An FPGA-Based Digital Control and Communication Module For Space Power Management and Distribution Systems

Zhan Ping and Zhiqiang Gao
Department of Electrical and Computer Engineering
Cleveland State University
Cleveland, OH 44115

Abstract: A field programmable gate array (FPGA)-based digital control and communication module (DCCM), designed to be the backbone for future space power management (PMAD) systems, is developed and implemented. In this paper, the hardware architecture and logic design of the module are addressed. A basic proportional-integral controller and an advanced linear active disturbance rejection controller are implemented on the FPGA for feasibility and performance tests. The logic implementation of the control algorithms is presented and the hardware test results are shown. The disturbance rejection ability of both control algorithms are evaluated and compared. The communication implementation is discussed. Finally, a multi-converter with communication function testbed is shown to explore the potential benefits of the new control and communication module.

Keywords: FPGA, PMAD, Modular Design, SoC, DC-DC Power Converter, Space Electronics.

I. Introduction

The space power management and distribution (PMAD) system is expected to act as the "electric utility," safely providing regulated power to spacecraft systems and scientific payloads [1]. The PMAD system will continuously perform this function in a space-limited spacecraft for many years, withstanding radiation, vacuum, and temperature changes of the hostile space environment. Therefore, the PMAD system requires reliability, flexibility and compactness. This poses many design challenges, particularly in the design of its core module for control and communication functions.

Custom-design is widely used in traditional space PMAD systems, which offers performance advantage, but, at the same time, makes the system inflexible and difficult to maintain. This motivates researchers to investigate modular designs. Compared with custom-design, modular design uses standard modules repeatedly to meet the variety of mission requirements. The modules are developed independently with the standard interface to other modules, thus reducing the development, assembly, integration, and maintenance costs. In addition, the redundancy can be easily achieved using the standard modules to improve system reliability [1].

An important function of the modern PMAD system is control and communication. Digital control technology has enabled not only the implementation of advanced control functions for each device in PMAD, but also communication and coordination among all system components. The rapid development in digital technologies, both in hardware and software, has been increasingly applied to space power electronics [2-5]. Consequently, the digital control and communication module will be a crucial part in future space power systems. This paper concerns the design and implementation of this module.

Providing regulated power to on-board electronics is a common task in PMAD systems, which is accomplished by DC-DC power converters. Depending on the mission, the power requirements are quite different. The objective of this research is to design and implement a digital control and communication module (DCCM) for DC-DC converters of all sizes. The primary requirements for such applications include voltage regulation, current sharing, and load balancing. Systems with multiple converters also commonly require interleaving, efficiency optimization, and load balancing [2].

The available digital control hardware options include Field Programmable Gate Array (FPGA), Complex Programmable Logic Device (CPLD), Digital Signal Processor (DSP), and Micro Control Unit (MCU). The FPGA is chosen because 1) its circuit can be programmed easily in both hardware and software; 2) its capacity and speed compete with Application-Specific Integrated Circuit (ASIC) chips and allow most functions in the control and communication module to be integrated on a single chip, known as the System On Chip (SoC) design; 3) it is compact and reliable; and 4) radiation-hardened packaging is available, making it attractive for space flights. Considering all of its features, FPGA provides a good platform solution for PMAD applications.

To enhance the control performance and to meet a variety of mission requirements, a novel linear active disturbance rejection control (LADRC) algorithm is implemented in this control module. LADRC has the unique ability to actively reject the disturbances that result from internal dynamic changes and external sources [6].

This paper is organized as follows. The hardware architecture of the DC-DC converter system, including the DC-DC Power Module, hardware design, and logic design of FPGA, is described in Section II. Proportional-Integral (PI) and LADRC control algorithms are introduced, and their implementations in the FPGA are described in Section III. Section IV shows the hardware test results of control algorithm disturbance rejection. The communication implementation and its application in the three-module testbed are discussed in Section V. Finally, concluding remarks are included in Section VI.

II. Hardware Architecture

A modular DC-DC converter system includes a DCCM and a power module, as illustrated in Figure 1. The sensors on the power module measure the output voltage and the output current, which are fed into the control module. In the control module, the sensor signals are first passed through an analog programmable filter and then digitized by the analog to digital converter (ADC). The FPGA-based circuitry executes the control algorithm and generates the PWM signal that is sent to the power module. It also supports the Ethernet and Controller Area Network (CAN) Bus communication networks, which connect the converter to other converters and remote operators. With the communication links, the multiple DC-DC converters can work together in a coordinated manner to improve efficiency and reduce current and voltage ripples. The Ethernet communication makes it possible for remote operators to monitor the health of the system and to perform controller tuning, if necessary.

2.1 The Power Module

A DC-DC power module, pictured in Figure 2, is used to test the DCCM. This Buck converter produces a regulated 12 Volts with a maximum current of 24 A from a 28 V DC source. The frequency of the PWM signal is 40 KHz. The common topology of the Buck converter is shown in Figure 3. The power converter consists of one active switch (MOSFET), one passive switch (Diode), one LC low pass filter at the output, two voltage sensors for input voltage and output voltage, and two current sensors for the input and output current. The output voltage value regulation is accomplished by adjusting the duty cycle of the MOSFET switch, also known as the pulse width modulation (PWM) method.

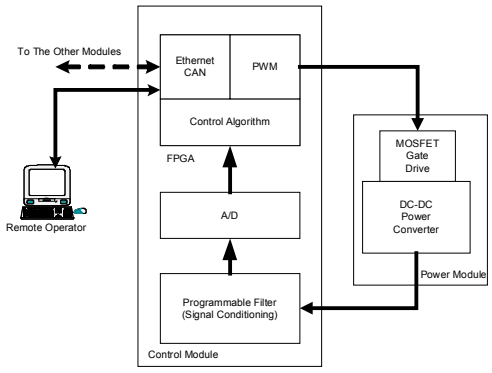

Figure 1 A Modular DC-DC Converter System

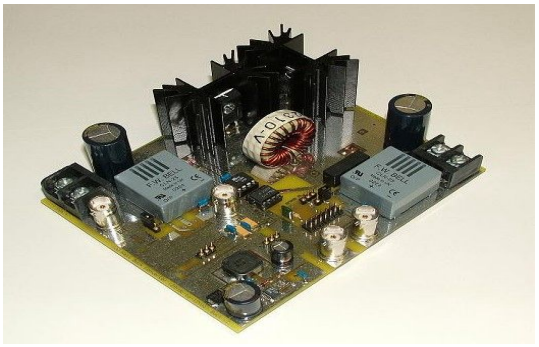

Figure 2 A DC-DC Power Converter Module

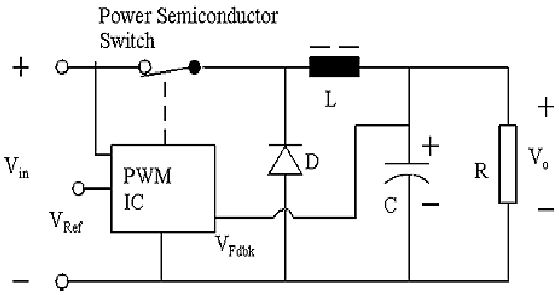

Figure 3 Schematic of the DC-DC Power Converter

Figure 4 FPGA-Based DCCM

2.2 The Control and Communication Module

The DCCM is pictured in Figure 4, which mainly consists of four ADC, four programmable filters, SRAM, Flash memory and one FPGA. DCCM supports the Ethernet and CAN communication. The hardware design and the FPGA logic implementation of DCCM will be addressed in this section.

2.2.1 Hardware Design of DCCM

Based on performance and flexibility considerations, a hardware architecture combining FPGA and the programmable analog filter was selected. All of the functions are eventually implemented onto chips in hardware and software design. In order to make the system more compact and reliable, the smallest number of chips possible is used.

The Lattice ispPAC80, a single chip programmable analog filter, is used for signal conditioning of the control module. It has the programmable gain range from 0 dB to 20 dB, and can be implemented in multiple filter types: Elliptical, Chebyshev, Bessel, Butterworth, Linear Phase, Gaussian and Legendre. In this design, the ispPAC80 replaces traditional analog components, such as Op Amps, eliminating the need for external resistors and capacitors. It provides the good performance, reliability, compactness and reconfiguration ability. These features give the system great flexibility for control adaptation and fault accommodation, as well as broad application possibilities.

The digital functions in the Altera Apex20K200E FPGA include control algorithms, PWM generator, communication interface, memory, and NIOS soft CPU core. The control algorithm can be implemented in hardware or software. Some advanced control algorithms, such as LADRC, are time-critical; the hardware implementation gives the control algorithm a precise calculation period and allows its timing to be unaffected by the other system functions. The PWM generator is also implemented in hardware, and its resolution and period match the control algorithm. The soft core CPU in the FPGA performs the communication and data processing. It also provides a platform for built-in intelligence functions such as fault diagnoses and accommodation in future PMAD systems.

2.2.2 Logic Implementation:

In order to satisfy the requirements of the PMAD system, a new architecture of the digital design in FPGA is used. As shown in Figure 5, two Nios CPUs, one master and the other slave, are implemented in FPGA. The master CPU is in charge of the housekeeping functions, including CAN bus and Ethernet communication functions and SRAM, FLASH memory control. The slave CPU takes care of the data processing and numerical algorithm computation. During operation, the master CPU supplies the parameters from the remote operator to the slave processor and the slave returns the status.

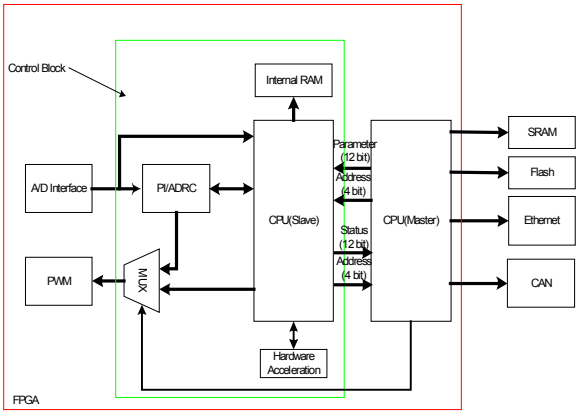

Figure 5 FPGA Logic Design Block Diagram

The control algorithms can be executed in either the slave CPU or the customized hardware logic in FPGA. Each method has its own advantages: the customized hardware logic is fast, compact and portable, while the slave CPU is more flexible and easy to implement (using C language). For health monitoring purposes, the intermediate values in the control algorithm computation need to be stored and sent to the remote operators. This can be easily achieved using the slave CPU. Therefore, the user chooses between speed and flexibility in determining the proper implementation method. For test purposes, control algorithms are realized in both customized hardware logic and slave CPU. The results are fed into a multiplexer and the master CPU controls the multiplexer to select one of the results and send it to the PWM generator. This implementation serves two purposes: 1) compare the speed and flexibility of both implementations; and 2) create a hardware redundancy for error check and fault tolerance.

III. FPGA Implementation of Control Algorithms

The choice of control algorithms is important for the performance of the DC-DC converter. In this section, the traditional PI control method and a more recent LADRC control algorithm are introduced and then implemented.

3.1 Control Algorithms

The PI controller is a commonly used method in industry. It has the form of

$$u(t) = k_p e(t) + k_I \int e(t)dt \qquad (1)$$

where *e(t)* is the difference between the desired and the actual output, k_p and k_I are design parameters. The transfer function of the controller is:

$$G_c(s) = \frac{(k_P s + k_I)}{s} \tag{2}$$

The PI controller is simple and easy to use, but its performance is quite limited. Much research effort has been devoted to advanced control strategies for DC-DC converters [4,5,7-10]. The LADRC method in [5,6] is adopted here based on the experimental results. The LADRC control law is given as follows: The linear extended state observer (LESO) is defined as:

$$\begin{aligned} \dot{z}(t) &= Az(t) + Bu(t) + L(y(t) - \hat{y}(t)) \\ \hat{y}(t) &= Cz(t) \end{aligned} \tag{3}$$

where

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, B = \begin{bmatrix} 0 \\ b_0 \\ 0 \end{bmatrix}, L = \begin{bmatrix} 3\omega_0 \\ 3\omega_0^2 \\ \omega_0^3 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix},$$

Here ω_0 is the bandwidth of the observer. The control law is

$$u = \frac{\omega_c^2 (r - z_1) - 2\omega_c z_2 - z_3}{b_0} \tag{4}$$

where r is the set point and ω_c is the control bandwidth. The LADRC has three design parameters, b_0, ω_0 and ω_c, which can be easily tuned [6].

3.2 Implementations

The control algorithms can be easily implemented in soft core CPU using C language, this implementation is already widely used. Here, the discussion is focused on the control algorithm implementation in custom logic using VHDL. The implementation of PI and LADRC will be discussed in this section. The hardware test results are presented and a comparison is made.

3.2.1 PI Implementation

The PI control algorithm is implemented in VHDL. The discrete version of (1) is:

$$u(k) = u(k-1) + (k_p + \frac{k_i T}{2})e(k) + (-k_p + \frac{k_i T}{2})e(k-1) \tag{5}$$

where *u(k)* is current control signal and *u(k-1)* is the previous one. In the VHDL code, the two multiplications in (5) are implemented in parallel. The constants of $k_i T/2$ are pre-calculated to save the logic usage of the FPGA and to increase the speed.

3.2.2 LADRC Implementation

The discrete equation of LADRC is written as:

$$\begin{aligned} z_1(k) &= z_1(k-1) + T(z_2(k-1) + 3\omega_0(y - z_1(k-1))) \\ z_2(k) &= z_2(k-1) + T(z_3(k-1) + 3\omega_0^2(y - z_1(k-1)) + b_0 u_0) \\ z_3(k) &= z_3(k-1) + T\omega_0^3(y - z_1(k-1)) \\ u(k) &= (k_p(r - z_1(k)) - k_d z_2(k) - z_3(k))/b_0 \end{aligned} \tag{6}$$

where ω_c is the bandwidth of the controller, and ω_0 is the bandwidth of the observer, *T* is the A/D sampling time, b_0 is the initial acceleration of y in step response.

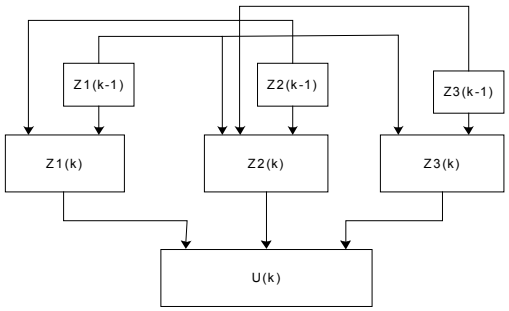

Figure 6 LADRC Implementation Block Diagram

To reduce the computation time, the fixed point format and combinational logic architecture is used for the VHDL programming. The discrete LADRC equation, illustrated in Figure 6, contains four parts: the calculations of $z_1(k)$, $z_2(k)$, $z_3(k)$, and $u(k)$. The first three can be done in parallel, since they only need the precious values of the variables, which were buffered by the flip-flops in VHDL. Obviously, $z_2(k)$ has the longest process time among the three. The $u(k)$ requires the current value of the $z_1(k)$, $z_2(k)$,and $z_3(k)$. So $u(k)$ can only be computed after the values of $z_1(k)$, $z_2(k)$,and $z_3(k)$ are obtained. The total calculation time is found to be 500ns in simulation.

Note that the controller parameters, k_p, k_d, ω_0, and b_0, are subject to change as the controller is tuned during the initiation stage. The parameters in equation (6), i.e., k_p, k_d, $3\omega_0$, $3\omega_0^2$, $3\omega_0^3$, and $1/b_0$, are functions of ω_c, ω_0 and b_0, respectively. Instead of computing them by customized logic, they are calculated in NIOS and are then fed to the FPGA circuit corresponding to (6). This greatly reduces the computation time and FPGA resources needed.

IV. Hardware Test Results

The disturbance rejection characteristic is used as a criterion to evaluate the control performance for the existing controller (PI) and the new one (LADRC). Here, the disturbance rejection is tested by changing the load current between the minimum current 3A and the maximum current 24A. The output voltage of the DC-DC power module was observed using an oscilloscope. The test results for load increase and

decrease are shown in Figures 7 and 8, respectively. Clearly, LADRC outperforms PI in both cases.

Note that in comparing the disturbance rejection in a DC-DC power converter, there are two quantities of interests: the maximum voltage deviation and the recovery time. Both of them and the improvement percentages are listed in Table I. The test results demonstrate significant improvement obtained by implementing LADRC in the DCCM.

Furthermore, there is a distinct advantage in implementing such a controller. Note that the existing PI is usually "tuned" by the trial and error method as it is implemented for a particular application. This is both time-consuming and unreliable. The LADRC controller, on the other hand, is tuned systematically. The ω_0 and ω_c are the observer and controller bandwidth, respectively, which are determined directly based on the design specifications. The only information needed for the LADRC design is the plant parameter b_0, which can be quite easily obtained from the converter. See [5,6] for details.

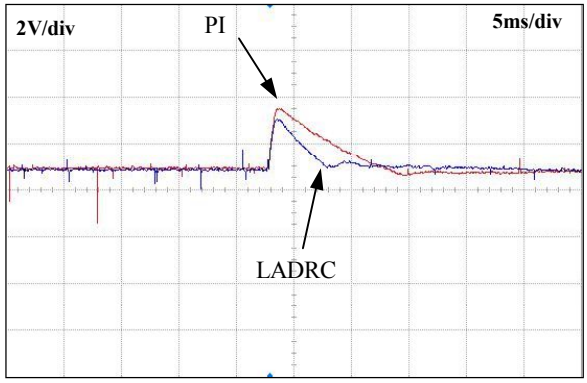

Figure 7 Load Step Down (3A-24A) Disturbance Rejection

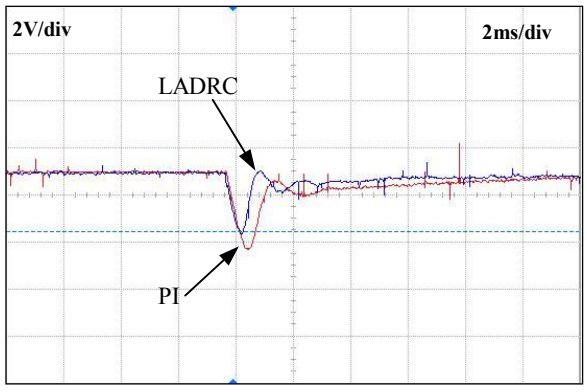

Figure 8 Load Step Up (24A-3A) Disturbance Rejection

Table I Comparison of Load Disturbance Rejection

Deviation of Voltage (V)	PI	LADRC	Improvement
Step-Up	2.6	2.1	19%
Step-Down	2.1	1.8	14%

Recover Time(ms)	PI	LADRC	Improvement
Step-Up	6.7	4.5	33%
Step-Down	12	9	25%

V. The DCCM Communication Implementation

The CAN and Ethernet daughter card allows the DCCM to support CAN and Ethernet communication functions. Each of these communication functions has its own advantages. The combined CAN and Ethernet architecture in DCCM is introduced in this section, as well as a discussion of CAN bus implementation in a three-module test bed.

5.1 CAN Bus and Ethernet

CAN is a serial bus system, which has two main services: the sending of a message (data frame transmission) and the requesting of a message (remote transmission request, RTR). CAN bus has some advanced features for real-time communication: 1) support of multi-master hierarchy; 2) support of broadcast communication; and 3) sophisticated error detecting mechanisms and re-transmission of faulty messages. The CAN bus has 1 Mbits/s data transmission rate over short distances (40 m) and low-speed (5 kbits/s) transmission rate at lengths of up to 10,000 m.

Ethernet provides high-speed data transmission over long distance. In DCCM, it supports 10/100Mbps Ethernet communication. Compared with CAN bus, however, the delay in the Ethernet data transmission is nondeterministic, since Ethernet uses the random back-off mechanism. The embedded systems, such as digitally controlled PMAD systems, often require deterministic real-time data communication, which makes the CAN bus attractive. Therefore, a combined CAN-Ethernet communication scheme is proposed, as shown below.

5.2The DCCM Communication Architecture

The architecture of the communication implementation for the DCCM is shown in Figure 9. A group of DCCM modules was connected both with CAN and Ethernet. These two communication methods have the different responsibilities. The low volume and time critical, data, such as the synchronization data, current and voltage, and number of modules on line, are transmitted by CAN bus. With the CAN bus real-time communication link, the load sharing, dynamic

resource allocation, and phase interleaving become feasible. On the other hand, the high volume data that is not time critical, such as health information, tuning data, and temperature, is transmitted by Ethernet. With the Ethernet communication, high level functions such as system health monitoring and controller tuning can be performed remotely. Overall, this CAN bus and Ethernet combination provides not only good real time performance but also high-speed data transmission over long distance. It provides the foundation for the development of future intelligent PMAD systems.

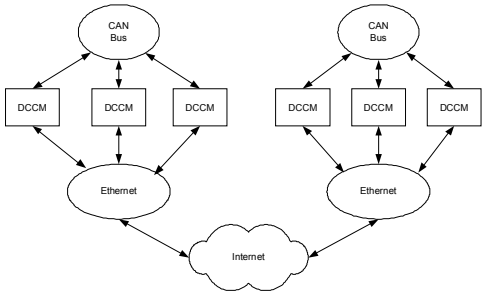

Figure 9 Communication Architecture

5.3 Three-Module Test Bed

To take advantage of the DCCM's communication functionality, a test bed consisting of three modular converters is set up to test the distributed control as well as the CAN Bus communication functions, as pictured in Figure 10. The three power converters are connected in parallel format, each controlled by its own DCCM. The converters communicate via CAN Bus to each other in a master-less, peer-to-peer, manner. Three objectives are established and evaluated for this setup: 1) share load current among the converters in a predetermined manner; 2) improve efficiency by automatically turning off one or more converters when the load is light; and 3) reduce current and voltage ripple by using interleaving. All three objectives are achieved and the readers are referred to [2] for details.

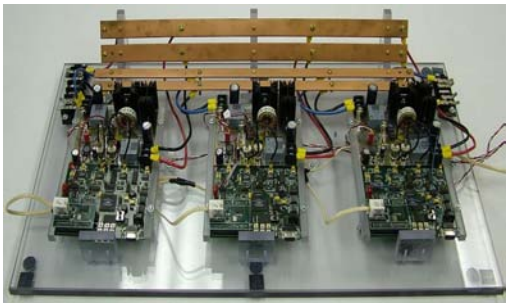

Figure 10 A Three-Module Test Bench

VI. Concluding Remarks

An FPGA-based digital control and communication module is designed, implemented, and tested for space PMAD systems. The hardware setup and the FPGA logic design are presented. Two different control algorithms are implemented in FPGA and tested for disturbance rejection properties. CAN and Ethernet based communication networks are implemented to facilitate deterministic data transmission in local networks and high speed data transmissions to remote operators, respectively.

Acknowledgements: This research is sponsored by NASA under grant NCC3-931. The authors would like to thank their colleagues at the Advanced Engineering Research Lab at CSU, including M. Gray, I. Jurcic, G. Tollis, J. Zeller, and B. Sun.

References

[1] Robert M. Button, "Intelligent Systems for Power Management and Distribution," 2000 SAE Power Systems Conference, San Diego, CA, October 31, 2000.

[2] Michael S. Gray, Zhiqiang Gao, and Robert M. Button, "Distributed, Master-less Control of Modular DC-DC Converters," International Energy Conversion Engineering Conference, Providence, RI, August 2004.

[3] R.R. Boudreaux, R.M. Nelms, and John Y. Hung, "Digital Control of DC-DC Converters: Microcontroller Implementation Issues," Combined Proceedings of HFP Power Conversion & Advanced Power Electronics Technology, Power Systems World '96. September 1996, pp. 168-180.

[4] John Sustersic, John Zeller, Zhiqiang Gao, and Robert Button, "Design and Implementation of a Digital Controller for DC-DC Power Converters," SAE Power System Conference, October 2000.

[5] Bosheng Sun and Zhiqiang Gao "A DSP-Based Active Disturbance Rejection Control Design for a 1KW H-Bridge DC-DC Power Converter," To be published in the *IEEE Transactions on Industrial Electronics*.

[6] Zhiqiang Gao, "Scaling and Bandwidth-Parameterization Based Controller Tuning," *Proc. of the 2003 American Control Conference*, June 2003, pp. 4989-4996.

[7] Marcelo C. Gonzalez, "Modeling of a Full-Bridge DC-DC Power Converter," Master's Thesis, Cleveland State University, Cleveland, OH, December 2001.

[8] Minshao Zhu, "A Nonlinear Digital Control Solution for a DC-DC Power Converter," Master's Thesis, Cleveland State University, Cleveland, OH, December 2002.

[9] Tarun Gupta, R.R. Boudreaux, R.M. Nelms, and John Y. Hung, "Implementation of a Fuzzy Controller for DC-DC Converters Using an Inexpensive 8-b Microcontroller," *IEEE Transactions on Industrial Electronics*, Vol. 44, No. 5, October 1997, pp. 661-668.

[10] Barry Arbetter and Dragan Maksimovic, "Feedforward Pulse Width Modulators for Switching Power Converters," *IEEE Transactions on Power Electronics*, Vol. 12, No. 2, March 1997.

[11] Gregory W. Donohoe, K. Joseph Hass, Stephen Bruder, and Pen-Shu Yeh, "A Reconfigurable Data Path Processor for Space Applications," NASA Goddard Space Flight Center, *Proc. Military and Aerospace Applications of Programmable Logic Devices 2000*, Laurel, MD, September 24-28, 2000.

[12] Gregory W. Donohoe, and James Lyke, "Adaptive Computing for Space," *Proc. 42nd Midwest Symposium on Circuits and Systems*, Las Cruces, NM, August 8-18, 1999.

[13] Zhiqiang Gao, "From Linear to Nonlinear Control Means: A Practical Progression," *ISA Transactions*, Vol. 41, No. 2, pp. 177-89.

[14] Zhiqiang Gao, Yi Huang and Jingqing Han, "An Alternative Paradigm for Control System Design," *Proc. of the 40th IEEE Conference on Decision and Control*, Orlando, FL, December 2001, pp. 4578-4585.