THE DESIGN AND IMPLEMNTATION OF A PID CONTROLLER IN AN FPGA CIRCUIT

Cristian ORIAN, Dorin PETREUŞ
Technical University of Cluj-Napoca, Cluj-Napoca, Romania
26-28 G. Baritiu Street, 400027 Phone +40264202348, Email: cristian.orian@ael.utcluj.ro

Abstract: The following paper presents the design and implementation of a PID regulator which is used to control a step-down converter. The transfer functions of both the control block and the buck converter were modeled in Mathcad. These models were instrumental during the design process and results validation, because they were used to plot a step response. The overall feedback loop was simulated in Psim and resulted in the same step response. An FPGA was used for the practical implementation. The circuit was described, synthesized and implemented in Xilinx Vivado. Relevant parts of the control circuit and measurements of the step response are presented.

Keywords: *PID control, FPGA, digital control, buck converter.*

I. INTRODUCTION

The design of power converters is very diverse due to their large scale utilization ranging from home appliances to power management systems and interfaces for distributed energy resources (DER) in microgrids. Most of the front end converters, which are designed to be supplied from renewable sources (e.g. batteries, photovoltaic panels), require the use of more sophisticated control algorithms, which maximize the power extraction and feature DER protection capabilities [1]. Load-connected converters must withstand disturbances (varying input voltage according to power extraction maximization, varying load) in order to meet their design specifications.

Although most converter controllers used to be analog, the advent of digital integrated circuits has enabled the controllers' transition to numerical form, which has created the premises for interesting new features. Among them, there is the possibility to design controllers whose parameters are immune to the influence of external factors (resistances and capacitances varying with temperature, voltages), increased signal processing abilities (which allows for the design of observers), communication capabilities or the possibility to design a control strategy which can be updated [2]. The drawback of the digital controllers is their reduced speed, which mostly relies on the computational capabilities of the hardware employed [3].

The control of a power converter via a PID regulator is a well established method [4]. It is widespread and it offers various benefits: it is well documented, it can be expected to produce regulation within a given range of disturbance values and its Bode plot can provide a more intuitive understanding of its intrinsic functionality. In fact, each of the three controller components has a physical significance [5]. The downside of this method is the fact that it works as part of a linear system. The power converter operates in switched mode, hence it is not linear, although it can be approximated as such provided that the open loop crossover frequency is not too close to the switching frequency (typically their ratio is about ¼) [6].

One way to determine the PID coefficients of a controller is to employ the Ziegler-Nichols (ZN) tuning method [7]. The method works well for low order systems, like a step-down converter. However, its performance is known to degrade when implemented in a high order system, increasing the settling times, causing high overshoots and even oscillations [8]. Moreover, nonlinear systems are also difficult to control using this method [9, 10]. In order to overcome these problems, other methods that return the PID coefficients have been proposed such as refined ZN, pole placement, fuzzy control and optimal control [7, 11-15].

The control algorithm was designed and implemented on an FPGA, which features high computational speeds, parallel processing capabilities, high reliability and flexibility [16 – 19]. Additionally, it integrates an ADC block. All of the aforementioned features have increased the interest for FPGAs in various applications such as plant control, embedded systems, motor drive or signal processing.

The practical controller was implemented using a Nexys 4 development board equipped with an Artix 7. The control circuit was described in Vivado and the floating point IP was used to implement all the required arithmetical operations. The clock frequency is 100MHz and the delays introduced by the arithmetic operators are one clock cycle each. The ADC samples data at 1MSPS, at a resolution of 12 bits and a reference voltage of 1V.

The paper is organized as follows: Section II provides an overview of the control algorithm, the Mathcad model and Psim simulations and Section III presents the experimental results. The conclusions are presented at the end.

II. MODELING AND SIMULATION

The PID control is widely used in dc-dc power converter applications because it is well documented and it produces

Manuscript received April 24, 2015; revised June 3, 2015

acceptable results in relation to a large variety of transient response specifications. The transfer function was augmented by inserting a first order low pass filter in series with the derivative component. This limits the overshoot of the regulator response, the amplification of noise at the input of the regulator and introduces a pole in the overall open loop transfer function which helps to bring its magnitude below 0dB. The transfer function is expressed in (1).

$$\frac{U(s)}{\varepsilon(s)} = K_p \cdot \left(1 + \frac{1}{s \cdot T_i} + \frac{s \cdot T_d}{1 + s \cdot \frac{T_d}{N}}\right) \tag{1}$$

The Bode plot of the selected PID algorithm is presented in figure 1. It exhibits a pole at the origin, whose purpose is to bring the steady state error to 0, two zeros and a second pole. The transfer function of the power stage of the linear model for a step down converter is known to exhibit 2 poles and one zero. The intuitive approach is to overlap the regulator zeros with the power stage poles and allow for an acceptable phase margin.

Since the controller is implemented in digital form, the transfer function was converted from the s to the z domain. In order to achieve that, the bilinear transformation was employed. The resulting z domain transfer function and corresponding coefficients are given in (2) and (3), where T denotes the sampling time of the error signal.

$$H_R(z) = \frac{a_0 + a_1 \cdot z^{-1} + (a_{21} + a_{22}) \cdot z^{-2}}{1 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}} \tag{2}$$

$$\begin{cases} a_0 = K_p + \dfrac{K_p \cdot T}{2 \cdot T_i} + \dfrac{2 \cdot N \cdot K_p \cdot T_d}{T \cdot N + 2 \cdot T_d} \\ a_1 = \dfrac{2 \cdot T^2 \cdot N \cdot K_p - 8 \cdot K_p \cdot T_i \cdot T_d - 8 \cdot N \cdot K_p \cdot T_i \cdot T_d}{2 \cdot T_i \cdot (T \cdot N + 2 \cdot T_d)} \\ a_{21} = \dfrac{4 \cdot K_p \cdot T_i \cdot T_d - 2 \cdot N \cdot K_p \cdot T_i \cdot T + T^2 \cdot N \cdot K_p}{2 \cdot T_i \cdot (T \cdot N + 2 \cdot T_d)} \\ a_{22} = \dfrac{4 \cdot N \cdot K_p \cdot T_i \cdot T_d - 2 \cdot K_p \cdot T \cdot T_d}{2 \cdot T_i \cdot (T \cdot N + 2 \cdot T_d)} \\ b_1 = -\dfrac{4 \cdot T_d}{T \cdot N + 2 \cdot T_d}; \quad b_2 = \dfrac{2 \cdot T_d - T \cdot N}{T \cdot N + 2 \cdot T_d} \end{cases} \tag{3}$$

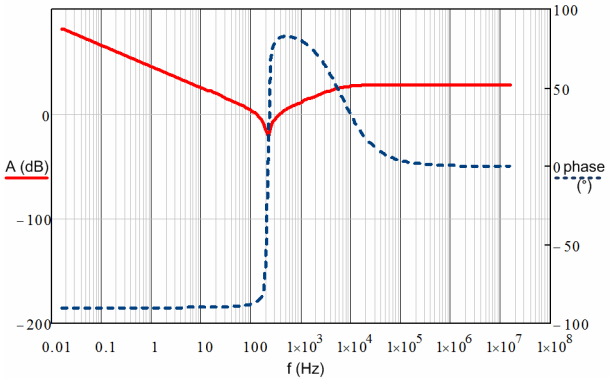

Figure 1. Bode plot of the PID controller.

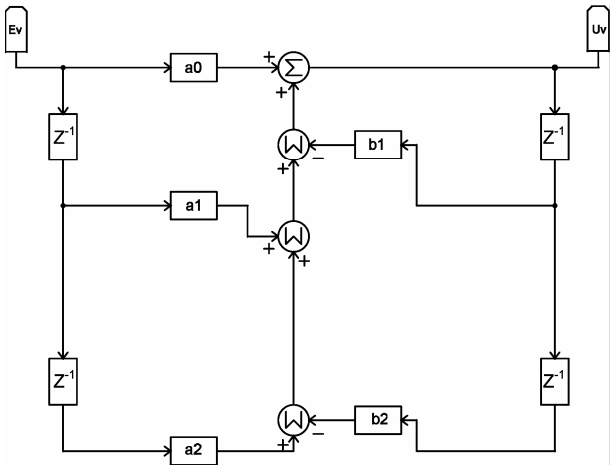

Figure 2. Time domain implementation of the control law.

The time domain transfer function is given in (4), (5) and the corresponding schematic is given in figure 2.

$$u(n) = E_1 - E_2, \tag{4}$$

$$\begin{cases} E_2 = a_0 \cdot e(n) + a_1 \cdot e(n-1) + (a_{21} + a_{22}) \cdot e(n-2) \\ E_2 = b_1 \cdot u(n-1) + b_2 \cdot u(n-2) \end{cases}, \tag{5}$$

The two algorithms were implemented for the step-down converter specified as follows:

$$\begin{cases} V_g = 9V \\ V_0 = 4V \\ R = 1.75\Omega \\ V_{ref} = 0.8V \end{cases} \quad \begin{cases} \beta = 0.2 \\ L = 29\mu H \\ C = 660\mu F \\ R_L = 5m\Omega, ESR = 20m\Omega \end{cases} \tag{6}$$

The coefficient β in (6) denotes the feedback network, which is a resistive voltage divider. It directly relates the nominal output voltage V_o to the reference value V_{ref}.

The simplified schematic of the converter is presented in figure 3. The switch on the output load is normally open. The disturbance is generated by closing the switch, which produces an overall load of 1.75Ω.

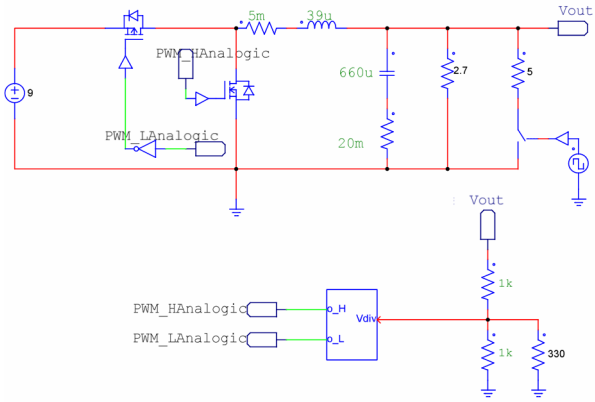

Figure 3. Schematic of the entire feedback loop.

The PID coefficients are given in (7). They were selected by using the pole placement method, which is intuitive, simple and reliable.

$$\begin{cases} K_p = 0.05 \\ T_i = 14.44\mu s \\ T_d = 8.91ms \\ N = 100 \end{cases} \quad (7)$$

The PID control was implemented in Mathcad and Psim. The Bode plots of the power plant and the PID controller transfer functions are presented in figure 4. The power plant model assumes no ripples on the inductor current ant output capacitor voltage. This leads to a linear equivalent of the step down converter. No dead time or processing delay was considered for the development of the transfer functions in figure 4.

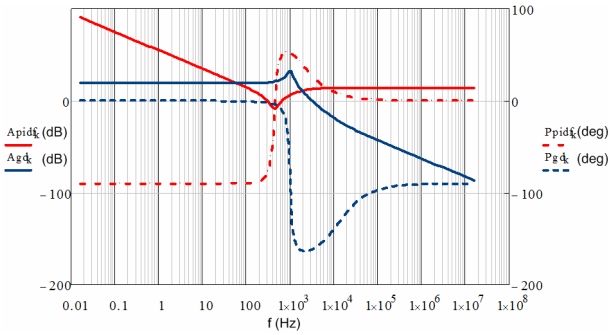

Figure 4. Bode plots of the PID control (red) and step down converter (blue).

The resulting open loop transfer function with and without a sampling time delay is presented in figure 5.

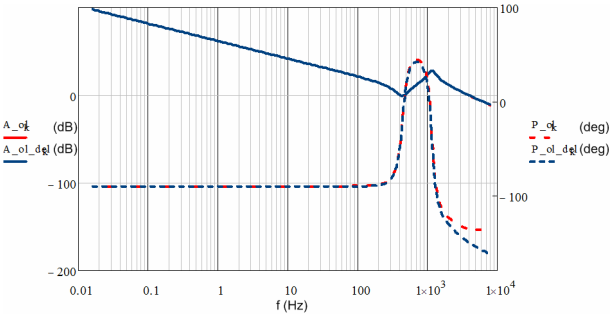

Figure 5. Bode plot of the open loop: no delay (red); 1 T_{sw} delay (red).

In order to account for the maximum delay, the switching period of 10µs is considered because it corresponds to the sampling rate of the feedback voltage. The delay takes the form of an exponential function in the s domain. However, considering a crossover frequency of 4 kHz as in figure 5, the expansion of this function into an infinite polynomial series yields

$$e^{-s\cdot 10^{-5}} \approx 1 - s\cdot 10^{-5} + \frac{\left(s\cdot 10^{-5}\right)^2}{2} \quad (8)$$

The dead time on the left hand side of (8) does not influence the Bode magnitude plot. However, it leads to the reduction of the phase margin in the overall open loop transfer function. The crossover frequency is relevant in the approximation above because the risk of instability is averted at frequencies which exceed it.

The augmented open-loop transfer function is presented in figure 5. It can be seen that the delay reduces the phase margin by approximately 20 degrees. Considering a transition in the load value from 2.7Ω to 1.75Ω, the ripple-free step response of the output voltage is presented in figure 6.

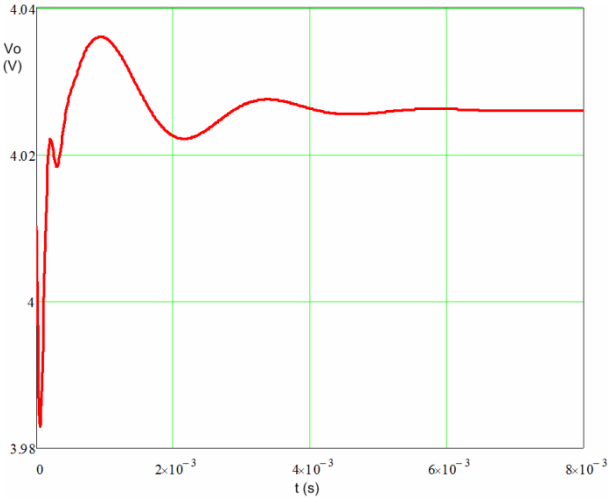

Figure 6. Step response modeled in Mathcad.

The z domain transfer function of the PID controller was determined by using the bilinear transform. The discrete time domain equivalent control law was expressed in (4) and was implemented in PSIM. The resulting output voltage transient response under the same step conditions as in the Mathcad model is presented in figure 7.

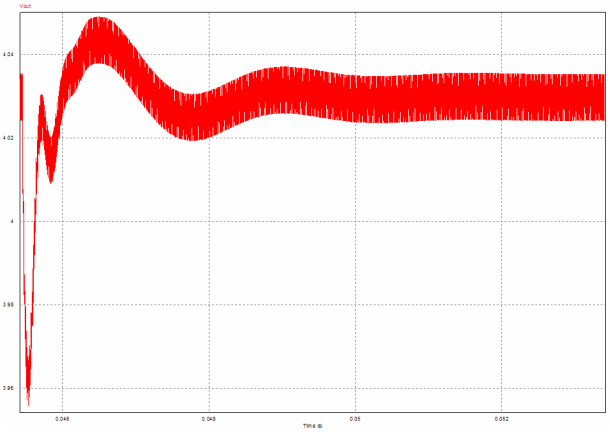

Figure 7. Step response simulated in Psim.

It can be seen that the two results, in Mathcad and in Psim, are correlated. The difference between the two is the ripple voltage in Psim, which is not present in Mathcad. The reason for this is the fact that the inductor current and output capacitor voltage are averaged in the model which was developed in Mathcad. This was necessary in order to produce a linear equivalent circuit of the step down converter and subsequently visualize the Bode plot in figure 4 and design a working PID controller.

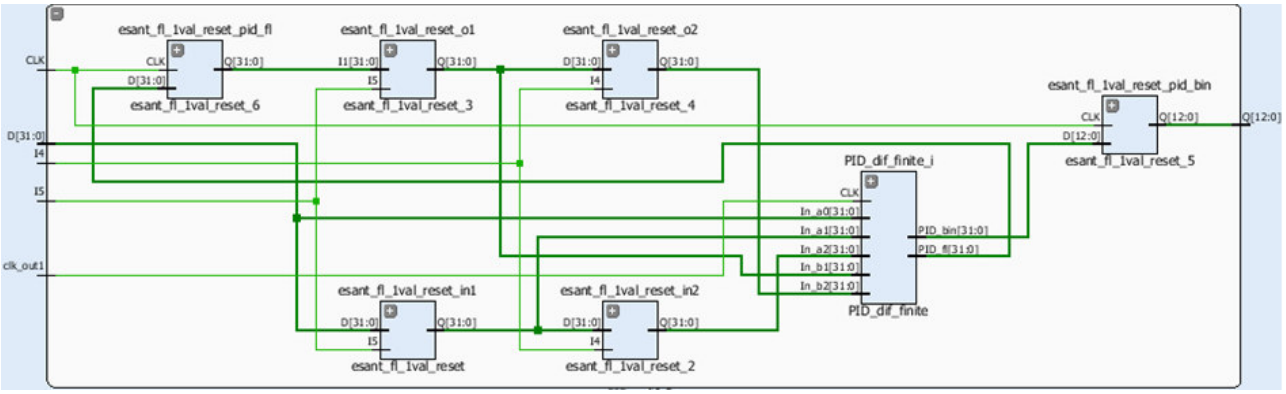

Figure 8. Sampling blocks of the implemented controller.

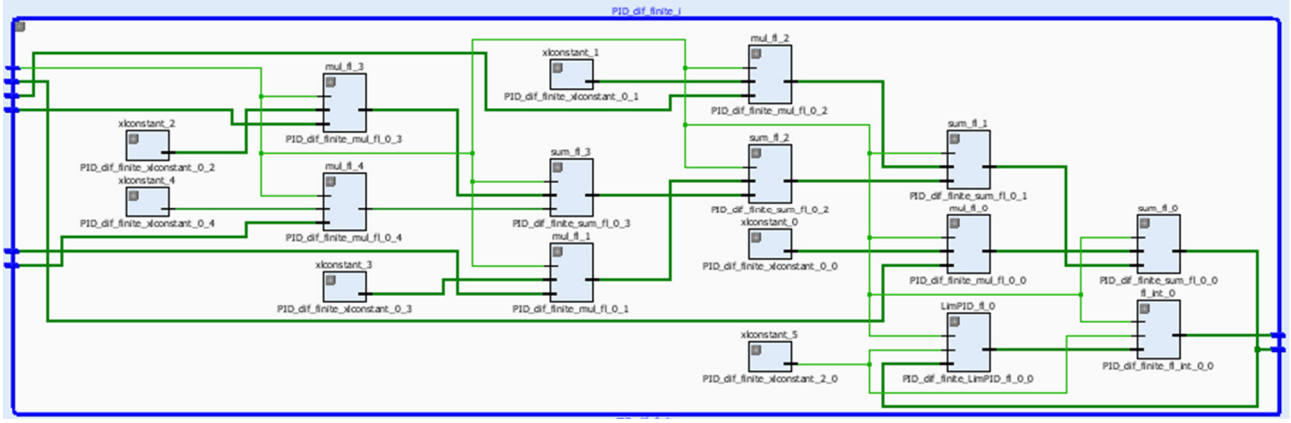

Figure 9. Implementation of the time domain control law.

III. PHYSICAL IMPLEMENTATION AND EXPERIMENTAL RESULTS

The controller was implemented using a Nexys 4 development board which is equipped with an Artix 7 FPGA manufactured by Xilinx. It features an ADC which converts at a maximum rate of 1 MSPS. An internal reference voltage of 1V is used and the resolution is 12 bits.

The switching frequency of the step-down converter was set to 100 kHz and the ADC output was sampled at the beginning of every switching period. Thus, the total worst-case delay is 10µs.

The output of the ADC is in binary format. However, the results in (4) require the use of fractional numbers. Therefore, the ADC output was converted to single-precision floating-point format and the floating-point operation block was used. This block is part of the Vivado ISE library and supports various arithmetical operations in single-precision floating-point format. The operands are 32 bits wide. Upon design synthesis, it is possible to check the resource utilization of the various floating-point operators:

Table 1. Resource utilization by the arithmetic operands.

	+	-	*
Min del	1 T_{CLK}	1 T_{CLK}	1 T_{CLK}
LUTs	389	389	801
Registers	135	135	135
Muxes	6	6	0

Employing the aforementioned operations in (4), it was possible to implement the time domain control law of the PID regulator. The total resource utilization of the control block is 5561 LUTs, 1215 registers and 24 muxes. The overall processing delay is 5 clock periods.

The schematic of the implemented time domain control law described in (4) is presented in figures 8 and 9. In order to preserve clarity, the schematic is broken into two figures: figure 8 does not detail the arithmetic operations, but it presents the delay blocks.

Figure 9 presents the arithmetic block, without the delay z^{-1} blocks, which is why there are 5 32 bit wide inputs in addition to the CLK signal. The combined schematics in figures 8 and 9 make up the implemented version of the simplified version presented in figure 2.

In order to validate the designed controller, its impulse response was compared to that modeled in Psim. The results are presented in figures 10 and 11.

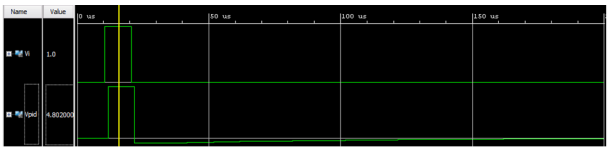

Figure 10. Impulse response behavioral simulation - Vivado.

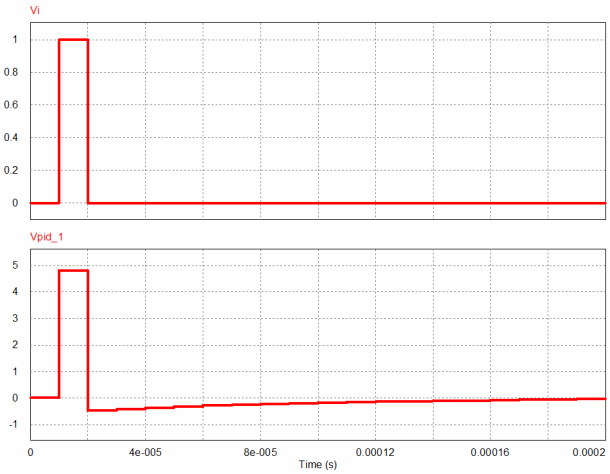

Figure 11. Impulse response simulation - Psim.

The step response under the same conditions as those considered in the Mathcad model and Psim simulation is presented in figure 12.

Figure 12. Step response - experimental result.

The purple waveform is the AC coupled output voltage represented as 20mV/div. The orange waveform signifies the command signal on the parallel supplementary 5Ω.

An FPGA-based PID controller was implemented. The controller was tuned to regulate the output of a step down converter. Both the controller and the converter were modeled in Mathcad and their Bode plots were presented. A step response was plotted.

The feedback loop comprising the same converter and PID regulator was simulated in Psim. Both a step response of the entire feedback loop and the impulse response of the isolated controller were plotted.

The unit response of the implemented control was simulated. The overall feedback loop was implemented and the experimental step response was presented.

ACKNOWLEDGEMENT

This paper is supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), ID/134378 financed from the European Social Fund and by the Romanian Government.

REFERENCES

[1] D. Petreus, D. Moga, A. Rusu, T. Patarau, S. Daraban, "A Maximum Power Point tracker for a photovoltaic system under changing luminosity conditions," Industrial Electronics (ISIE), 2010 IEEE International Symposium on , pp.556,561, 4-7 July 2010

[2] F. Poulain, L. Praly, and R. Ortega, “An observer for permanent magnet synchronous motors with currents and voltages as only measurements,” in Proc. 47th IEEE CDC, Dec. 2008, pp. 5390–5395.

[3] D. Maksimovic, R. Zane, R. Erickson, "Impact of digital control in power electronics," Power Semiconductor Devices and ICs, 2004. Proceedings. ISPSD '04. The 16th International Symposium on , vol., no., pp.13,22, 24-27 May 2004

[4] Y. Li, K. H. Ang, and G. C. Y. Chong, “Patents, software, hardware for PID control,” IEEE Control Syst. Mag., vol. 26, no. 1, pp. 42–54, Feb. 2006.

[5] M. Araki, “PID Control,” Control Syst., Robot. Autom., vol. 2, pp. 1–7, 2002

[6] R. D. Middlebrook and S. Cuk, “A general unified approach to modeling switching-converter power stages,” in Proc. IEEE PESC,1976, pp. 18–34.

[7] J. G. Ziegler and N. B. Nichols, “Optimum settings for automatic controllers,”Trans. ASME, vol. 64, no. 8, pp. 759–768, Nov. 1942.

[8] G. M. Malwatkar, S. H. Sonawane, and L. M. Waghmare, “Tuning PID controllers for higher-order oscillatory systems with improved performance,”ISA Trans., vol. 48, no. 3, pp. 347–353, Jul. 2009.

[9] W. Li, X. G. Chang, F. M. Wahl, and J. Farrell, “Tracking control of a manipulator under uncertainty by fuzzy PID controller,”Fuzzy Sets Syst., vol. 122, no. 1, pp. 125–137, Aug. 2001.

[10] I. Cervantes and J. Alvarez-Ramirez, “On the PID tracking control of robot manipulators,”Syst. Control Lett., vol. 42, no. 1, pp. 37–46, Jan. 2001.

[11] A. O’Dwyer,Handbook of PI and PID Controller Tuning Rules,3rded. London, UK: Imperial College Press, 2009.

[12] J. L. Meza, V. Santibanez, R. Soto, and M. A. Llama, “Fuzzy selftuning PID semiglobal regulator for robot manipulators,”IEEE Trans. Ind. Electron., vol. 59, no. 6, pp. 2709–2717, Jun. 2012.

[13] X.-G. Duan, H. Deng, and H.-X. Li, “A saturation-based tuning method for fuzzy PID controller,” IEEE Trans. Ind. Electron., vol. 60, no. 11, pp. 5177–5185, Nov. 2013.

[14] X. Jing and L. Cheng, “An optimal PID control algorithm for training feedforward neural networks,”IEEE Trans. Ind. Electron., vol. 60, no. 6, pp. 2273–2283, Jun. 2013.

[15] K. S. Tang, K. F. Man, G. Chen, and S. Kwong, “An optimal fuzzy PID controller,”IEEE Trans. Ind. Electron., vol. 48, no. 4, pp. 757–765, Aug. 2001.

[16] L. Idkhajine, E. Monmasson, and A. Maalouf, “Fully FPGA-based sensorless control for synchronous ac drive using an extended Kalman filter,” IEEE Trans. Ind. Electron., vol. 59, no. 10, pp. 3908–3918, Oct. 2012.

[17] K. Ying-Shieh and T. Ming-Hung, “FPGA-based speed control IC for PMSM drive with adaptive fuzzy control,”IEEE Trans. Power Electron., vol. 22, no. 6, pp. 2476–2486, Nov. 2007.

[18] V. D. Colli, R. D. Stefano, and F. Marignetti, “A system-on-chip sensorless control for a permanent magnet synchronous motor,”IEEE Trans. Ind. Electron., vol. 57, no. 11, pp. 3822–3829, Nov. 2010.

[19] Q. P. Ha, Y.-H. Yu, and N. K. Quang, “FPGA-based cooperative control of indoor multiple robots,”Int. J. Adv. Mechatron. Syst., vol. 4, no. 5/6, pp. 248–259, 2012.