

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

FPGA-Based Implementation of Dual Kalman Filter for PV MPPT Applications

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:

Ricco, M., Manganiello, P., Monmasson, E., Petrone, G., Spagnuolo, G. (2017). FPGA-Based Implementation of Dual Kalman Filter for PV MPPT Applications. IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, 13(1), 176-185 [10.1109/TII.2015.2462313].

Availability:

This version is available at: <https://hdl.handle.net/11585/676422> since: 2019-12-13

Published:

DOI: <http://doi.org/10.1109/TII.2015.2462313>

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>).
When citing, please refer to the published version.

(Article begins on next page)

This is the final peer-reviewed accepted manuscript of:

M. Ricco, P. Manganiello, E. Monmasson, G. Petrone and G. Spagnuolo, "FPGA-Based Implementation of Dual Kalman Filter for PV MPPT Applications" in IEEE Transactions on Industrial Informatics, vol. 13, no. 1, pp. 176-185, Feb. 2017

The final published version is available online at:

<https://doi.org/10.1109/TII.2015.2462313>

Rights / License:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>)

When citing, please refer to the published version.

FPGA-Based Implementation of Dual Kalman Filter for PV MPPT Applications

Mattia Ricco, Patrizio Manganiello, Eric Monmasson, *Senior Member, IEEE*, Giovanni Petrone, *Member, IEEE* and Giovanni Spagnuolo, *Senior Member, IEEE*

Abstract—The way of implementing an adaptive maximum power point tracking algorithm for photovoltaic applications in a Field Programmable Gate Array is described in this paper. A dual Kalman filter allows estimating the settling time of the whole system, including the PV source and the dc/dc converter controlling the operating point thereof, so that the tracking algorithm self adapts its parameters to the actual weather conditions. The real-time identification need of this application requires a FPGA platform, so that the intrinsic algorithm parallelism is exploited and the execution time is reduced. The tradeoff solutions proposed in this paper, accounting for the algorithm complexity and the limited FPGA hardware, as well as some solutions for optimizing the implementation are described. The proposed adaptive algorithm is implemented in a low-cost Xilinx Spartan-6 FPGA and it is validated through experimental tests.

Index Terms—Field Programmable Gate Array, adaptive MPPT controller, Dual Kalman Filter, Photovoltaic system

I. INTRODUCTION

The Perturbe & Observe (P&O) algorithm is one of the most common Maximum Power Point Tracking (MPPT) methods applied in PhotoVoltaic (PV) systems to track the Maximum Power Point (MPP) in any environmental condition. Such a technique consists in perturbing the PV voltage by applying a step change to the converter duty-cycle d and in measuring the corresponding PV output power P . Then, the next perturbation is evaluated by means of the following control law [1]:

$$d_{(k+1)T_p} = d_{kT_p} + \Delta d \cdot \text{sign}(P_{kT_p} - P_{(k-1)T_p}) \quad (1)$$

where Δd is the perturbation amplitude and T_p is the perturbation period, i.e. the time interval between two consecutive perturbations. In [1] an in-depth discussion about the dependency of the MPPT performance with respect to

these two parameters is given. Such analysis leads to the following general considerations: as concerns the perturbation amplitude, a too low value makes the algorithm too sensible to the measurement errors and disturbances introduced by the switching converter. Whereas a too high value increases the oscillations around the MPP reducing the whole MPPT efficiency. Regarding the perturbation period, a too low value might be not enough to ensure the steady state regime for the switching converter, thus leading to an unpredictable MPPT behaviour. On the other hand, a too high value of T_p slows down the MPPT controller, worsening the MPPT efficiency. This is particularly true during cloudy days where a prompt reaction of the MPPT allows tracking the MPP fastly. As a consequence the MPPT parameters are usually designed by considering the worst case conditions and according to the guidelines given in [1]. Nevertheless, the on-line optimization of these two MPPT parameters allows improving the tracking performance by an adaptation to the time varying PV operating conditions. Several authors focus their effort on the optimization of the perturbation amplitude of perturbative MPPT methods, not only for PV applications [2] [3], but also for other renewable energy systems [4] [5]. Instead, only in [6] an on-line optimization of the perturbation period, based on a non-parametric identification approach, has been proposed.

In this paper the Dual Kalman Filter (DKF), which is a parametric identification technique, is used to the aim of adapting the perturbation period. The identification of the PV parameters allows estimating the settling time of the PV system, which includes the PV source and the dc/dc converter actuating the MPPT action, and then, to choose adequately the MPPT perturbation period. Thus, the standard design of T_p , based on the worst case, can be advantageously replaced, once for all, by an on-line estimation of this parameter, leading a more flexible P&O algorithm. Unfortunately, the MPPT stepwise perturbation signal is not persistent enough to excite the PV system for the DKF-based identification [7]. Thus a Pseudo-Random Binary Sequence (PRBS) is superimposed to the dc/dc converter duty-cycle only for a suitable time interval needed by the DKF-based identification process. Hence, in the paper an appropriate convergence criterion is proposed for terminating the DKF process when the convergence of the parameters is reached. If compared with the approach proposed in [6], the method presented in this paper gives the estimation of the unknown parameters along with the confidence intervals. As a consequence, the time needed for achieving the system identification depends on the required accuracy. The knowledge of the estimation accuracy avoids

Manuscript received January 31, 2015; revised May 12, 2015; accepted for publication June 21, 2015.

Copyright ©2009 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

M.Ricco and E.Monmasson are with the SATIE, Université de Cergy-Pontoise, Cergy Pontoise, France (e-mail: ricco.mattia@u-cergy.fr, eric.monmasson@u-cergy.fr).

P.Manganiello is with the Seconda Università degli Studi di Napoli, D.I.I.I., Aversa (CE), Italy, and with the Università di Salerno, D.I.E.M., Fisciano (SA), Italy (e-mail: patriziomanganiello@gmail.com).

G.Petrone, G.Spagnuolo are with the Università di Salerno, D.I.E.M., Fisciano (SA), Italy (e-mail: gpetrone@unisa.it, gspagnuolo@unisa.it).

This work has been supported by the Université Franco Italienne (UFI/UIF) through Bando Vinci 2013 awarded to M. Ricco. It has been also partially supported by FARB 2013/2014/2015 funds the University of Salerno and by "Modelli, circuiti ed algoritmi per l'inseguimento del punto di massima potenza di impianti fotovoltaici in condizioni di mismatching" (L.R.5/2002-2007) from Campania Region (Italy).

any underestimation of the settling time and it can be used for additional diagnosis or monitoring purposes.

In order to achieve an efficient adaptive control, the identification must be performed in real-time, i.e. before a significant variation of the system operating conditions occurs. In addition, a faster identification method is less intrusive than a slower one, as it perturbs the system during a shorter time. Moreover, as the DKF is a recursive algorithm, its execution time must be lower than the sampling period, which is equal to the switching period. This last point is very challenging in this study since the power converter operates at high switching frequency.

In order to cope with these timing constraints, a hardware solution is adopted. Many studies have confirmed that the Field Programmable Gate Array (FPGA) technology is a good candidate when high speed performance is required [8]. Indeed, thanks to the exploitation of the intrinsic parallelism of the algorithm to be implemented, a significant reduction of the execution time is achieved. Unfortunately, to design a high performance hardware architecture for the complex identification algorithm described above is a challenging task, because the resources offered by a low cost target component, like the one used in this work, are limited. To the best of the authors' knowledge, this is the first time that the whole conversion chain, including the PV source and the switching converter, is modeled into a FPGA device for estimating its states and parameters through a DKF.

In literature, some authors propose FPGA implementation of model-based identification techniques. However, the requirements to be met and the adopted optimizations are different. For instance, [9] presents the implementation of a Recursive-Least-Square (RLS) algorithm aimed at identifying some parameters in order to develop a self-tuning regulator. The need of high computation precision and large dynamic range entails the use of a floating-point format. The authors in [10] propose the FPGA-based sensorless speed control. In this case the most important aspect is the needed resources, then a parallel reduced-order extended Kalman filter is proposed. On the other hand, in PV identification applications, both the time and area performance are of prime importance. Thus, specific attention has been paid to the choice of the PV model and of the parameters to be estimated. Moreover, in order to reduce the execution time, pipelined multipliers have been adopted in the developed architecture and the algorithm has been factorized in order to reduce the required resources.

The paper is organized as follows. In section II the Dual Kalman Filter is firstly introduced. Then, in the following section, it is applied to the PV system under study. It has to be mentioned that in the same section explanations are also given regarding the chosen dynamic model of the PV system that fits the identification issue to be addressed. Then, having given in section IV the characteristics of the experimental setup, the design of the DFK algorithm is presented in section V and validated by simulation in section VI. In section VII, the corresponding hardware architecture is given. Experimental tests for validating the proposed adaptive MPPT algorithm are given in section VIII and finally, conclusions are drawn.

II. DUAL KALMAN FILTER

A dynamic system can be expressed by its discrete-time State-Space (SS) model parametrized by the time-varying parameter vector $\theta(k)$, as shown in (2).

$$\begin{cases} x(k+1) = f(x(k), u(k), \theta(k)) + w(k) \\ y(k) = g(x(k), u(k), \theta(k)) + r(k) \end{cases} \quad (2)$$

where $x(k) \in \mathbf{R}^p$, $u(k) \in \mathbf{R}^n$ and $y(k) \in \mathbf{R}^m$ are the state vector, the input signals and the output signals, respectively. $f(\cdot)$ and $g(\cdot)$ are generally non-linear functions that govern the dynamic behaviour of the system. $w(k)$ and $r(k)$ are the process noise and measurement noise, respectively. They are assumed zero-mean white gaussian random processes, with respective covariance matrices W and R and reciprocally uncorrelated. $\theta(k)$ is the vector of the time-varying parameters, whose dynamic is approximated by the following state space model.

$$\begin{cases} \theta(k+1) = \theta(k) + v(k) \\ y(k) = g(x(k), u(k), \theta(k)) + e(k) \end{cases} \quad (3)$$

The first equation describes the dynamic of the parameters. As can be seen, the parameter dynamic is very slow in comparison with the state dynamic. $v(k)$ is the parameter noise that takes into account some driving processes that could lead parameter changes over the time. Its corresponding covariance matrix is V . Some additive estimation errors are also considered through the signal $e(k)$ of covariance matrix E . The main dual Kalman filter expressions are summarized in Table I.

Table I
MAIN DKF RELATIONS

| State Prediction | |
|--|---|
| $\hat{x}(k k-1) = f(\hat{x}(k-1 k-1), u(k-1), \hat{\theta}(k k-1))$ $P_{\hat{x}}(k k-1) = \hat{A}(k-1)P_{\hat{x}}(k-1 k-1)\hat{A}^T(k-1) + W$ | |
| Parameter Prediction | |
| $\hat{\theta}(k k-1) = \hat{\theta}(k-1 k-1)$ $P_{\hat{\theta}}(k k-1) = P_{\hat{\theta}}(k-1 k-1) + V$ | |
| State Update | |
| SJM | $\hat{A}(k) = \frac{\partial f(\hat{x}(k), u(k), \hat{\theta}(k))}{\partial \hat{x}(k)} \quad \hat{C}_x(k) = \frac{\partial g(\hat{x}(k), u(k), \hat{\theta}(k))}{\partial \hat{x}(k)}$ |
| KG | $L_x(k) = \frac{P_{\hat{x}}(k k-1)\hat{C}_x^T(k)}{C_x(k)P_{\hat{x}}(k k-1)\hat{C}_x^T(k) + R}$ |
| $\hat{x}(k k) = \hat{x}(k k-1) + L_x(k)[y(k) - \hat{y}(k)]$ $P_{\hat{x}}(k k) = (I - L_x(k)\hat{C}_x(k))P_{\hat{x}}(k k-1)$ | |
| Parameter Update | |
| PJM | $\hat{C}_{\theta}(k) = \frac{dg(\hat{x}(k), u(k), \hat{\theta}(k))}{d\hat{\theta}(k)}$ |
| KG | $L_{\theta}(k) = \frac{P_{\hat{\theta}}(k k-1)\hat{C}_{\theta}^T(k)}{C_{\theta}(k)P_{\hat{\theta}}(k k-1)\hat{C}_{\theta}^T(k) + E}$ |
| $\hat{\theta}(k k) = \hat{\theta}(k k-1) + L_{\theta}(k)[y(k) - \hat{y}(k)]$ $P_{\hat{\theta}}(k k) = (I - L_{\theta}(k)\hat{C}_{\theta}(k))P_{\hat{\theta}}(k k-1)$ | |

\hat{x} and $\hat{\theta}$ are the estimated state vector and the estimated parameter vector, respectively. $P_{\hat{x}}$ and $P_{\hat{\theta}}$ are the state error and parameter error covariance matrices, respectively. If they reach high values, then the estimations are not reliable. On the other hand, low values involve a good estimation. \hat{A} and \hat{C}_x are the State Jacobian Matrices (SJM) defined in Table I. \hat{C}_{θ} is the Parameter Jacobian Matrix (PJM), which calculation is described in [11]. L_x and L_{θ} are the state and parameter Kalman Gain (KG), respectively.

Once the DKF has been developed, its initialization and tuning have to be carried out. For this purpose, some guidelines are provided in [12]. Here a suitable initial choice of the state vector $\hat{x}(0|0)$ and of the parameter vector $\hat{\theta}(0|0)$ has to be adopted according to an a priori knowledge of the system. On the other hand, the tuning of the DKF is achieved by a trial-and-error approach.

The two specific aspects of the DKF application to the PV-MPPT problem treated in this manuscript are the adoption of a suitable perturbation signal and the definition of a proper convergence criterion. As for the first aspect, the perturbation signal on which the P&O technique is based is not sufficient to stimulate the system in a proper way, so that the DKF diverges. As a consequence, the DKF has to be launched in an asynchronous way: the MPPT perturbation process is stopped and a Pseudo-Random Binary Sequence (PRBS) is superimposed to the converter duty-cycle [13] in order to stimulate correctly the system. Thus, it is of interest to define a convergence criterion in order to assess that the identification procedure is terminated and the MPPT operation can be reactivated. The DKF reaches the convergence when the estimation error falls within a given boundary error, as shown in (4).

$$\begin{aligned} |\theta_n - \hat{\theta}_n| &\leq \theta_{n,err} \quad \forall n \\ \theta_{n,err} &= \theta_n \cdot err_{\theta_n} \end{aligned} \quad (4)$$

where θ_n and $\hat{\theta}_n$ are the real and the estimated n-th component of the parameter vector. $\theta_{n,err}$ defines an error threshold, which depends on the parameter value θ_n and on an imposed relative error err_{θ_n} . The error covariance matrix $P_{\hat{\theta}}$, given by the filter, provides the variance of the estimates and, then, the standard deviation σ can be derived. Therefore, the DKF reaches the convergence when σ falls below the imposed error:

$$\sigma_n < \theta_{n,err} \quad \forall n \quad (5)$$

In (5), $\theta_{n,err}$ depends on the real parameter value, which is unknown. However, if the consistency is verified [12], it is possible to replace the real value with the identified one. Then, the following criterion is obtained.

$$\begin{aligned} \sigma_n &< \hat{\theta}_{n,err} \quad \forall n \\ \hat{\theta}_{n,err} &= \hat{\theta}_n \cdot err_{\theta_n} \end{aligned} \quad (6)$$

Hence, the DKF reaches the convergence when the standard deviation is less than the chosen error bound. It is worth to note that the latter depends on the identified parameters and the relative error that is fixed by the designer. This introduces an additional degree of freedom in comparison with the identification method proposed in [6]. Indeed, a compromise

between the identification time and the identification accuracy can be reached by tuning the error bound.

III. PV SYSTEM MODEL

The best model structure of the PV system has to be chosen in order to avoid high algorithm complexity, larger identification time and divergence of the Kalman filter. The fast dynamic of the PV source makes the states of the system not observable in discrete-time with the commercial low-cost ADC, having a low sampling frequency. Hence, identifying only the PV source is not possible with the DKF. Therefore, the whole system has to be identified, where the dynamic is given by the dc/dc converter. The complete model is achieved by using the single-diode PV model [1]. However, this model includes too many parameters (photocurrent, series resistance, parallel resistance, converter inductance and capacitance with their parasitic resistance). Moreover, the PV parameters change too rapidly and too strongly causing mistakes in the identification. For these reasons, the small signal model around a given operating point of both the PV panels and the boost converter is considered. The DKF will then estimate the time-varying parameters of this dynamic model. Figure 1 shows the whole PV system and its small signal model used for the DKF identification process. The battery can be modelled by means of a dc generator in series with a series resistance and a number of series-connected parallel R-C groups. However, the latter can be neglected in the proposed analysis because the battery dynamic modelled by these R-C groups is significantly slower than the PV one. Thus only the series resistance should have to be taken into account. This parameter can be transferred to the input terminals of the boost dc/dc converter and joined to the parasitic resistance of the inductor. This approach has been considered and then the small signal battery model results in a short circuit.

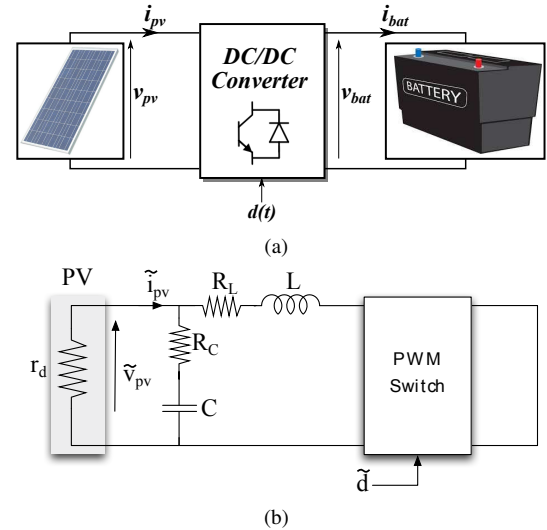


Figure 1. a) whole PV system; b) PV small signal model in the neighbourhood of an operating point.

In small-signal the PV modules can be seen as a differential resistance r_d that depends on the irradiance level and the temperature, as well as the operating point. In [6] the same model has been shown and the following transfer function between the panel voltage and the duty cycle has been derived:

$$G_{vp,d}(s) = \frac{\tilde{v}_{pv}}{\tilde{d}} = \mu \cdot \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (7)$$

where μ , ω_n and ζ are the DC gain, the natural frequency and the damping factor, respectively. The relation between these parameters and the physical ones (Figure 1) is

$$\begin{aligned} \omega_n &= \sqrt{\frac{R_L + r_d}{LC(R_c + r_d)}} \\ \zeta &= \frac{\omega_n}{2} \left[\frac{L}{R_L + r_d} + CR_c + C(R_L/r_d) \right] \\ \mu &= \frac{-V_{bat}r_d}{R_L + r_d} \end{aligned}$$

where V_{bat} is the nominal battery voltage. From (7), the Observable Canonical Form (OCF) SS model can be obtained:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -2\zeta\omega_n & 1 \\ -\omega_n^2 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ \mu\omega_n^2 \end{bmatrix} \tilde{d} \\ \tilde{v}_{pv} &= [1 \quad 0] x \end{aligned} \quad (8)$$

According to the time-domain expression of the step response of a second-order transfer function [1], the settling time can be expressed as

$$T_\varepsilon \simeq -\frac{1}{\zeta\omega_n} \cdot \ln\left(\frac{\varepsilon}{2}\right) \quad (9)$$

Then, the SS Model, parametrized by the T_ε , ω_n and μ , becomes:

$$\begin{aligned} \dot{x} &= f(x, u, \theta) = A_c x + B_c u = \begin{bmatrix} 2\frac{\ln(\frac{\varepsilon}{2})}{T_\varepsilon} & 1 \\ -\omega_n^2 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ \mu\omega_n^2 \end{bmatrix} u \\ y &= g(x, u, \theta) = C_c x + D_c u = [1 \quad 0] x \end{aligned} \quad (10)$$

where the matrices A_c , B_c , C_c and D_c are the continuous time matrices of the SS model. The input, the output, the states and the parameters of this model are:

$$\begin{aligned} x &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \tilde{v}_{pv}(t) \\ \int (\mu\omega_n^2 \tilde{d}(t) - \omega_n^2 \tilde{v}_{pv}(t)) dt \end{bmatrix} \\ u &= \tilde{d}(t) \quad y = \tilde{v}_{pv}(t) \quad \theta = \begin{bmatrix} T_\varepsilon \\ \omega_n \\ \mu \end{bmatrix} \end{aligned}$$

where \tilde{d} and \tilde{v}_{pv} are the perturbation superimposed to the nominal duty-cycle and the corresponding panel voltage variation, respectively. The continuous time model achieved in (10) has to be discretized in order to apply the DKF. A good accuracy and a low complexity of the discrete model are necessary in order to achieve good identification results without increasing the execution time. Excellent accuracy is obtained by using implicit discretization methods. However, a complex discrete model is achieved causing a significant increase of the execution time. For this reason, the attention has been focused on the explicit approach, in particular on the explicit midpoint method.

Thus, the states and the output prediction in the DKF becomes:

$$\begin{aligned} \hat{x}(k|k-1) &= f(\hat{x}(k-1|k-1), u(k-1), \hat{\theta}(k|k-1)) \\ &= \hat{A}(k) \hat{x}(k-1|k-1) + \hat{B}(k) u(k-1) \end{aligned} \quad (11)$$

$$\hat{y}(k) = \hat{C}(k) \hat{x}(k|k-1) + \hat{D}(k) u(k-1)$$

where:

$$\begin{aligned} \hat{A} &= I + T_s \cdot \hat{A}_c + \frac{T_s^2}{2} \cdot \hat{A}_c^2 \\ \hat{B} &= T_s \cdot \hat{B}_c + \frac{T_s^2}{2} \cdot \hat{A}_c \hat{B}_c \\ \hat{C} &= \hat{C}_c \\ \hat{D} &= \hat{D}_c \end{aligned}$$

IV. SYSTEM SPECIFICATIONS

The synoptic of the whole PV system, including the FPGA control unit where the DKF-based adaptive MPPT is implemented, is shown in Figure 2. It consists of a Kyocera KC120-1 PV module characterized by an open circuit voltage and a short circuit current equal to 21.5 V and 7.45 A, respectively. It is connected to a boost dc/dc power converter, which nominal parameters are summarized in Table II. The output of the converter is linked to a 36 V battery pack. The PV current and PV voltage are acquired by means of two Analog to Digital Converters (ADCs) with resolution equal to 40.4 mV and 20 mA, respectively. The sampling frequency is fixed equal to the switching frequency in order to filter the ripple component affecting the measurement signals. The whole PV system is controlled through a low-cost Spartan-6 FPGA hosting both the MPPT algorithm and the DKF. Real-time transfer of data between the FPGA board and a host computer is ensured by a USB-JTAG interface.

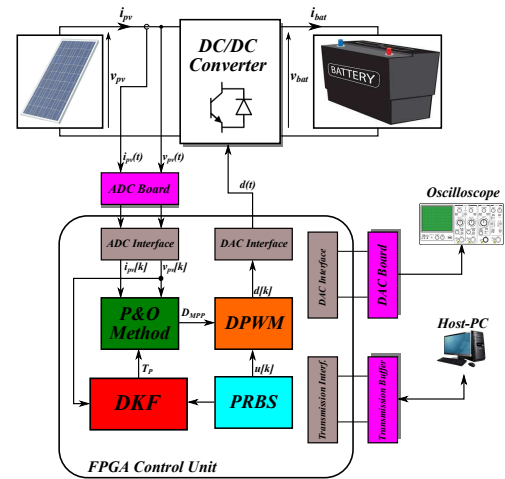


Figure 2. Synoptic of the adaptive MPPT algorithm based on a DKF and intended for PV applications

The PRBS signal is chosen according to [6]. It has a length M equal to 1023, an amplitude of 0.03125 and an injection frequency equal to the converter switching frequency (195 kHz). Finally, the P&O algorithm and the DPWM module are designed according to [14].

Table II
COMPONENTS NOMINAL VALUES

| Parameter | Absolute Value |
|-------------------------------------|----------------|
| Input inductance L | 115 μH |
| Equivalent loss resistance R_L | 100 m Ω |
| Input Capacitance C | 50 μF |
| Capacitor ESR R_{in} | 10 m Ω |
| Converter switching frequency f_s | 195 kHz |

V. ALGORITHM DEVELOPMENT

The DKF is now validated. In order to prepare its digital implementation, the whole adaptive MPPT algorithm is divided into modules easier to implement. Then, the normalization and the quantization of the algorithm are performed. Some algorithm optimizations are carried out and the procedure for calculating the settling time is presented. Finally, the flowchart of the whole adaptive MPPT algorithm is shown.

A. Modular Partitioning

The modular partitioning is a fundamental step to reduce the time-to-market and to achieve a more flexible and hierarchical architecture. The designer has to be able to divide the whole adaptive MPPT algorithm into reusable, independent and simpler sub-algorithms, called modules. Figure 3 shows the five hierarchy levels determined for the developed adaptive MPPT algorithm based on DKF.

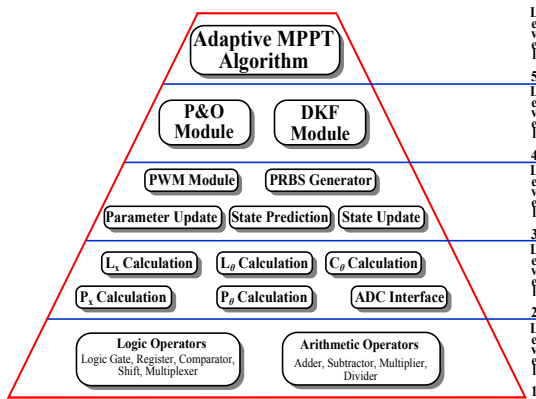


Figure 3. Modular Partitioning of the Adaptive MPPT Controller Based on DKF

The lowest level is composed of both arithmetic and logic operators, which can be considered as fine-grain operators [8]. The second level contains the indispensable module for implementing the DKF (KG calculation, SJM and PJM calculation, covariance matrix calculation), as well as the interface module (ADC interface). At the third level of hierarchy the modules dedicated to the control of the converter (PWM module), the modules concerning the DKF algorithm (state prediction, state update and parameter update modules) and the PRBS generator can be found. The fourth level consists of coarse-grain modules, i.e. the P&O module and the DKF module. Finally, the whole adaptive MPPT controller is placed into the highest level.

B. Algorithm Digital Realization

The aim of this step is to prepare the developed identification algorithm for the implementation in the FPGA device. It consists of two main steps: the normalization and the quantization.

1) **Algorithm Normalization:** the aim of the normalization is to facilitate the use of the fixed-point representation for encoding the variables of the algorithm. To this purpose, each variable φ is divided by its base-value φ_B , obtaining the corresponding per-unit counterpart φ_n , as shown in (12).

$$\varphi_n = \frac{\varphi}{\varphi_B} \quad (12)$$

The base-values depend on the nominal value of the variables and the ADC resolution. The defined main base-values are: $v_{pv,B}$ for the panel voltage, e_B for the PRBS sequence, $x_{k,B}$ for the estimated states, $\theta_{k,B}$ for the estimated parameters and the internal variables of the DKF. Their values are given in Table III.

Table III
BASE-VALUES OF THE MAIN ALGORITHM VARIABLES.

| Panel Voltage | PRBS sequence |
|--|-----------------|
| $v_{pv,B} = 21.5V$ | $e_B = 0.03125$ |
| DKF States | |
| $x_{1,B} = 2.5 V$; $x_{2,B} = 35000 rad^2V/s$ | |
| DKF Parameters | |
| $f_{e,B} = 5000 Hz$; $w_{n,B} = 20000 rad/s$; $\mu_B = 50 V$ | |

2) **Algorithm Quantization:** during this step the fixed point data format is chosen. The identification accuracy, the filter convergence and the identification time are strongly related to this choice. Normally, the fixed-point representation can be expressed using the label $s[wQf]$ for signed data and $u[wQf]$ for unsigned data, where w and f are the number of bits of the word and the number of bits of the fractional part, respectively. After having made several simulations the chosen data formats are: $s[42Q40]$ for the DKF, $s[12Q11]$ for the MPPT algorithm and $u[10Q9]$ for the DPWM.

C. Algorithm Optimization

Table IV gives a first evaluation of the DKF complexity developed till now. In order to reduce the algorithm complexity, the designer can operate either at the model selection stage or during the algorithm realization step.

Table IV
NUMBER OF OPERATIONS OF THE PROPOSED DKF ALGORITHM

| Kalman Filter Modules | + | - | * | ÷ |
|-----------------------|-----------|-----------|------------|-----------|
| STATE PREDICTION | 10 | 0 | 20 | 4 |
| STATE INNOVATION | 25 | 5 | 90 | 6 |
| PARAMETER INNOVATION | 56 | 29 | 215 | 10 |
| TOTAL | 91 | 34 | 325 | 20 |

On one hand, an accurate choice of the parameters to be identified has to be made in order to obtain the required information by using a less complex system model. For the

application under study, a complexity reduction can be reached estimating instead of the settling time T_e its inverse f_e that will be called "settling frequency". This allows avoiding divisions in the SS model (10). Its base-value is given in Table III.

On the other hand, a good choice of the base-values allows an important reduction of the number of multiplications. In the DKF, it is possible to choose the base-values of the internal variable as power of two. Thus, when a product due to the normalization has to be carried out, this becomes a simple shift operation.

Applying these optimizations to the DKF algorithm the complexity is significantly reduced, as shown in Table V.

Table V
POST-EVALUATION OF THE KF ALGORITHM COMPLEXITY

| Kalman Filter Modules | + | - | * | ÷ |
|-----------------------|-----------|-----------|------------|----------|
| STATE PREDICTION | 10 | 0 | 24 | 0 |
| STATE INNOVATION | 25 | 5 | 51 | 2 |
| PARAMETER INNOVATION | 56 | 29 | 137 | 3 |
| TOTAL | 91 | 34 | 212 | 5 |

With the optimized DKF the f_e is estimated. Thus, in order to evaluate an overestimated settling time, the equation in (13) is adopted.

$$T_e = \frac{1}{\hat{f}_e - \sigma_{f_e}} \quad (13)$$

where σ_{f_e} is the settling frequency standard deviation. In order to reduce the computational burden, the σ_{f_e} has been replaced by an overestimation of it given by the imposed error in the convergence criterion (5). This guarantees an additional security margin to the calculation of the MPPT period in comparison to the adaptive controller proposed in [6]. Besides, it is worth mentioning that the computation of the ratio given in (13) does not require an additional divisor since this computation, taking place after the DKF estimation, can use one of the divisor of this IP.

Figure 4 shows the flowchart of the whole adaptive MPPT algorithm implemented in the FPGA. At the beginning, the MPPT is running with a non-optimized perturbation period. When the identification procedure is activated and as soon as the steady-state is reached, the MPPT algorithm is stopped and the recursive DKF identification technique is launched. Within the interval between two successive PRBS sample injections, different steps have to be executed. Firstly, the evaluation of the prediction of both states and parameters is carried out. After that, the updating step is made by using the difference between the actual panel voltage and the predicted panel voltage. Finally, the DKF convergence is evaluated according to the previously proposed convergence criterion. If the convergence is reached, the identification procedure, as well as the PRBS injection, are stopped and the MPPT is reactivated with a new optimized T_p equal to the new estimation of T_e . When the identification is reactivated the initial states and parameters are fixed equal to those obtained during the last estimation.

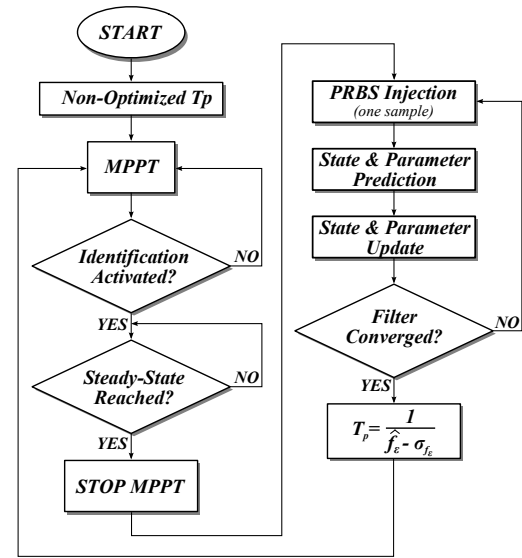


Figure 4. Flowchart of the whole adaptive MPPT algorithm implemented in the FPGA.

VI. ALGORITHM VALIDATION

The model in Figure 1 has been simulated in Matlab/Simulink. Due to the recursive formulation of the DKF, the initialization of the filter is necessary. In (14), a reasonable choice of the initial conditions of the filter is shown.

$$\begin{aligned} x_0 &= \begin{bmatrix} 0 & 0 \end{bmatrix}^T \\ P_{x,0} &= \begin{bmatrix} 1 & 1 \\ 0.01 & 10000 \end{bmatrix} \\ \theta_0 &= \begin{bmatrix} 500 & 10000 & -10 \end{bmatrix}^T \\ P_{\theta,0} &= \begin{bmatrix} 24 \cdot 10^4 & 0 & 0 \\ 0 & 25 \cdot 10^6 & 0 \\ 0 & 0 & 9 \cdot 10^2 \end{bmatrix} \end{aligned} \quad (14)$$

Moreover, a trial-and-error approach has been used for tuning the filter [12]. The parameters in (15) have been adopted during the simulation and experimental tests.

$$\begin{aligned} W &= \begin{bmatrix} 3.145 \cdot 10^{-7} & 3.125 \cdot 10^{-8} \\ 3.125 \cdot 10^{-8} & 3.125 \cdot 10^{-11} \end{bmatrix} \\ R &= 0.00025 \\ V &= \begin{bmatrix} 6.25 \cdot 10^{-7} & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 3.125 \cdot 10^{-3} \end{bmatrix} \\ E &= 0.8 \end{aligned} \quad (15)$$

Figure 5 shows the identified parameters obtained with the nominal boost parameters presented in Table II and with a differential resistance r_d equal to 5 Ω .

The green curves in Figure 5 define the confidence interval of the estimated parameter, which is equal to the estimated parameter \pm the corresponding standard deviation σ . The latter corresponds to the square root of the variance given by the error covariance matrix $P_{\hat{\theta}}$. The dotted red curves correspond to the reference values of the small-signal model. As it can be seen in Figure 5, the estimated values of the PV parameters

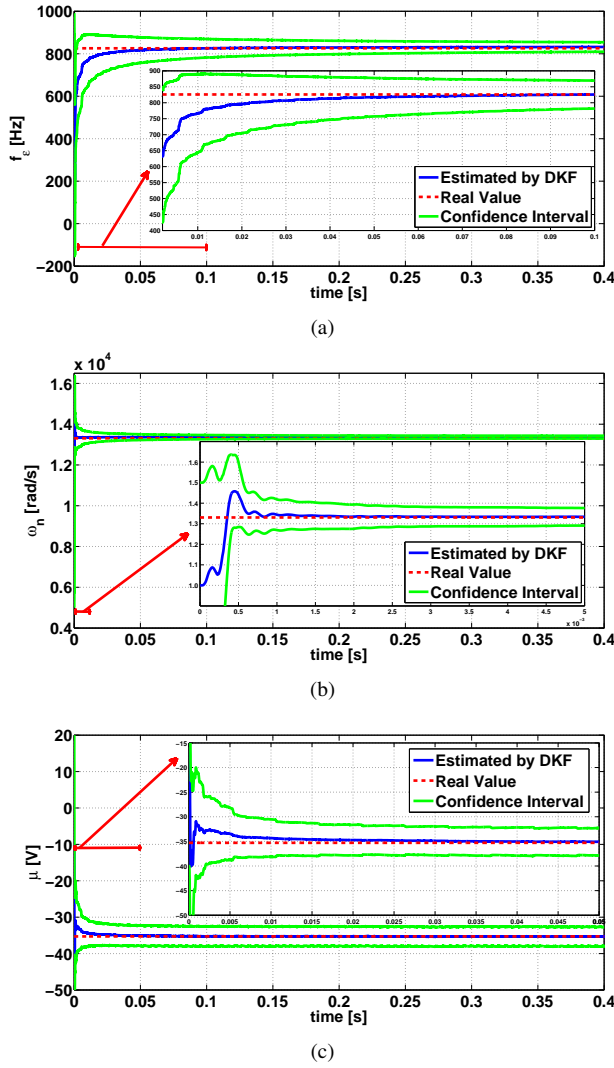


Figure 5. Estimated parameters by the DKF by using Matlab/Simulink tool with the nominal boost parameters presented in Table II and with a differential resistance equal to 5Ω : a) Estimated settling frequency b) Estimated natural frequency c) Estimated DC gain.

tend to the actual ones.

Some simulations have been performed by changing the parameters in the following ranges: $C = [50, 80] \mu F$, $R_C = [10, 20] m\Omega$, $L = [80, 115] \mu H$, $R_L = [100, 200] m\Omega$, with $r_d = 5 \Omega$. The results reveal that the method in [6] gives an error in estimating the settling time that is in the range $[-12.7, +3.4]\%$, where its sign is not predictable. Thus an underestimation as well as an overestimation of the settling time can be obtained. Conversely, the DKF method proposed in the manuscript gives an error in the range $[+0.9, +2.9]\%$, so that an overestimation of the settling time is always given. This ensures that the DKF method prevents a poor behaviour of the MPPT technique.

VII. ARCHITECTURE DEVELOPMENT

In order to better cope with the different application constraints (modularity constraint, area constraint and time constraint), the developed FPGA-based architecture has been hand-coded. For such complex algorithms, the fulfilment of

all these constraints is a difficult challenge for designers. The main steps to design an FPGA-based architecture are presented below.

A. Performance Evaluation

Firstly, an evaluation of the time/area performance of the developed identification algorithm preserving the whole parallelism is carried out. The aim is to verify if the corresponding FPGA architecture satisfies all the constraints. If this is the case, it is possible to go directly to the architecture design. Otherwise, if the fully parallel architecture does not meet the previous constraints, some architecture optimizations have to be made, like in the studied case.

B. Architecture Optimization

In order to cope with the time constraint, a fully pipelined architecture has been developed in order to increase the maximum possible clock frequency to drive the architecture. It consists in placing registers between operators in order to cut the maximum delay path and then to decrease the propagation delay. However, the corresponding architecture exhibits a low maximum clock frequency ($53 MHz$), limited by the multipliers, and requires a large amount of hardware resources. In order to deal with these issues, pipelined multipliers have been adopted and the Algorithm Architecture Adequation (A^3) methodology has been used [8]. In the following, both optimizations are discussed.

1) **Pipelined Multiplier:** A pipelined multiplier with area optimization strategy can be used in order to increase the clock frequency and decrease the number of used DSP units. However, the pipelining leads to increase the latency (number of clock cycles). Therefore, a compromise between the maximum clock frequency and the latency has to be made. Several multipliers with different pipeline levels have been tested.

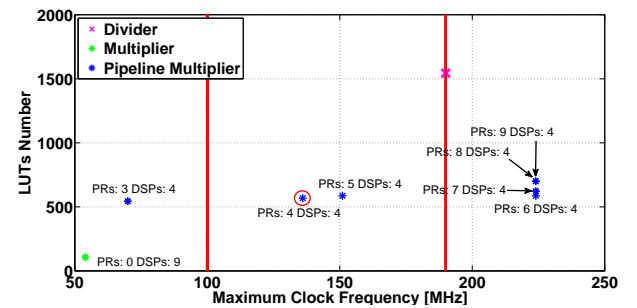


Figure 6. Comparison between multipliers with different pipeline levels and the divider in terms of maximum clock frequency, used DSP units and LUT number.

Figure 6 shows the comparison between different multipliers with different Pipeline Register (PR) number. Their maximum clock frequency, Look Up Table (LUT) number and DSP number are compared. The maximum clock frequency equal to $190 MHz$ is imposed by the divider. Therefore, supposing to work at least with a clock frequency equal to $100 MHz$, the multiplier with 4 pipeline registers is chosen. With this choice, a significant reduction of the required DSP blocks and an increase of the clock frequency are achieved.

2) **Algorithm Architecture Adequation (A^3):** The aim of this step is to deal with the area mismatching between the available FPGA resources and the ones required by the DKF algorithm. Being the heaviest operators in the proposed algorithm, a factorization is applied to the dividers and the multipliers. The A^3 algorithm consists in three steps:

- **Data Flow Graph (DFG) Design:** the algorithm is represented through its corresponding graph.
- **Data Dependency Evaluation:** the data dependency is evaluated and the best factorization level is determined taking into account the limitation of the FPGA internal resources.
- **Factorized DFG Design:** the factorized algorithm is represented through its corresponding factorized graph. In order to delimit the factorized border some nodes have to be introduced: Fork (F), Join (J) and Iterate (I) [8].

The factorization level is determined as described below. In a XC6SLX45 Xilinx Spartan-6 FPGA 58 DSPs Slices are available. Of the 5 dividers necessary to this design, 3 are used in the state Kalman gain calculation and 2 in the parameter Kalman gain calculation. Using only one divider for the both Kalman filters will be too confusing. So, in order to preserve the architecture modularity, two dividers have to be used in the architecture, one for each KF. As a consequence only 26 DSPs remain to implement all the requested multipliers. As each multiplier is requiring 4 DSP units, at most 6 multipliers can be implemented. According to the modularity constraint, they are allocated in the following way: one in the State Prediction module, two in the State Innovation module and three in the Parameter Innovation module.

C. Architecture Design

It consists in developing the hardware architecture of each modules. This architecture is constituted by a data path and a control unit. The final data flow graph is obtained after having factorized the initial data flow graph that included all the potential parallelism of the targeted algorithm. Each factorization node (F, J and I) is then replaced by its corresponding operator. A multiplexer is used for implementing a fork and registers with the adequate enable control signals in place of the join and the iterative nodes. The control unit has to ensure the proper synchronization between all the tasks of the process.

The state prediction architecture is shown in Figure 7, and the whole DKF architecture is presented in Figure 8.

D. Time-Area Analysis

After synthesis, the execution time is determined by evaluating the latency number of each module. Table VI shows the latency and the corresponding execution time of the ADC block and the internal DKF modules.

$T_{clk,ADC}$ is the clock of the ADC blocks, being equal to 12,5 MHz. It is worth to note that the total execution time is the sum of the ADC time, the State Prediction time and the biggest time between the State Innovation and the Parameter Innovation, since the latter two are executed in parallel. The total DKF execution time is less than the used sampling period (5 μs), thus the timing constraint is fulfilled.

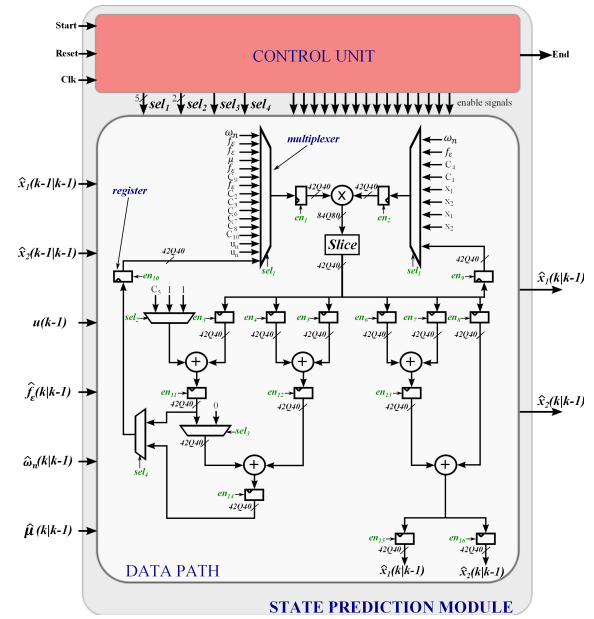


Figure 7. Hardware FPGA Architecture : State Prediction Module.

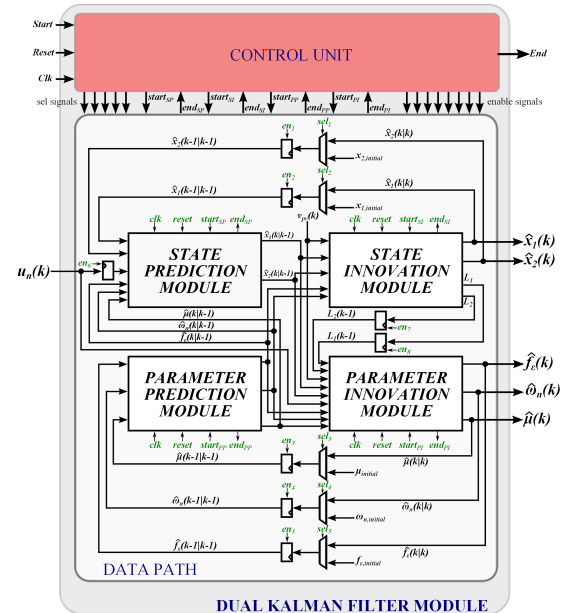


Figure 8. Hardware FPGA Architecture : Dual Kalman Filter Module.

Table VI
EXECUTION TIME FOR THE DUAL KALMAN FILTER.

| | Latency Number | Execution time |
|------------------------------------|------------------|----------------|
| ADC Conversion t_{ADC} | $16 T_{clk,ADC}$ | 1,28 μs |
| State Prediction t_{SP} | $N_{SP} = 29$ | 290 ns |
| State Innovation t_{SI} | $N_{SI} = 123$ | 1,23 μs |
| Parameter Innovation t_{PI} | $N_{PI} = 192$ | 1,92 μs |
| $T_{tot} = t_{ADC} + t_{SP} + t_I$ | | 3,49 μs |

On the other hand, the area analysis is carried out by the synthesizer tool. Table VII shows the needed resources to implement the proposed algorithm as well as the interface hardware blocks. As can be seen, the area constraint is fulfilled.

Table VII
CONSUMED HARDWARE RESOURCES OF THE ADAPTIVE MPPT
CONTROLLER

| Hardware Resource | Available | Consumed | Percentage |
|---------------------|-----------|----------|------------|
| Slice Registers | 54576 | 20770 | 38 % |
| Slice LUTs | 27288 | 15733 | 57 % |
| Occupied Slices | 6822 | 6094 | 89 % |
| 16-KByte RAM Blocks | 116 | 99 | 85 % |
| DSP48A1s Slices | 58 | 51 | 87 % |

Hence, the implemented hardware architecture fulfils both the area and the timing constraints, while preserving as well the modular constraint.

VIII. EXPERIMENTAL TESTS

In this section the experimental validation of the proposed adaptive MPPT algorithm based on the DKF is presented. The experimental test bench has been described in Section IV. The ChipScope tool is adopted for real-time transfer of data in order to acquire the internal signals of the FPGA. The whole algorithm is implemented in a Xilinx XC6SLX45 Spartan-6 FPGA device, with a 100 MHz clock frequency.

Figure 9 shows the identified settling frequency and the corresponding confidence interval. According to the proposed convergence criterion, two identification times have been pointed out in Figure 9: the first one corresponds to the time achieved by using an error bound equal to 15 %, whereas, the second one by using an error bound equal to 12,5 %. In these two cases the DKF is converging in less than 15 ms. Once the convergence is reached, the value of the identified settling frequency and of the corresponding standard deviation are used for evaluating the settling time as explained in section V-C.

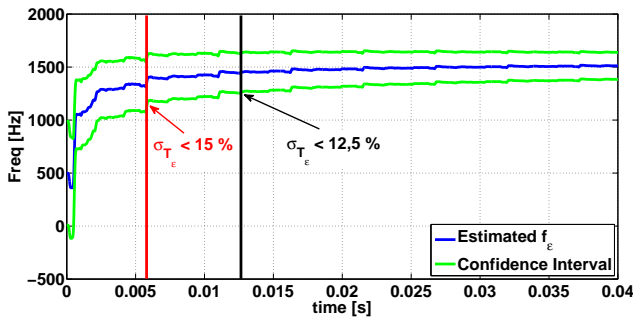
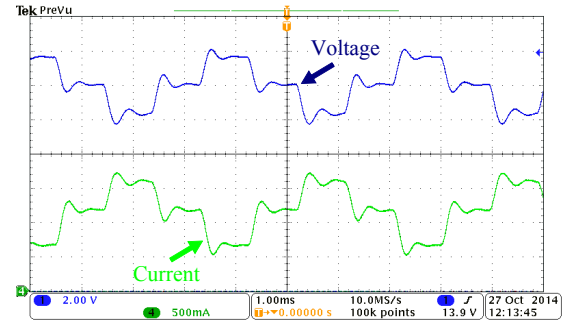


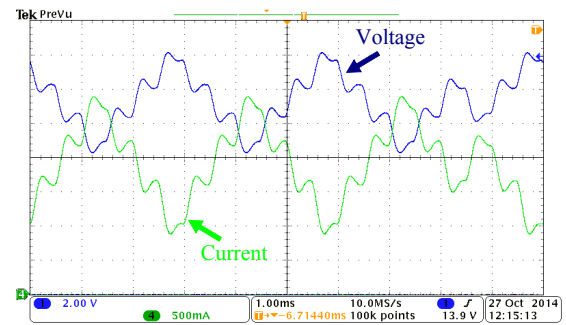
Figure 9. Settling Frequency estimation with the corresponding confidence interval.

The perturbation amplitude of the P&O MPPT has been selected according to the guidelines shown in [1], so that its value is equal to 0.03125. Figure 10(a) shows the behaviour of the PV voltage and current when the perturbation period T_p of the MPPT is equal to the settling time estimated by the DKF algorithm adopting an error bound equal to 15 %. In this condition, the system exhibits a stable steady-state behaviour. This means that a proper value of T_p has been used. On the contrary, a different T_p value can deteriorate the MPPT

performance. Indeed, as shown in Figure 10(b), choosing a value of T_p smaller than the estimated one makes the PV voltage and the PV current oscillating with an unpredictable behaviour, deteriorating the MPPT efficiency. Conversely, a bigger value slows down the MPPT controller as already mentioned in the Introduction.



(a)



(b)

Figure 10. MPPT behaviour by using $C = 50\mu\text{F}$: (a) $T_p = T_{p,min} = 940\mu\text{s}$, (b) $T_p = 500\mu\text{s}$.

These tests confirm the effectiveness of the proposed adaptive MPPT method. Besides, the estimated parameters can also be used for diagnosis or monitoring purposes.

IX. CONCLUSIONS

An FPGA-based adaptive MPPT controller for PV applications based on DKF has been presented in this paper. The developed adaptive controller allows optimizing the perturbation period of the P&O algorithm during the normal operation of the system. In comparison to the non parametric identification technique presented in [6] the DKF supplies not only the parameter values but also their confidence intervals. These quantities allow avoiding any underestimation of the settling time. Moreover, this additional information can be advantageously used for monitoring and diagnosis purposes.

In order to cope with the different and conflicting requirements, a proper design methodology has been adopted and several optimizations, both on the algorithm and on the architecture, have been made. Both the timing requirement and the area constraint have been fulfilled and the experimental results confirm the interest of the implemented adaptive MPPT controller.

REFERENCES

- [1] N. Femia, G. Petrone, G. Spagnuolo, and M. Vitelli, "Power Electronics and Control Techniques for Maximum Energy Harvesting in Photovoltaic Systems". CRC Press, 2013.
- [2] L. Piegari and R. Rizzo, "Adaptive perturb and observe algorithm for photovoltaic maximum power point tracking," *Renewable Power Generation, IET*, vol. 4, no. 4, pp. 317–328, 2010.
- [3] F. Zhang, K. Thanapalan, A. Procter, S. Carr, and J. Maddy, "Adaptive hybrid maximum power point tracking method for a photovoltaic system," *IEEE Transactions on Energy Conversion*, vol. 28, no. 2, pp. 353–360, 2013.
- [4] Y. Xia, K. Ahmed, and B. Williams, "Wind turbine power coefficient analysis of a new maximum power point tracking technique," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 3, pp. 1122–1132, 2013.
- [5] L. Belhadji, S. Bacha, I. Munteanu, A. Rumeau, and D. Roye, "Adaptive MPPT applied to variable-speed microhydropower plant," *IEEE Transactions on Energy Conversion*, vol. 28, no. 1, pp. 34–43, 2013.
- [6] P. Manganiello, M. Ricco, G. Petrone, E. Monmasson, and G. Spagnuolo, "Optimization of perturbative PV MPPT methods through online system identification," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6812–6821, Dec 2014.
- [7] T. Soderstrom and P. Stoica, "System Identification", 2011th ed., P. Hall, Ed., 1989.
- [8] E. Monmasson, L. Idkhajine, M. N. Cirstea, I. Bahri, A. Tisan, and M. W. Naouar, "FPGAs in industrial control applications," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 2, pp. 224–243, 2011.
- [9] Z. Salicic, J. Cao, and S. K. Nguang, "A floating-point FPGA-based self-tuning regulator," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 2, pp. 693–704, April 2006.
- [10] N. K. Quang, N. T. Hieu, and Q. Ha, "FPGA-based sensorless PMSM speed control using reduced-order extended Kalman filters," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6574–6582, Dec 2014.
- [11] G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation," *Journal of Power Sources*, vol. 134, no. 2, pp. 277 – 292, 2004.
- [12] H. Durrant-Whyte, "Introduction to estimation and the Kalman filter," 2006.
- [13] K. Godfrey, "Perturbation signals for system identification", Prentice-Hall, Ed., 1993.
- [14] M. Ricco, P. Manganiello, G. Petrone, E. Monmasson, and G. Spagnuolo, "FPGA-based implementation of an adaptive perturb & observe MPPT controller for PV applications," in *Industrial Electronics (ISIE), 2014 IEEE 23rd International Symposium on*, June 2014, pp. 1876–1881.



Eric Monmasson (M'96 - SM'06) received the Ing. and Ph.D. degrees from the Ecole Nationale Supérieure d'Ingénieurs d'Electrotechnique d'Electronique d'Informatique et d'Hydraulique de Toulouse (ENSEEHT), Toulouse, France, in 1989 and 1993, respectively. He is currently a full professor at the University of Cergy-Pontoise, France. He is also with the SATIE laboratory. He is an associate editor of IEEE Transactions on Industrial Electronics and IEEE Transactions on Industrial Informatics.



Giovanni Petrone received the M.S. degree in Electronic Engineering from the University of Salerno, Italy, in 2001 and the Ph.D. in Electrical Engineering from the University of Napoli "Federico II", Italy, in 2004. Since October 2014 he is Associate Professor of Electrical Engineering at the University of Salerno. His main research interests are in the analysis and design of switching converters for telecommunication applications, renewable energy sources in distributed power systems, tolerance analysis of electronic circuits.



Mattia Ricco received the Master Degree, cum laude, in Electronic Engineering from the University of Salerno, Italy, in 2011 and the Ph.D. in Electrical and Electronic Engineering from the University of Cergy-Pontoise, France, and in Information Engineering from the University of Salerno, Italy, in 2015. His main research interests are the design and the implementation, in FPGA devices, of digital controllers and identification algorithms for power electronics and Photovoltaic systems.



Giovanni Spagnuolo (M'98 - SM'10) received the Master Degree in Electronic Engineering from the University of Salerno, Italy, in 1993 and the Ph.D. in Electrical Engineering from the University of Napoli "Federico II", Italy, in 1997. Since January 2004 he is Associate Professor of Electrical Engineering at the University of Salerno. He is Editor of the IEEE Journal of Photovoltaics. He is an Associate Editor of the IEEE Transactions on Industrial Electronics.



Patrizio Manganiello received the Master Degree, cum laude, in Electronic Engineering from the University of Salerno, Italy, in 2011 and the Ph.D. in Energy Conversion from the Second University of Naples, Italy, in 2014. His main research interests are the analysis, design and control of switching converters for renewable energy source as well as the on-line diagnostics of the whole energy conversion system by means of linear and non-linear identification techniques.