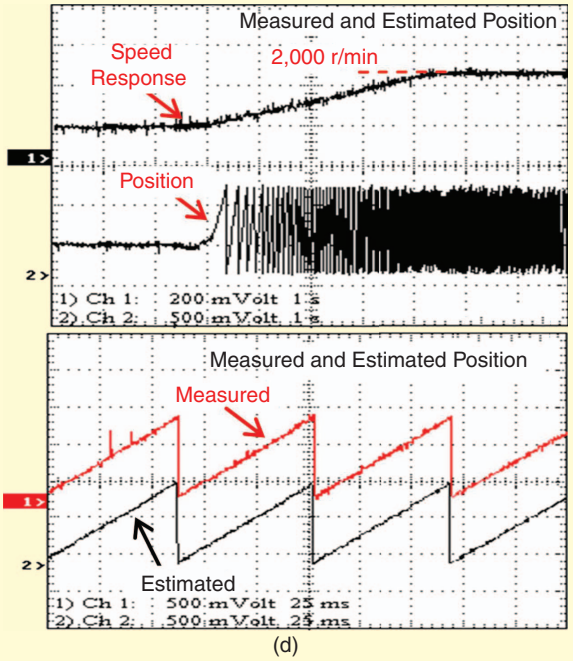
(a)



(b)

EKF Modules		EKF Complexity			
Prediction Module		×	+	–	1/*x*
$\hat{X}_{nk/k-1} = \hat{X}_{nk-1/k-1} + T_s \cdot f(\hat{X}_{nk-1/k-1}, u_{nk-1})$		10	6	0	0
EKF Compensator–Kahnan Gain Calculation		×	+	–	1/*x*
Jacobian Matrix	$Fd_{nk} = \left.\frac{\partial(\hat{X}_{nk-1} + T_s \cdot f(\hat{X}_{nk-1}, u_{nk-1}))}{\partial X_n}\right\|_{X_n = \hat{X}_{nk-1/k-1}}$	4	1	0	0
Covariance Matrix Prediction	$P_{nk-1/k-1} = Fd_{nk} \cdot P_{nk-1/k-1} \cdot Fd_{nk} + Q$ Initial Condition P_0	318	244	16	1
Kalman Gain Calculation	$K_{nk} = P_{nk/k-1} \cdot H^t \cdot (H\, P_{nk/k-1} H^t + R)^{-1}$				
Updating Covariance Matrix	$P_{nk/k} = P_{nk/k-1} - K_{nk} H\, P_{nk/k-1}$				
Innovation Module		×	+	–	1/*x*
$\hat{X}_{nk/k} = \hat{X}_{nk/k-1} + K_{nk}(y_{nk} - H\, \hat{X}_{nld\,k-1})$		8	8	8	0
Total		340	259	24	1

(c)



(d)

FIGURE 9 – FPGA-based EKF for sensorless synchronous motor drives. (a) Synoptic of the EKF module, (b) timing diagram, (c) EKF equations and complexity evaluation, and (d) experimental results: waveforms of the estimated speed and position.

The designer can take advantage of HDLs to build his or her own circuit by using a hierarchical and modular approach defined at different levels of abstraction using the top-down methodology.

of ac drives with the indirect flux detection by online reactance measurement (INFORM) method [42]. Such a method requires a noise-free measurement of current derivative term that is not easy to obtain. Because of their ability to achieve a very accurate synchronization between the current sample measurement unit and PWM unit, dedicated FPGA-based hardware architectures can be of great interest for such kinds of methods [43].

3) The third reason is that the remaining time within each sampling period can be used for controlling several similar plants with a unique controller. Thus, in [44], the authors have proposed an FPGA-based controller that is able to control up to four ac drives with a sampling period of only 50 μs [see Figure 7(c)]. Thanks to the ever-increasing density of the components, this pioneering work in the field of centralized control of multimotor applications is now finding exciting extension with the use of a system-on-programmable-chip solution like the one presented in [45].
4) The fourth reason relies on the possibility to use oversampling strategies [Figure 7(d)]. Thus, some of the authors have proposed increasing the sampling frequency of current- or voltage-control loops. In [46], Chapuis et al. propose a quasi-analog digital direct torque control of an induction motor, where the torque regulation is updated every 2 μs however, a protection module is added to prevent too frequent switching on the same inverter leg. In [47], Corradini et al. present a multisampled control strategy for high-switching frequency SMPS, which significantly reduces the delay introduced by the small-signal digital PWM. The derived aliasing effect is compensated by a repetitive controller to avoid any additional delay due to filtering action.

Beyond increasing the sampling frequency of control loops, oversampling can also be used to improve the quality of the measurement of quantities. For example, in [48], De Castro et al. proposed an oversampled strategy for measuring the mean value of the input current of a flyback converter to implement a digital charge control algorithm for power factor correction. Along the same line, in [49], Fratta et al. presented a quasi-ideal direct measure of the mean value of the current. It consists of a moving average filter that is able to cancel all PWM harmonics. In this work, the proposed practical sampling rate is equal to eight. More recently, some authors have also pointed out the interest of using delta sigma ($\Delta\Sigma$) ADCs for measuring currents in power applications. Such ADCs are known for their high resolution, but up to now their use was only limited to slow down dynamic applications such as audio. Indeed, the increase of the resolution is counterbalanced by the increase of the oversampling rate [50]. However, with a clock frequency of tens of megahertz, current generation of $\Delta\Sigma$ ADCs is now a good candidate for power applications too. In [51], Peters et al. presented a regular-sampled current measurement based on delta-sigma modulators for ac drives. In this article, the currents are measured through a second-order delta-sigma modulator associated to a Sinc3 filter with a decimation ratio of 32. Besides, the proposed filter is also synchronized with the PWM carrier to obtain high-quality true average mean currents.

5) The fifth and last reason is the possibility to add new control functionalities within this remaining time of each sampling period [see Figure7(e)]. From this perspective, a real-time processing extension that naturally comes to mind is the implementation of a predictive control strategy like the one presented in [52] for synchronous motor drives.

Another exciting challenge is the addition of health-monitoring processing. Very promising works have already been proposed for SMPS applications, such as in [53], where an online identification strategy through cross correlation was successfully implemented. Such a type of identification is also used for the autotuning of the digital controller [54].

In the same vein, diagnosis algorithms can also be implemented. To this purpose, [55] and [56] have proposed wavelet-based diagnosis algorithms for induction motor broken-rotor-bar detection and fault detection and classification in transmission lines, respectively. In both cases, a low-cost FPGA was used for the implementation.

Conclusions and Future Trends

To conclude, FPGA-based customized digital control platforms are definitely an attractive solution for implementing evermore demanding power converter digital control systems. Among the new requirements in power electronics that have a direct impact on digital controllers, we can mention the integration, segmentation of the power, and reliability. Indeed, the integration level is becoming higher and higher, especially for embedded applications where expectation for volume and weight reductions is important. Doing so, a significant reduction of the

size of passive filter can be achieved. An immediate consequence is the increase of the switching frequency of the power converters, which in turn implies a shorter sampling period for the controller. The goal pursued by the segmentation of power is the reduction of the stress of power switches via interleaved and multilevel structures. Such approach introduces a significant level of parallelism in the system that once again is perfectly adapted to FPGA-based controllers.

Finally, a higher degree of reliability is required. As seen before, health monitoring or diagnosis processing tasks can be quite easily added in an FPGA-based controller when the constraints on the switching frequency are important.

Beyond these three cases, it has been shown that FPGA-based controllers can be an efficient option either for the high-demanding applications or the switching-frequency constrained applications. The first group corresponds to high switching-frequency, high level of parallelism, and HIL applications. Regarding the second group for which the switching frequency is limited, it has been shown that we can take advantage of the rapidity of an FPGA-based controller for designing a quasi-analog controller, for reducing the complexity of the controller, perfectly synchronizing the measurement unit and the DPWM unit, sharing the controller resources among several plants, for implementing oversampling control strategies and including additional tasks to the control, such as prediction, health monitoring, and diagnosis.

In the near future, the complexity of the digital control systems will continue to grow, and the tasks devoted to the control algorithm will no longer be limited to regulation but have to manage electromagnetic interference (EMI), communications, health monitoring, diagnosis, and fault-adaptive online control. As a consequence and from an algorithm point of view, signal-processing functions for power electronics applications will be the center of intensive research. From this perspective, an FPGA-based controller represents a very attractive solution.

We conclude by shortly introducing the main challenges in terms of architecture, this aspect being closely related to the algorithm and at the same time offers in itself a degree of creativity. Among the most exciting trends, we can quote the introduction of the floating-point computing evaluation. What is its cost in terms of hardware resources? Is it necessary for power electronics applications where the dynamic range of variables and the complexity of the control algorithm are still limited? Another interesting field of research is the codesign approach, with questions like how to make an optimized partitioning between the software part of the controller architecture and its hardware part?

Another issue concerns mixed integration. Indeed, as seen before and due to their rapidity, FPGA-based controllers are now introducing only a small delay in the loop and because their high density rate can easily work with a high accuracy (high number of bits). This makes the controller quasi-transparent, moving the limitations of time and accuracy toward the interfaces like ADCs, DPWM, and drivers. On this basis, for very demanding applications like SMPS or aircraft, it makes sense to increase the performances to work on mixed A/D integrated solutions.

Finally, we shall finish this enumeration by reminding that SRAM-based FPGA architecture can be reconfigured on the fly. This feature has to be explored keeping in mind fault detection and reconfiguration controller or optimized architecture. In this case, the choice of the operating system is crucial.

Biographies

Eric Monmasson (eric.monmasson@u-cergy.fr) is currently a full professor and head of the Institut Universitaire Professionnalisé de Génie Electrique et d'Informatique Indutrielle (IUP GEII), University of Cergy-Pontoise in France. He is also with the Systèmes et Applications des Technologies de l'Information et de l'Energie Laboratory (SATIE, UMR CNRS8029). He is the chair of the technical committee on electronic SoC of the IEEE Industrial Electronics Society. He is also a member of the steering committee of the European Power Electronics Association and on the number one technical committee of the International Association for Mathematics and Computers in Simulation (IMACS). He is an associate editor of *IEEE Transactions on Industrial Electronics*. He is the author or coauthor of two books and more than 100 scientific papers. He is a Senior Member of the IEEE. His current research interests include advanced control of electrical motors and generators and the use of FPGAs for energy control systems.

Lahoucine Idkhajine received an M.S. degree in electrical engineering from IUP-GEII, University of Cergy-Pontoise, France, in 2007. He has been preparing for a Ph.D. degree since October 2007 in SATIE Laboratory of the University of Cergy-Pontoise. He is a Student Member of the IEEE. His research work is devoted to the digital implementation, using FPGA devices, of controllers and observers for ac motor drives.

Mohamed Wissem Naouar received his B.S. and M.S. degrees in electrical engineering from the Ecole Nationale d'Ingénieurs de Tunis (ENIT), Tunisia, in 2003 and 2004, respectively. In 2007, he received his Ph.D. degree in electrical engineering from ENIT and the University of Cergy Pontoise. In 2008, he became an associate professor in the Electrical Engineering Department of ENIT. He is a Member of the IEEE. His research interests include the advanced control of electrical systems and the use of FPGAs for industrial control systems.

References

[1] H. A. Toliyat and S. G. Campbell, *DSP-Based Electromechanical Motion Control*. Boca Raton: CRC Press, 2004.

[2] E. Monmasson and M. Cirstea, "FPGA design methodology for industrial control systems—A review," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1824–1842, Aug. 2007.

[3] J. J. Rodriguez-Andina, M. J. Moure, and M. D. Valdes, "Features, design tools, and application domains of FPGAs," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1810–1823, Aug. 2007.

[4] (2009). *Xilinx Data Book*, [Online]. Available: www.xilinx.com
[5] (2009). *Altera Data Book*, [Online]. Available: www.altera.com
[6] (2009). *Actel Data Book*, [Online]. Available: www.Actel.com
[7] H. Calderon, C. Elena, and S. Vassiliadis, "Soft core processors and embedded processing: A survey and analysis," in *Proc. Safe ProRisc Workshop*, 2005, pp. 483–488, CD-ROM.
[8] L. Idkhajine, E. Monmasson, M. W. Naouar, A. Prata, and K. Bouallaga, "Fully integrated FPGA-based controller for synchronous motor drive," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4006–4017, Oct. 2009.
[9] D. L. Perry, *VHDL*. New York: McGraw-Hill, 2004.
[10] P. J. Ashenden, *The Designer's Guide to VHDL*. San Mateo, CA: Morgan Kaufmann, 1995.
[11] S. Palnitkar, *Verilog HDL: A Guide to Digital Design and Synthesis*. Englewood Cliffs, NJ: Prentice Hall, 1996.
[12] *IEEE Standard VHDL Language Reference Manual*, IEEE Standard 1076-1993, 1993.
[13] A. A. Jerraya, H. Ding, P. Kission, and M. Rahmouni, *Behavioral Synthesis and Component Reuse with VHDL*. Norwell, MA: Kluwer, 1998.
[14] 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.
[15] M.-W. Naouar, E. Monmasson, A. A. Naassani, I. Slama-Belkhodja, and N. Patin, "FPGA-based current controllers for AC machine drives—A review," *IEEE Trans. Ind. Electron.*, vol. 54, no. 4, pp. 1907–1925, Aug. 2007.
[16] F. Zhengwei, J. E. Carletta, and R. J. Veillette, "A methodology for FPGA-based control implementation," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 6, pp. 977–987, Nov. 2005.
[17] R. H. Middleton and G. C. Goodwin, *Digital Control and Estimation—A Unified Approach*. Englewood Cliffs, NJ: Prentice-Hall, 1990.
[18] T. Grandpierre, C. Lavrenne, and Y. Sorel, "Optimized rapid prototyping for real-time embedded heterogeneous multiprocessor," in *Proc. CODES'99 7th Int. Workshop Hardware/Software Co-Design Conf.*, 1999. CD-ROM.
[19] O. Lucia, O. Jimenez, L. A. Barragan, I. Urriza, J. M. Burdio, and D. Navarro, "Real-time FPGA-based hardware-in-the-loop development test-bench for multiple output power converters," in *Proc. IEEE APEC'10 Conf.*, 2010, pp. 309–314.
[20] C. Dufour, J. Belanger, S. Abourida, and V. Lapointe, "FPGA-based real-time simulation of finite-element analysis permanent magnet synchronous machine drives," in *Proc. IEEE PESC'07 Conf.*, 2007, pp. 909–915.
[21] F. Ricci and H. Le-Huy, "An FPGA-based rapid prototyping platform for variable-speed drives," in *Proc. IEEE IECON'02 Conf.*, 2002, pp. 1156–1161.
[22] O. Garcia, P. Zumel, A. de Castro, J. A. Cobos, and J. Uceda, "Automotive DC–DC bidirectional converter made with many interleaved buck stages," *IEEE Trans. Power Electron.*, vol. 21, no. 3, pp. 578–586, May 2006.
[23] O. Lopez, J. Álvarez, J. Doval-Gandoy, and F. D. Freijedo, "Multilevel multiphase space vector PWM algorithm," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 1933–1942, May 2008.
[24] B. J. Patella, A. Prodic, A. Zirger, and D. Maksimovic, "High-frequency digital controller IC for dc–dc converters," *IEEE Trans. Power Electron.*, vol. 18, no. 1, pp. 438–446, Jan. 2003.
[25] S. C. Huerta, A. de Castro, O. Garcia, and J. A. Cobos, "FPGA-based digital pulsewidth modulator with time resolution under 2 ns," *IEEE Trans. Power Electron.*, vol. 23, no. 6, pp. 3135–3141, Nov. 2008.
[26] A. V. Peterchev and S. R. Sanders, "Quantization resolution and limit cycling in digitally controlled PWM converters," *IEEE Trans. Power Electron.*, vol. 18, no. 1, pp. 301–308, Jan. 2003.
[27] M. Hartmann, S. D. Round, H. Ertl, and J. W. Kolar, "Digital current controller for a 1 MHz, 10 kW three-phase VIENNA rectifier," *IEEE Trans. Power Electron.*, vol. 24, no. 11, pp. 2496–2508, Nov. 2009.
[28] A. Myaing and V. Dinavahi, "FPGA-based real-time emulation of power electronic systems with detailed representation of device characteristics," *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 358–368, Jan. 2011.
[29] L. Idkhajine, E. Monmasson, and A. Maalouf, "A fully FPGA-based sensorless control for AC drive using an extended Kalman filter," in *Proc. IEEE IECON'09 Conf.*, 2009, pp. 2925–2930.
[30] A. Timbus, M. Liserre, R. Teodorescu, P. Rodriguez, and F. Blaabjerg, "Evaluation of current controllers for distributed power generation systems," *IEEE Trans. Power Electron.*, vol. 24, no. 3, pp. 654–664, Mar. 2009.
[31] D. N. Zmood and D. G. Holmes, "Stationary frame current regulation of PWM inverters with zero steady-state error," *IEEE Trans. Power Electron.*, vol. 18, no. 3, pp. 814–822, Mar. 2003.
[32] L. Limongi, R. Bojoi, G. Griva, and A. Tenconi, "Digital current-control schemes," *IEEE Ind. Electron. Mag.*, vol. 3, no. 1, pp. 20–31, Mar. 2009.
[33] J. Miret, M. Castilla, J. Matas, J. M. Guerrero, and J. C. Vasquez, "Selective harmonic-compensation control for single-phase active power filter with high harmonic rejection," *IEEE Trans. Ind. Electron.*, vol. 56, no. 8, pp. 3117–3127, Aug. 2009.
[34] P. Rodriguez, A. Luna, I. Etxeberria, J. R. Hermoso, and R. Teodorescu, "Multiple second order generalized integrators for harmonic synchronization of power converters," in *Proc. IEEE ECCE'09 Conf.*, pp. 2239–2246.
[35] M. J. Newman and D. G. Holmes, "Delta operator digital filters for high performance inverter applications," *IEEE Trans. Power Electron.*, vol. 18, no. 1, pp. 447–454, Jan. 2003.
[36] L. Harnefors, "implementation of resonant controllers and filters in fixed-point arithmetic," *IEEE Trans. Ind. Electron.*, vol. 56, no. 4, pp. 1273–1281, Apr. 2009.
[37] P. Mattavelli and F. P. Marafão, "Repetitive-based control for selective harmonic compensation in active power filters," *IEEE Trans. Ind. Electron.*, vol. 51, no. 5, pp. 1018–1024, Oct. 2004.
[38] A. Rodríguez, C. Girón, M. Rizo, V. Sáez, E. Bueno, and F. J. Rodríguez, "Comparison of current controllers based on repetitive-based control and second order generalized integrators for active power filters," in *Proc. IEEE IECON Conf.*, 2009, pp. 3223–3228.
[39] E. Mamarelis, C. Andres Ramos-Paja, G. Petrone, G. Spagnuolo, M. Vitelli, and R. Giral, "FPGA-based controller for mitigation of the 100 Hz oscillation in grid connected PV systems," in *Proc. IEEE ICIT'10 Conf.*, 2010, pp. 925–930, CD-ROM.
[40] S. Karimi, A. Gaillard, P. Poure, and S. Saadate, "FPGA-based real-time power converter failure diagnosis for wind energy conversion systems," *IEEE Trans. Ind. Electron.*, vol. 55, no. 12, pp. 4299–4308, Dec. 2008.
[41] F. Blaabjerg, P. C. Kjaer, P. O. Rasmussen, and C. Cossar, "Improved digital current control methods in switched reluctance motor drives," *IEEE Trans. Power Electron.*, vol. 14, no. 3, pp. 563–572, May 1999.
[42] M. Schroedl, "Sensorless control of AC machines at low speed and standstill based on the 'INFORM' method," *IEEE-IAS Conf. Rec.*, 1996, vol. 1, pp. 270–277.
[43] D. Samuelsen, "AC machine control robust and sensorless control by parameter independency," Ph.D. dissertation, Narvik Univ. College, Norway, 2009.
[44] K. Tazi, E. Monmasson, and J. P. Louis, "Description of an entirely reconfigurable architecture dedicated to the current vector control of a set of AC machines," in *Proc. IEEE-IECON'99 Conf.*, pp. 1415–1420.
[45] Y.-S. Kung, R.-F. Fung, and T.-Y. Tai, "Realization of a motion control IC for *x-y* table based on novel FPGA technology," *IEEE Trans. Ind. Electron.*, vol. 56, no. 1, pp. 43–53, Jan. 2009.
[46] Y. A. Chapuis, J. P. Blonde, and F. Braun, "FPGA implementation by modular design reuse mode to optimize hardware architecture and performance of AC motor controller algorithm," in *Proc. EPE-PEMC Conf.*, 2004, CD-ROM
[47] L. Corradini, P. Mattavelli, E. Tedeschi, and D. Trevisan, "High bandwidth multisampled digitally controlled DC/DC converters using ripple compensation," *IEEE Trans. Ind. Electron.*, vol. 55, no. 4, pp. 1501–1508, Apr. 2008.
[48] A. De Castro, P. Zumel, O. Garcia, T. Riesgo, and J. Uceda, "Concurrent and simple digital controller of an AC/DC converter with power factor correction based on an FPGA," *IEEE Trans. Power Electron.*, vol. 18, no. 1, pt. 2, pp. 334–343, Jan. 2003.
[49] A. Fratta, G. Griffero, and S. Nieddu, "Comparative analysis among DSP and FPGA-based control capabilities in PWM power converters," in *Proc. IEEE IECON Conf.*, 2004, vol. 1, pp 257–262.
[50] P. M. Aziz, H. V. Sorensen, and J. Van Der Spiegel, "An overview of sigma-delta converters," *IEEE Signal Processing Mag.*, vol. 13, no. 1, pp. 61–84, Jan. 1996.
[51] W. Peters, B. Schulz, S. Mathapati, and J. Bocker, "Regular-sampled current measurement in AC drives using delta-sigma modulators," in *Proc. EPE'09 Conf.*, 2009, pp. 1–9.
[52] M. W. Naouar, A. A. Naassani, E. Monmasson, and I. Slama-Belkhodja, "FPGA-based predictive current for synchronous machine speed drive," *IEEE Trans. Power Electron.*, vol. 23, no. 4, pp. 2115–2126, July 2008.
[53] B. Miao, R. Zane, and D. Maksimovic, "System identification of power converters with digital control through cross-correlation methods," *IEEE Trans. Power Electron.*, vol. 20, no. 5, pp. 1093–109, Sept. 2005.
[54] M. Shirazi, J. Morroni, A. Dolgov, R. Zane, and D. Maksimovic, "Integration of frequency response measurement capabilities in digital controllers for DC–DC converters," *IEEE Trans. Power Electron.*, vol. 23, no. 5, pp. 2524–2535, Sept. 2008.
[55] A. Ordaz-Moreno, R. De Jesus Romero-Troncoso, J. A. Vite-Frias, J. R. Rivera-Gillen, and A. Garcia-Perez, "Automatic online diagnosis algorithm for broken-bar detection on induction motors based on discrete wavelet transform for FPGA implementation," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 2193–2202, May 2008.
[56] P. Simi Valsan and K. Shanti Swarup, "High-speed fault classification in power lines: Theory and FPGA-based implementation," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1793–1800, May 2009.