Adaptive wavelet neural controller design for a DC-DC power converter using an FPGA chip

Yu-Hsiung Lin and Chun-Fei Hsu*
Department of Electrical Engineering
Chung Hua University
Hsinchu 300, Taiwan, Republic of China
707, Sec.2, WuFu Rd., Hsinchu, Taiwan 30012, R.O.C.
bear@chu.edu.tw; fei@chu.edu.tw

Abstract: - The DC-DC power converters are widely used; however, the controller design for the DC-DC power converters cannot easily design if the load dynamics vary widely. This paper proposes an adaptive wavelet neural control (AWNC) system for a forward DC-DC power converter. The proposed AWNC system is composed of a neural controller and a robust controller. The neural controller uses a wavelet neural network to online mimic an ideal controller and the robust controller is designed to cope with the approximation error between the neural controller and the ideal controller. A proportional-integral (PI) type parameter tuning mechanism is derived based on the Lyapunov stability theory; thus not only the system stability can be guaranteed but also the convergence of the tracking error can be speeded up. Finally, a comparison among the PI controller, the fuzzy controller, the adaptive neural fuzzy control, the adaptive neuro-wavelet control and the proposed AWNC is made. The experimental results show that the proposed AWNC system is robust with regard to different input voltages and load resistance variations.

Key-Words: - adaptive control; robust control; sliding-mode control, wavelet neural network; DC-DC power converter.

1 Introduction

The DC-DC power converters are widely used in computers, electrical peripheral, and etc. The output voltage of the DC-DC power converter is controlled by adjusting the ON time of the switching action, which in turn adjusts the width of a voltage pulse at the output. This is known as pulse-width-modulation (PWM) approach. By varying the duty-ratio of PWM, the DC-DC power converter can convert one level of electrical voltage to the other level of electrical voltage [1]. Recently, a series of papers have considered the control of the DC-DC converters based on the PI controller [2] and sliding-mode controller [3, 4] since they are simple to implement and easy to design. However, a trade-off problem between the robustness and transient response arises in the PI controller and sliding-mode controller. In general, a DC-DC power converter with fixed controller parameters cannot perform well if the load dynamics vary widely.

Since the digital controller have been shown the feasibility, the digitally controlled DC-DC power converter began to appear in [5-10]. One interesting potential benefit is the use of auto-tuning controller parameters, so the dynamic response can be set at the software level. In [5], a fuzzy controller can achieve a desired regulator performance; however, the fuzzy rules base should be pre-selected by some trial-and-error tuning procedure. In [6], [7] and [9], though the satisfactory regulator performance can be achieved using the online tuning algorithm, the computation loading of these methods is too heavy. An auto-tuning PID controller which can automatically online tune the control gains is proposed in [8]. However, it should know the transfer function of the controlled converter. In [10], an auto-tuning adaptive digital controller with maximum efficiency point tracking is presented. However, it lacked the stability analytics for the closed-loop control system.

If the exact model of the controlled dynamics is known, there exists an ideal controller to achieve a favorable control performance [11]. However, the mathematical model is usually imprecise or unobtainable because of the existence of measurement and parameter uncertainties. To attack this problem, several adaptive neural controllers have been proposed in [12-15] which provide online learning algorithms that don't require preliminary off-line tuning. To achieve a better learning performance, some researchers have developed the structure of neural network based on the wavelet

functions to construct the wavelet neural network (WNN) [16, 17]. There have been considerable interests in exploring the applications of WNN to deal with the non-linearity and uncertainty in control systems [18-20].

This paper proposed an adaptive wavelet neural control (AWNC) system to control the duty-ratio of PWM for a forward DC-DC power converter. The proposed AWNC system is composed of a neural controller and a robust controller. The neural controller uses a WNN to mimic an ideal controller and the robust controller is designed to eliminate the effect of the approximation error introduced by the neural controller upon the system stability in the Lyapunov sense. Moreover, a proportional-integral (PI)-type parameter tuning mechanism is derived to speed up the convergence of the tracking error. Finally, a field-programmable gate array (FPGA) chip is adopted to implement the AWNC scheme for possible low-cost and high-performance industrial applications. To investigate the effectiveness of the proposed AWNC system, a comparison among the PI controller [2], the fuzzy controller [5], the adaptive neural fuzzy control [6], the adaptive neuro-wavelet control [7] and the proposed AWNC system is made. It is shown by the experimental results that the proposed AWNC system provides better regulator performance than the other control schemes.

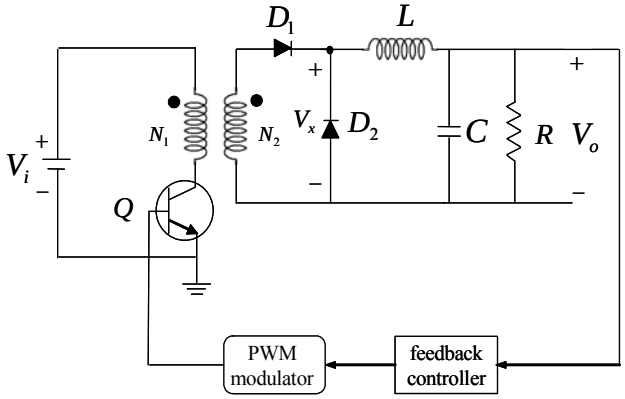

Fig. 1 The forward DC-DC power converter.

2 Problem Formulation

DC-DC power converter is very popular because of its high efficiency and small size. In this study, a forward DC-DC power converter is discussed as shown in Fig. 1, where V_i and V_o are the input and output voltages of the converter, respectively, D_1 and D_2 are the diodes, L is the inductor, C is the output capacitor, R is the resistor, and Q is the transistor which control the converter circuit operating in different modes. When the transistor is ON, V_i appears across the primary and then generates [1, 5]

$$V_x = \frac{N_2}{N_1} V_i \tag{1}$$

where N_1 is the turns of primary power winding and N_2 is the turns of slave power winding. By the averaging method, the output voltage can be expressed as [1]

$$V_o = \frac{N_2}{N_1} V_i d \,. \tag{2}$$

Equation (2) shows that the DC-DC power converter can convert one level of electrical voltage to the desired level by varying the duty-ratio of PWM. The switch frequency of PWM is constant and the duty cycle, $d(N)$, varies with the load resistance variations at the N-th sampling time. The output of the designed controller, $\delta d(N)$, is the change of duty cycle. Then, the duty cycle is determined by adding the previous duty cycle $d(N-1)$ to the change of duty cycle $\delta d(N)$, i.e. [1]

$$d(N) = d(N-1) + \delta d(N) \,. \tag{3}$$

Considering N_1, N_2 and V_i as constants and differentiating both sides of (2) with respect to time, yield

$$\dot{V}_o = \frac{N_2}{N_1} V_i \delta \dot{d} = g\, u \tag{4}$$

where $g \equiv \frac{N_2}{N_1} V_i$ is the positive constant control gain and $u \equiv \delta \dot{d}$ is the control input. The control problem of the DC-DC power converter is to control the change of duty cycle $\delta \dot{d}$ so the output voltage V_o can provide a desired output voltage. The output error voltage is defined as

$$e = V_{ref} - V_o \tag{5}$$

where V_{ref} is the output reference voltage. If the system parameters are known, an ideal controller can be designed as [11]

$$u^* = g^{-1}(\dot{V}_{ref} + ke) \,. \tag{6}$$

where k is a positive constant. Substituting (6) into (4), we can obtain

$$\dot{e} + ke = 0 \,. \tag{7}$$

It is implied that $\lim_{t\to\infty} e(t) = 0$. Since the term g may be unknown or perturbed, the ideal controller cannot be precisely implemented in the real-time applications.

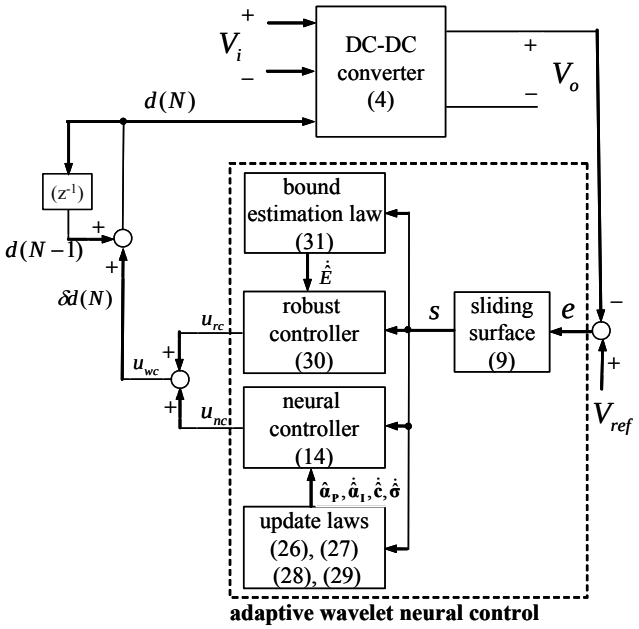

Fig. 2 The block diagram of AWNC for the DC-DC power converter.

3 Design of the AWNC System

This paper proposes an AWNC design method as shown in Fig. 2, where the controller output is defined as

$$u_{wc} = \hat{u}_{nc} + u_{rc}. \tag{8}$$

The neural controller $\hat{u}_{nc}$ is investigated to mimic the ideal controller and the robust controller u_{rc} is designed to compensate for the approximation error between the neural controller and the ideal controller. The sliding surface is defined as

$$s = e + k\int_0^t e(\tau)d\tau. \tag{9}$$

where k is a positive constant.

3.1 Wavelet neural network

The WNN output with m wavelet basis functions can perform the mapping according to [18, 20]

$$u_{nc} = \sum_{j=1}^{m} \alpha_j \Theta_j(\sigma_j, (s - c_j)) \tag{10}$$

where s is the input variable, $\Theta_j(\sigma_j, (s-c_j))$ is the wavelet function, σ_j and c_j are the dilation and translation parameters, respectively, α_j is the output layer weight and m is the number of neurons in the translation layer. Each wavelet network's neuron with "Mexican hat" mother wavelet function in the translation layer can be represented by

$$\Theta_j = (1 - \omega^2 (s-c_j)^2)\exp(-\sigma_j^2 (s-c_j)^2). \tag{11}$$

where ω is a positive constant. For ease of notation, (10) can be expressed in a vector form as

$$u_{nc} = \boldsymbol{\alpha}^T \boldsymbol{\Theta}(s, \boldsymbol{\sigma}, \mathbf{c}) \tag{12}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_m]^T$, $\boldsymbol{\Theta} = [\Theta_1, \Theta_2, \ldots, \Theta_m]^T$, $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_m]^T$ and $\mathbf{c} = [c_1, c_2, \ldots, c_m]^T$. By the universal approximation theorem, there exists an optimal WNN as [16, 18]

$$u^* = u_{nc}^* + \Delta = \boldsymbol{\alpha}^{*T} \boldsymbol{\Theta}(s, \boldsymbol{\sigma}^*, \mathbf{c}^*) + \Delta = \boldsymbol{\alpha}^{*T} \boldsymbol{\Theta}^* + \Delta \tag{13}$$

where Δ denotes the approximation error, $\boldsymbol{\alpha}^*$ and $\boldsymbol{\Theta}^*$ are the optimal parameter vectors of $\boldsymbol{\alpha}$ and $\boldsymbol{\Theta}$, respectively, $\boldsymbol{\sigma}^*$ and $\mathbf{c}^*$ are the optimal parameter vectors of $\boldsymbol{\sigma}$ and $\mathbf{c}$, respectively. In fact, the optimal parameter vectors are needed to best approximate are difficult to determine. Thus, an estimate WNN $\hat{u}_{nc}$ is defined as

$$\hat{u}_{nc} = \hat{\boldsymbol{\alpha}}^T \boldsymbol{\Theta}(s, \hat{\boldsymbol{\sigma}}, \hat{\mathbf{c}}) = \hat{\boldsymbol{\alpha}}^T \hat{\boldsymbol{\Theta}} \tag{14}$$

where $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\Theta}}$ are the estimation vector of $\boldsymbol{\alpha}$ and $\boldsymbol{\Theta}$, respectively, and $\hat{\boldsymbol{\sigma}}$ and $\hat{\mathbf{c}}$ are the estimation vector of $\boldsymbol{\sigma}$ and $\mathbf{c}$, respectively. To speed up the parameter convergence, the optimal parameter vector $\boldsymbol{\alpha}^*$ is decomposed into two parts as [21]

$$\boldsymbol{\alpha}^* = \eta_P \boldsymbol{\alpha}_\mathbf{P}^* + \eta_I \boldsymbol{\alpha}_\mathbf{I}^* \tag{15}$$

where $\boldsymbol{\alpha}_\mathbf{P}^*$ and $\boldsymbol{\alpha}_\mathbf{I}^*$ are the proportional and integral terms of $\boldsymbol{\alpha}^*$, respectively, η_P and η_I are positive coefficients, and $\boldsymbol{\alpha}_\mathbf{I}^* = \int_0^t \boldsymbol{\alpha}_\mathbf{P}^* d\tau$. Similarly, the estimation parameter vector $\hat{\boldsymbol{\alpha}}_\mathbf{a}$ is decomposed into two parts as [21]

$$\hat{\boldsymbol{\alpha}} = \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P} + \eta_I \hat{\boldsymbol{\alpha}}_\mathbf{I} \tag{16}$$

where $\hat{\boldsymbol{\alpha}}_\mathbf{P}$ and $\hat{\boldsymbol{\alpha}}_\mathbf{I}$ are the proportional and integral terms of $\hat{\boldsymbol{\alpha}}$, respectively, and $\hat{\boldsymbol{\alpha}}_\mathbf{I} = \int_0^t \hat{\boldsymbol{\alpha}}_\mathbf{P} d\tau$. Thus, $\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha}^* - \hat{\boldsymbol{\alpha}}$ can be expressed as

$$\tilde{\boldsymbol{\alpha}} = \eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I} - \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P} + \eta_P \boldsymbol{\alpha}_\mathbf{P}^* \tag{17}$$

where $\tilde{\boldsymbol{\alpha}}_\mathbf{I} = \boldsymbol{\alpha}_\mathbf{I}^* - \hat{\boldsymbol{\alpha}}_\mathbf{I}$. Define the estimated error $\tilde{u}$ as

$$\begin{aligned}
\tilde{u} &= u^* - \hat{u}_{nc} \\
&= \boldsymbol{\alpha}^{*T} \boldsymbol{\Theta}^* - \hat{\boldsymbol{\alpha}}^T \hat{\boldsymbol{\Theta}} + \Delta \\
&= \tilde{\boldsymbol{\alpha}}^T \hat{\boldsymbol{\Theta}} + \hat{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \tilde{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \Delta \\
&= (\eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I} - \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P} + \eta_P \boldsymbol{\alpha}_\mathbf{P}^*)^T \hat{\boldsymbol{\Theta}} + \hat{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \tilde{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \Delta \\
&= \eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \hat{\boldsymbol{\Theta}} - \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P}^T \hat{\boldsymbol{\Theta}} + \eta_P \boldsymbol{\alpha}_\mathbf{P}^{*T} \hat{\boldsymbol{\Theta}} + \hat{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \tilde{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \Delta
\end{aligned} \tag{18}$$

where $\tilde{\boldsymbol{\Theta}} = \boldsymbol{\Theta}^* - \hat{\boldsymbol{\Theta}}$. The Taylor expansion linearization technique is employed to transform the nonlinear function into a partially linear form [13], i.e.

$$\tilde{\boldsymbol{\Theta}} = \mathbf{A}^T \tilde{\boldsymbol{\sigma}} + \mathbf{B}^T \tilde{\mathbf{c}} + \mathbf{h} \tag{19}$$

where $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}^* - \hat{\boldsymbol{\sigma}}$, $\tilde{\mathbf{c}} = \mathbf{c}^* - \hat{\mathbf{c}}$, $\mathbf{h}$ is a vector of higher-order terms, $\mathbf{A} = \left[\frac{\partial\Theta_1}{\partial\boldsymbol{\sigma}} \frac{\partial\Theta_2}{\partial\boldsymbol{\sigma}} \cdots \frac{\partial\Theta_m}{\partial\boldsymbol{\sigma}}\right]|_{\boldsymbol{\sigma}=\hat{\boldsymbol{\sigma}}}$ and $\mathbf{B} = \left[\frac{\partial\Theta_1}{\partial\mathbf{c}} \frac{\partial\Theta_2}{\partial\mathbf{c}} \cdots \frac{\partial\Theta_m}{\partial\mathbf{c}}\right]|_{\mathbf{c}=\hat{\mathbf{c}}}$. Substitution of (19) into (18) yields

$$\begin{aligned}\tilde{u} &= \eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \hat{\boldsymbol{\Theta}} - \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P}^T \hat{\boldsymbol{\Theta}} + \eta_P \boldsymbol{\alpha}_\mathbf{P}^{*T} \hat{\boldsymbol{\Theta}} \\ &\quad + \hat{\boldsymbol{\alpha}}^T (\mathbf{A}^T \tilde{\mathbf{c}} + \mathbf{B}^T \tilde{\boldsymbol{\sigma}} + \mathbf{h}) + \tilde{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \Delta \\ &= \eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \hat{\boldsymbol{\Theta}} - \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P}^T \hat{\boldsymbol{\Theta}} + \tilde{\mathbf{c}}^T \mathbf{A} \hat{\boldsymbol{\alpha}} + \tilde{\boldsymbol{\sigma}}^T \mathbf{B} \hat{\boldsymbol{\alpha}} + \varepsilon \end{aligned} \quad (20)$$

where $\hat{\boldsymbol{\alpha}}^T \mathbf{A}^T \tilde{\mathbf{c}} = \tilde{\mathbf{c}}^T \mathbf{A} \hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\alpha}}^T \mathbf{B}^T \tilde{\boldsymbol{\sigma}} = \tilde{\boldsymbol{\sigma}}^T \mathbf{B} \hat{\boldsymbol{\alpha}}$ are used since they are scalars and $\varepsilon = \hat{\boldsymbol{\alpha}}^T \mathbf{h} + \tilde{\boldsymbol{\alpha}}^T \tilde{\boldsymbol{\Theta}} + \eta_P \boldsymbol{\alpha}_\mathbf{P}^{*T} \hat{\boldsymbol{\Theta}} + \Delta$ denotes the lump of approximation error. It is assumed to be bounded by $0 \le |\varepsilon| \le E$ where E is a positive constant.

3.2 Controller design

Substituting (8) into (4) yields

$$\dot{V}_o = g(\hat{u}_{nc} + u_{rc}). \quad (21)$$

From (6) and (21), the error dynamic equation can be obtained

$$\dot{s} = g(u^* - \hat{u}_{nc} - u_{rc}). \quad (22)$$

Using (20), the error dynamic equation can be re-expressed as

$$\dot{s} = g(\eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \hat{\boldsymbol{\Theta}} - \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P}^T \hat{\boldsymbol{\Theta}} + \tilde{\mathbf{c}}^T \mathbf{A} \hat{\boldsymbol{\alpha}} + \tilde{\boldsymbol{\sigma}}^T \mathbf{B} \hat{\boldsymbol{\alpha}} + \varepsilon - u_{rc}). \quad (23)$$

To proof the stability of the AWNC system, define a Lyapunov function candidate in the following form

$$\begin{aligned} V(s, \tilde{\boldsymbol{\alpha}}, \tilde{\mathbf{c}}, \tilde{\boldsymbol{\sigma}}, \tilde{E}, t) &= \frac{1}{2} s^2 \\ &\quad + g(\eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \tilde{\boldsymbol{\alpha}}_\mathbf{I} + \frac{1}{2\eta_2} \tilde{\mathbf{c}}^T \tilde{\mathbf{c}} + \frac{1}{2\eta_3} \tilde{\boldsymbol{\sigma}}^T \tilde{\boldsymbol{\sigma}} + \frac{1}{2\eta_4} \tilde{E}^2) \end{aligned} \quad (24)$$

where η_2, η_3 and η_4 are positive constants and $\tilde{E} = E - \hat{E}$ in which $\hat{E}$ is the estimated error bound. Differentiating (24) with respect to time and using (23), we have

$$\begin{aligned} \dot{V} &= s\dot{s} + \eta_I g \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \dot{\tilde{\boldsymbol{\alpha}}}_\mathbf{I} + \frac{g}{\eta_2} \tilde{\mathbf{c}}^T \dot{\tilde{\mathbf{c}}} + \frac{g}{\eta_3} \tilde{\boldsymbol{\sigma}}^T \dot{\tilde{\boldsymbol{\sigma}}} + \frac{g}{\eta_4} \tilde{E} \dot{\tilde{E}} \\ &= sg(\eta_I \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \hat{\boldsymbol{\Theta}} - \eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P}^T \hat{\boldsymbol{\Theta}} + \tilde{\mathbf{c}}^T \mathbf{A} \hat{\boldsymbol{\alpha}} + \tilde{\boldsymbol{\sigma}}^T \mathbf{B} \hat{\boldsymbol{\alpha}} + \varepsilon - u_{rc}) \\ &\quad + \eta_I g \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \dot{\tilde{\boldsymbol{\alpha}}}_\mathbf{I} + \frac{g}{\eta_2} \tilde{\mathbf{c}}^T \dot{\tilde{\mathbf{c}}} + \frac{g}{\eta_3} \tilde{\boldsymbol{\sigma}}^T \dot{\tilde{\boldsymbol{\sigma}}} + \frac{g}{\eta_4} \tilde{E} \dot{\tilde{E}} \\ &= \tilde{\boldsymbol{\alpha}}_\mathbf{I}^T \eta_I g(s\hat{\boldsymbol{\Theta}} + \dot{\tilde{\boldsymbol{\alpha}}}_\mathbf{I}) - sg\eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P}^T \hat{\boldsymbol{\Theta}} + \tilde{\mathbf{c}}^T g(s\mathbf{A}\hat{\boldsymbol{\alpha}} + \frac{\dot{\tilde{\mathbf{c}}}}{\eta_2}) \\ &\quad + \tilde{\boldsymbol{\sigma}}^T g(s\mathbf{B}\hat{\boldsymbol{\alpha}} + \frac{\dot{\tilde{\boldsymbol{\sigma}}}}{\eta_3}) + sg(\varepsilon - u_{rc}) + \frac{g}{\eta_4} \tilde{E} \dot{\tilde{E}}. \end{aligned} \quad (25)$$

If the adaptive laws of the neural controller are designed as

$$\dot{\hat{\boldsymbol{\alpha}}}_\mathbf{P} = s\hat{\boldsymbol{\Theta}} \quad (26)$$

$$\dot{\hat{\boldsymbol{\alpha}}}_\mathbf{I} = -\dot{\tilde{\boldsymbol{\alpha}}}_\mathbf{I} = s\hat{\boldsymbol{\Theta}} \quad (27)$$

$$\dot{\hat{\mathbf{c}}} = -\dot{\tilde{\mathbf{c}}} = \eta_2 s\mathbf{A}\hat{\boldsymbol{\alpha}} \quad (28)$$

$$\dot{\hat{\boldsymbol{\sigma}}} = -\dot{\tilde{\boldsymbol{\sigma}}} = \eta_3 s\mathbf{B}\hat{\boldsymbol{\alpha}} \quad (29)$$

and robust controller is designed as

$$u_{rc} = \hat{E}\,\mathrm{sgn}(s) \quad (30)$$

with the bound estimation law

$$\dot{\hat{E}} = -\dot{\tilde{E}} = \eta_4 |s| \quad (31)$$

then (25) can be rewritten as

$$\begin{aligned} \dot{V} &= -g\eta_P \hat{\boldsymbol{\alpha}}_\mathbf{P}^T \hat{\boldsymbol{\alpha}}_\mathbf{P} + \varepsilon sg - \hat{E}|s|g + \frac{g}{\eta_4} \tilde{E} \dot{\tilde{E}} \\ &\le \varepsilon sg - \hat{E}|s|g + \frac{g}{\eta_4} \tilde{E} \dot{\tilde{E}} \\ &= \varepsilon sg - E|s|g \\ &\le |\varepsilon||s|g - E|s|g \\ &= -(E - |\varepsilon|)|s|g \le 0. \end{aligned} \quad (32)$$

This implies that $\dot{V}$ is negative semidefinite. Define the following term

$$P \equiv (E - |\varepsilon|)|s|g \le -\dot{V}. \quad (33)$$

Because $V(s, \tilde{\boldsymbol{\alpha}}, \tilde{\mathbf{c}}, \tilde{\boldsymbol{\sigma}}, \tilde{E}, 0)$ is bounded and $V(s, \tilde{\boldsymbol{\alpha}}, \tilde{\mathbf{c}}, \tilde{\boldsymbol{\sigma}}, \tilde{E}, t)$ is nonincreasing and bounded, then

$$\int_0^t P(\tau) d\tau \le V(s, \tilde{\boldsymbol{\alpha}}, \tilde{\mathbf{c}}, \tilde{\boldsymbol{\sigma}}, \tilde{E}, 0) - V(s, \tilde{\boldsymbol{\alpha}}, \tilde{\mathbf{c}}, \tilde{\boldsymbol{\sigma}}, \tilde{E}, t) < \infty. \quad (34)$$

Also, because $\dot{P}(t)$ is bounded, it can be shown that $\lim_{t\to\infty} P(t) = 0$ by Barbalat's Lemma [11]. That is $s \to 0$ as $t \to \infty$, then the system stability of the proposed AWNC system can be guaranteed.

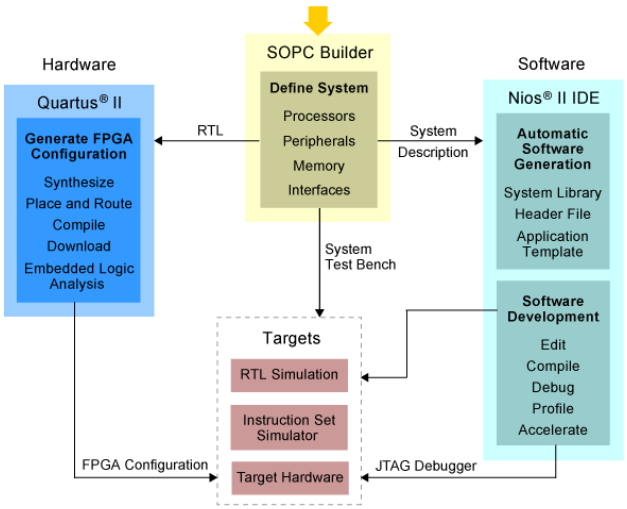

Fig. 3 Altera offers development tools.

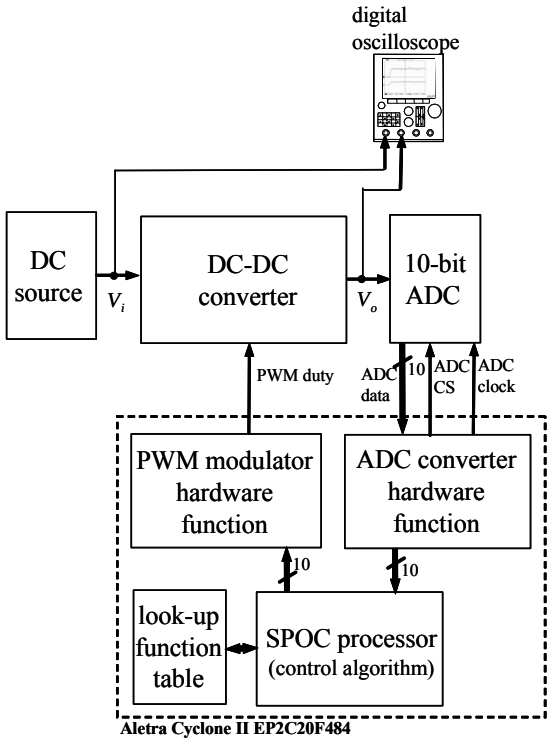

Fig. 4 Experimental setup.

4 Experimental Results

A field-programmable gate array (FPGA) chip incorporates the architecture of gate arrays and the programmability of a programmable logic device. An FPGA chip usually consists of more than thousands of logic gates. Some of these logic gates are combined together to form a configurable logic block thereby simplifying high-level circuit design. All the internal logic elements and all the control procedures of the FPGA are executed continuously and simultaneously [22, 23]. The circuits and algorithms can be developed by the very high speed integrated circuit hardware description language. Therefore, the execution time of FPGA is faster than either the digital signal processor (DSP) or the personal computer (PC).

Altera offers a variety of development tools to accelerate the embedded design process as shown in Fig. 3 [22]. Together with selected partners, Altera offers a range of software development tools, hardware development tools, and development kits that include everything you need to quickly and easily create and implement a design. The Quartus® II software is the development tool for programmable logic devices and the Nios® II Integrated Development Environment (IDE) is a collection of tools used to develop embedded software for the Nios II processor. The Nios II IDE includes a complete set of C/C++ software development tools is a collection of software generation, management, and deployment tools for the Nios II processor.

This study used the Aletra Cyclone II EP2C20F484 FPGA device, the Quartus® II, the Nios® II IDE and the Verilog hardware description language to implement the control system. The FPGA-based experimental setup for the forward DC-DC power converter is shown in Fig. 4. A 10-bit A/D converter (TLC1549) IC is used to measure the output voltage [24]. In the main program, the initialization of controller parameters is preceded. Next, the interrupt interval for the interrupt service routine (ISR) with a 0.1msec sampling rate is set. Then, the controller sample times can be governed by the built-in timer which generates periodic interrupts.

To investigate the effectiveness of the proposed AWNC system, a comparison among the PI controller [2], the fuzzy controller [5], the adaptive neural fuzzy control [6], the adaptive neuro-wavelet control [7] and the proposed AWNC system is made. Two experimental cases are addressed as follows: (1) Case 1 (the input voltage is set as $V_i = 30V$); (2) Case 2 (the input voltage is set as $V_i = 35V$). In both cases, some load resistance variations with step changes are tested: (i) from 100Ω to 10Ω at 120msec and (ii) from 10Ω to 100Ω at 240msec. The circuit parameter values of the forward DC-DC converter are chosen as $N_P : N_m = 4:3$, $R = 100\,\Omega$, $L = 50\,\mu H$ and $C = 150\,\mu F$. Moreover, a first-order transfer function is chosen as the reference model for periodic step command to use as a soft-start technique (i.e. the reference voltage V_{ref} evolves from 0 to the desired value linearly)

$$\frac{135}{S+135} \tag{35}$$

where *S* is the Laplace operator.

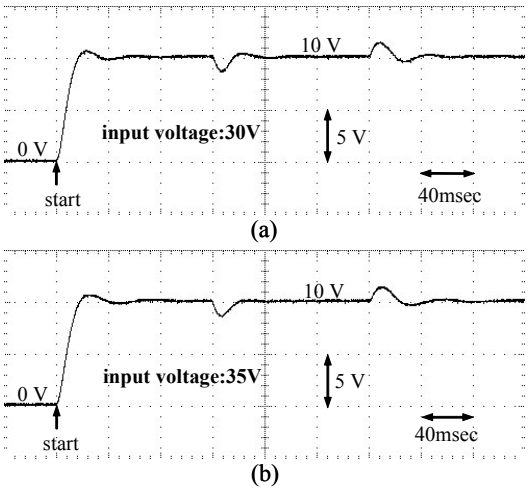

Fig. 5 Experimental results of the PI controller.

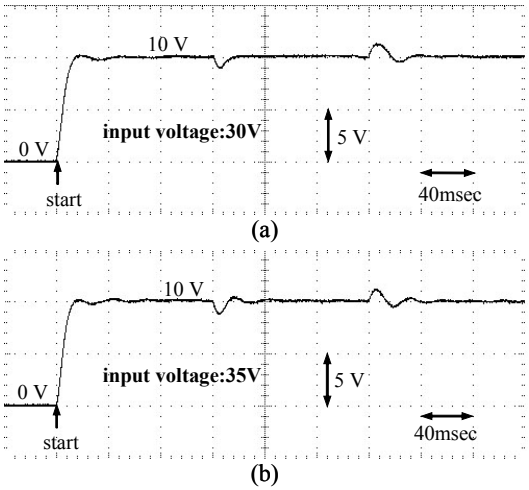

Fig. 6 Experimental results of the fuzzy controller.

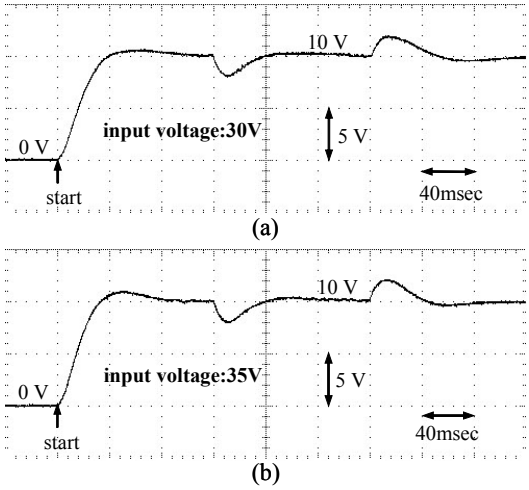

Fig. 7 Experimental results of the adaptive neural fuzzy controller.

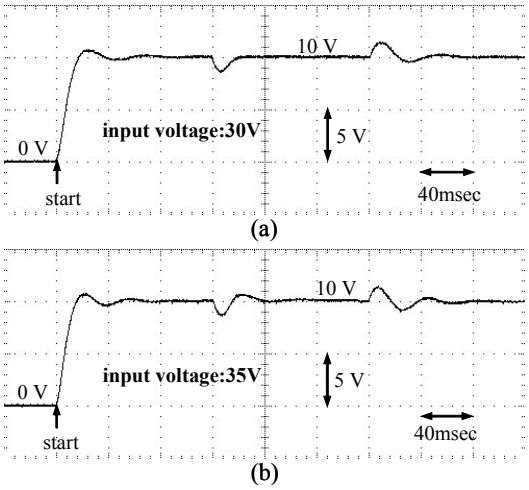

Fig. 8 Experimental results of the adaptive neuro-wavelet controller.

4.1 Comparison of different control methods

First, a PI controller proposed in [2] is applied to control the forward DC-DC converter. The PI controller is given as

$$\delta d_{pi} = 6e + 0.001\dot{e}\,. \tag{36}$$

In (36), it is a PD control for δd and it can be regarded as a PI control for d. The experimental results of the PI controller for Case 1 and Case 2 are shown in Figs. 5(a) and 5(b), respectively. From the experimental results, the PI controller can achieve a regulator performance; however, the controller gains are determined through a lot of trials. The design of the PI controller is based on the completely understanding of the model and through some time-consuming design procedures; moreover, if the model is changed, the PI controller should be re-designed.

Next, the fuzzy controller [5] is applied to control the forward DC-DC power converter again. The fuzzy control rules are given in the following form

Rule i: IF e is F_e^i and $\dot{e}$ is $F_{\dot{e}}^i$, THEN δd_{fc} is ρ_i (37)

where $\rho_i, i = 1,2,\cdots,n$ is the singleton control action and F_e^i and $F_{\dot{e}}^i$ are the labels of the fuzzy sets. The defuzzification of the controller output is accomplished by the method of center-of-gravity

$$\delta d_{fc} = \frac{\sum_i w_i \times \rho_i}{\sum_i w_i} \tag{38}$$

where w_i is the firing weight of the i-th rule. The experimental results of the fuzzy controller for Case 1 and Case 2 are shown in Figs. 6(a) and 6(b), respectively. Though the fuzzy controller can achieve a favorable regulator performance; however, the fuzzy rules base is constructed through a lot of trial-and-error to ensure proper behavior in the operating conditions.

Then, the adaptive neural fuzzy control [6] is applied to control the forward DC-DC power converter again. The experimental results of the adaptive neural fuzzy control for Case 1 and Case 2 are shown in Figs. 7(a) and 7(b), respectively. From the experimental results, the robust tracking performance of the adaptive neural fuzzy control system is obvious under the occurrence of the load resistance variations after training. However, the initial transient response is not good. Finally, the adaptive neuro-wavelet control [7] is applied to control the forward DC-DC power converter again. The experimental results of the adaptive neuro-wavelet control for Case 1 and Case 2 are shown in Figs. 8(a) and 8(b), respectively. It is seen that the

adaptive neuro-wavelet control can achieve a good regulation performance; however, the initial transient response is not good.

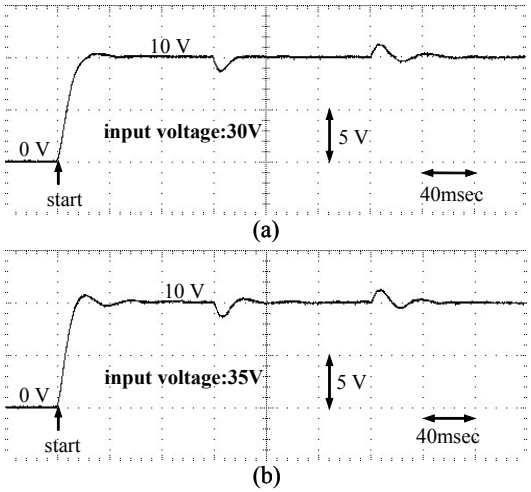

Fig. 9 Experimental results of the AWNC system.

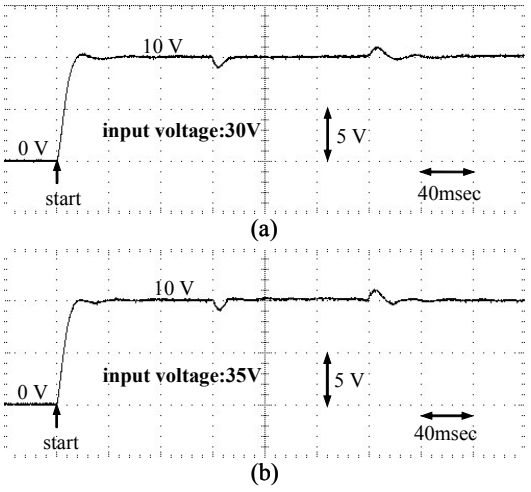

Fig. 10 Experimental results of the trained AWNC system.

4.2 AWNC system with a PI-type adaptation law

The proposed AWNC system is applied to control the forward DC-DC power converter again. The parameters are selected as $k = 200$, $\eta_I = 10$, $\eta_P = 1$, $\eta_2 = \eta_3 = 0.1$, and $\eta_4 = 0.01$. The choice of these parameters is through some trails. The experimental results of the proposed AWNC system for Case 1 and Case 2 are shown in Figs. 9(a) and 9(b), respectively. It is seen the proposed AWNC system can achieve better regulation performance than the adaptive neuro-wavelet control in [7]. The proposed AWNC system can automatically tune the controller parameters without the accurate system model. Further, the trained AWNC control system is applied to the forward DC-DC power converter again. The experimental results of the trained AWNC system for Case 1 and Case 2 are shown in Figs. 10(a) and 10(b), respectively. The experimental results show the trained AWNC system has been a feasible approach which can achieve favorable robust characteristics for the forward DC-DC power converter with the input voltage and load resistance variations.

Table 1 Comparison of controller characteristics.

Controller	Controller Parameters	Load Variation Regulation Ability	Convergence Speed
PI control [2]	trial and error	middle	fast
fuzzy control [5]	trial and error	middle	fast
adaptive neural fuzzy control [6]	on-line learning	excellent	slow
adaptive neuro-wavelet control [7]	on-line learning	excellent	middle
AWNC	on-line learning	excellent	fast

5 Conclusion

This paper has successfully implemented an adaptive wavelet neural control (AWNC) scheme for a forward DC-DC power converter using a field-programmable gate array chip. The experimental results have verified the effectiveness of the proposed AWNC method even under different input voltages and load resistance variations since the on-line learning scheme is applied. Additionally, the comparison of controller characteristics among the PI controller [2], the fuzzy controller [5], the adaptive neural fuzzy control [6], the adaptive neuro-wavelet control [7] and the proposed AWNC system is made in Table 1. It shows that the proposed AWNC system is more suitable for the forward DC-DC power converter.

The major contributions of this paper are: 1) the successful development of the AWNC system based on an FPGA approach. The parameter learning algorithm of the AWNC system is design based on the Lyapunov stability theorem to guarantee the system stability; 2) the successful applications of the AWNC system to control the forward DC-DC power converter. And, the proposed AWNC methodology can be easily extended to other DC-DC power converters; 3) the derived PI-type parameter tuning mechanism can speed up the convergence of the tracking error.

Acknowledgments

The authors are grateful to the reviewers for their valuable comments. The authors appreciate the partial financial support from the National Science Council of Republic of China under grant NSC 99-2628-E-216-002.

References:

[1] A.I. Pressman, *Switching Power Supply Design*, New York: McGraw-Hill, 1998.

[2] J. Alvarez-Ramirez, I. Cervantes, G. Espinosa-Perez, P. Maya, A. Morales, A stable design of PI control for DC-DC converters with an RHS zero, *IEEE Trans Circuits Syst. I*, Vol.48, No.1, 2001, pp. 103-106.

[3] M. Castilla, L.G. Vicuna, J.M. Guerrero, J. Matas, J. Miret, Sliding-mode control of quantum series-parallel resonant converters via input-output linearization, *IEEE Trans Ind. Electron.*, Vol.52, No.2, 2005, pp. 566-575.

[4] S.C. Tan, Y.M. Lai, K. Tse, K.H. Cheung, Adaptive feedforward and feedback control schemes for sliding mode controlled power converters, *IEEE Trans Power Electron.*, Vol.21, No.1, 2006, pp. 182-192.

[5] D. He, R.M. Nelms, Fuzzy logic average current-mode control for DC-DC converters using an inexpensive 8-bit microcontroller, *IEEE Trans Ind. Appl.*, Vol.41, No.6, 2005, pp. 1531-1538.

[6] A. Rubaai, A.R. Ofoli, L. Burge, M. Garuba, Hardware implementation of an adaptive network-based fuzzy controller for DC-DC converters, *IEEE Trans Ind. Appl.*, Vol.41, No.6, 2005, pp. 1557-1565.

[7] C.M. Lin, K.N. Hung, C.F. Hsu, Adaptive neuro-wavelet control for switching power supplies, *IEEE Trans Power Electron.*, Vol.22, No.1, 2007, pp. 87-95.

[8] W. Stefanutti, P. Mattavelli, S. Saggini, M. Ghioni, Autotuning of digitally controlled DC-DC converters based on relay feedback, *IEEE Trans Power Electron.*, Vol.22, No.1, 2007, pp. 199-207.

[9] C.F. Hsu, Design of intelligent power controller for DC-DC converters using CMAC neural network, *Neural Comput. Appl.*, Vol.18, No.1, 2009, pp. 93-103.

[10] W. Al-Hoor, J.A. Abu-Qahouq, L. Huang, W.B. Mikhael, I. Batarseh, Adaptive digital controller and design considerations for a variable switching frequency voltage regulator, *IEEE Trans Power Electron.*, Vol.24, No.11, 2009, pp. 2589-2602.

[11] J.J.E. Slotine, W.P. Li, *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, 1991.

[12] M.A. Duarte-Mermoud, A.M. Suarez, D.F. Bassi, Multivariable predictive control of a pressurized tank using neural networks, *Neural Comput. Appl.*, Vol.15, No.1, 2005, pp. 18-25.

[13] C.M. Lin, Y.F. Peng, Missile guidance law design using adaptive cerebellar model articulation controller, *IEEE Trans Neural Netw.*, Vol.16, No.3, 2005, pp. 636-644.

[14] C.M. Lin and C.H. Chen, Robust fault-tolerant control for biped robot using recurrent cerebellar model articulation controller, *IEEE Trans Syst. Man Cybern. Pt. B*, Vol.37, No.1, 2007, pp. 110-123.

[15] C.F. Hsu, Adaptive recurrent neural network control using a structure adaptation algorithm, *Neural Comput. Appl.*, Vol.18, No.2, 2009, pp. 115-125.

[16] S.A. Billings, H.L. Wei, A new class of wavelet networks for nonlinear system identification, *IEEE Trans Neural Netw.*, Vol.16, No.4, 2005, pp. 862-874.

[17] S. Postalcioglu, Y. Becerikli, Wavelet networks for nonlinear system modeling, *Neural Comput. Appl.*, Vol.16, No.4, 2007, pp. 433-441.

[18] C.K. Lin, Adaptive tracking controller design for robotic systems using Gaussian wavelet networks, *IEE Proc Contr. Theory Appl.*, Vol.149, No.4, 2002, pp. 316-322.

[19] C.D. Sousa, E.M. Hemerly, R.K.H. Galvao, Adaptive control for mobile robot using wavelet networks, *IEEE Trans Syst. Man Cybern. Pt. B*, Vol.32, No.4, 2002, pp. 493-504.

[20] C.F. Hsu, C.M. Lin, T.T. Lee, Wavelet adaptive backstepping control for a class of nonlinear systems, *IEEE Trans Neural Netw.*, Vol.17, No.5, 2006, pp. 1175-1183.

[21] C.F. Hsu, C.M. Chung, C.M. Lin, C.Y. Hsu, Adaptive CMAC neural control of chaotic systems with a PI-type learning algorithm, *Expert Syst. with Appl.*, Vol.36, No.9, 2009, pp. 11836-11843.

[22] [Online] http://www.altera.com/

[23] [Online] http://www.terasic.com.tw/

[24] [Online] http://www.ti.com/