1-1-2006

Power Electronics Controller Prototyping Tool For Power System Applications

Yong Cheng

POWER ELECTRONICS CONTROLLER PROTOTYPING TOOL FOR POWER SYSTEM APPLICATIONS

By

Yong Cheng

A Thesis
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in Electrical Engineering
in the Department of Electrical and Computer Engineering

Mississippi State, Mississippi

May 2006

POWER ELECTRONICS CONTROLLER PROTOTYPING TOOL FOR POWER SYSTEM APPLICATIONS

By

Yong Cheng

Approved:

Dr. Herbert L. Ginn Assistant Professor of Electrical and Computer Engineering (Director of Thesis)	Dr. Noel Schulz Associate Professor of Electrical and Computer Engineering (Committee Member)
Dr. Randolph F. Follett Assistant Professor of Electrical and Computer Engineering (Committee Member)	Dr. Nicholas Younan Professor of Electrical and Computer Engineering (Graduate Coordinator)
Dr. Roger King Associate Dean of College of Engineering	

Name: Yong Cheng

Date of Degree: May 13th, 2006

Institution: Mississippi State University

Major Field: Electrical and Computer Engineering

Major Professor: Herbert L. Ginn

Title of Study: POWER ELECTRONICS CONTROLLER PROTOTYPING TOOL FOR POWER SYSTEM APPLICATIONS

Pages in Study: 68

Candidate for Degree of Master of Science

Many types of devices based on power converters have been proposed and studied for utility applications. In recent years most of the control systems for these converters have been digital. Unfortunately, such digital controllers, which are often based on a digital signal processor (DSP), are difficult to model in simulation. Thus, hardware prototypes are usually required. This thesis presents a tool for fast prototyping that helps overcome these difficulties. Namely, a hardware-in-the-loop simulation is provided for the digital controller in order to evaluate control algorithms without the voltage source converter and power system. The controller in the loop design methodology is described and the division between the real-time power system model and the hardware controller with an interface is shown. Also, the modulation type, integration time step selection and synchronization between the controller and the real-time system simulation are discussed. The hardware configuration for the real-time simulator and the software implementation of the simulator is discussed.

In this thesis an example application of a shunt active compensator following this formal procedure is presented. The active compensator prototyping was first developed in MATLAB/Simulink. Then, following a formal design procedure, the power system was modeled in a digital simulator and the controller was implemented in a digital controller board. Finally, a hardware-in-the-loop test was carried out to validate the performance of the hardware controller for the active compensator. Although the tools and methods presented here are aimed at shunt connected current controller application, they may be generalized for use in the development of any digitally controlled power electronic converter.

DEDICATION

I would like to dedicate this research to my beloved family.

ACKNOWLEDGMENTS

I would like to express my deepest appreciation to my academic advisor, Dr. Herbert L. Ginn III, who has given direction and support throughout my graduate program and for his constant help and support both technically and emotionally during my studies.

I would also like to thank my committee members, Dr. Noel Schulz and Dr. Randolph F. Follett, for their invaluable suggestions and guidance during my research and course work.

I sincerely appreciate Mississippi State University and ONR for their financial support during my thesis work. The Department of Electrical and Computer Engineering also provided necessary support and made this research possible.

I will always remember all of my lab members here at Mississippi State University for their help and support during my studies and research. I quite enjoyed my time with them at Mississippi State University.

Special thanks are given to my wife, my lovely baby, my parents and my parents-in-law for their encouragement, understanding and support.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

CHAPTER I

INTRODUCTION

Utility power systems have long been viewed as an important application area for power electronics. Many types of devices based on power converters have been proposed and studied for utility applications. The functions of these power electronic devices include such applications as power flow control, system stability and security enhancement, improving efficiency, power quality, and protection. Such devices for power system applications can be grouped as two types of power electronic converter systems based on converter topology: shunt connected current controllers and series connected voltage controllers. Active Compensator (shunt connected), Static Synchronous Compensator (STATCOM), and Mini-HVDC are common shunt connected current controllers for power system applications. Static Series Synchronous Compensator (SSSC), Dynamic Voltage Restorer (DVR) and Active Compensator (series connected) are common series connected voltage controllers for power system applications. Shunt connected current controllers can be grouped by applications where the bidirectional voltage source converter is the basic power electronic module. It is also the basic module for a series voltage controller.

Advances in power conversion technologies are making it easier to build high power converters capable of handling the power level required for power system applications. In recent years, research related to the power electronics building block

(PEBB) concept has attempted to categorize various power electronic converters and divide their function in an effort to standardize their design and construction. Thus PEBBs have been developed to achieve: increased power density, "user friendly" design, and multi-functionality.

Generally, a PEBB can be considered as a circuit topology or a physical block. It is a basic function block and can be used for more complex applications. The AC switch, AC/DC voltage source converter, and DC/DC converter are three basic PEBBs for utility system controllers. Through different combinations of these blocks, it is possible to implement needed applications for utility power systems [12].

In conjunction with the development of PEBBs, digital controls, integrated with higher frequency and more robust power circuits, enable modular power systems with lower size, weight, and cost, while also increasing performance. The development of a digital control system in power application has resulted from the microelectronics revolution. Beginning with the use of discrete analog and digital components, the progression has been to microprocessor, microcontrollers and digital signal processors (DSP). Now, it is possible to implement complex control algorithm on-line for power electronics at a reasonable cost. It is possible to use different control algorithms including proportional integral (PI), fuzzy logic, deadbeat, and neural nets for improving the dynamic and steady state performance of a system controller.

During power electronic converter design, various control strategies must be evaluated based on performance criteria before application in a real system. Due to the long periods of time often required for hardware prototyping the evaluation stage is

usually conducted in simulation. Unfortunately, the digital controller, which is often based on a DSP, is difficult to model accurately in simulation. The designer can not guarantee that the digital controller will perform as indicated in the simulation. Therefore, a controller-in-the-loop prototyping tool is the topic of this thesis, as it allows the developer to evaluate the performance of a DSP based digital controller without the need to complete the entire hardware prototype.

In order to efficiently design power electronic devices for power applications, various design tools and procedures are employed. A formal procedure could help transfer the design from the model to the final hardware implementation. First, the system controller can be modeled with the power system in software for prototyping. A purely software environment helps the designer focus on the algorithm implementation and analysis with ideal power equipment. Then, the control subsystem can be exported to a DSP, and the power system can be modeled in a real-time system simulation. With data acquisition facilities at both sides, a hardware-in-the-loop environment can be established. When satisfactory performance is achieved through the hardware-in-the-loop test, the controller is ready to be inserted into the hardware of the real plant for the final test. Thus, the formal procedure guides the designer to develop the system controller in an efficient and economical way [10].

This thesis presents a design procedure for fast prototyping that helps overcome the difficulties. Namely, a hardware-in-the-loop simulation is provided for the digital controller in order to evaluate control algorithms without the voltage source converter and power system. In this thesis, an example application of a shunt active compensator

following the formal procedure is presented. The shunt active compensator is one common type of shunt connected current controller. Although the tools and methods presented here are aimed at shunt connected current controller applications, they may be generalized for use in the development of any digitally controlled power electronic converter. The active compensator prototyping is developed in MATLAB/Simulink. Then, following a formal design procedure, the power system is modeled in a digital simulator, and the controller is implemented in a digital controller board. Finally, a hardware-in-the-loop test is performed to validate the performance of the hardware controller for the active compensator.

The structure of the thesis is as follows. Chapter II describes the controller in the loop design methodology. A typical system and interface are introduced. The division between the real-time power system model and hardware controller with interface is shown. Also, the modulation type, integration time step selection and synchronization between the controller and the real-time system simulation are discussed. Chapter III presents the real-time system simulator design for the hardware-in-the-loop test. The hardware configuration for the real-time system simulator and the software implementation of the simulator are discussed. Chapter IV describes a shunt active compensator test system. The chapter gives a review of active compensators and describes the implementation of the active compensator used for the test. Chapter V presents the test results of a shunt active compensator. Test results include the results in the MATLAB simulation, hardware-in-the-loop test, and the results of a physical hardware test. Finally, Chapter VI concludes the thesis and proposes future work.

CHAPTER II

CONTROLLER IN THE LOOP DESIGN METHODOLOGY

This chapter describes the controller-in-the-loop design methodology. A typical system, and interfaces are introduced, and the division between a real-time system model and a hardware controller with interface is shown. Also, the modulation type, integration time step selection, and synchronization between the controller and the real-time system simulation are discussed in this chapter.

2.1 Typical System and Interface

Based on topology, power electronic converters used in power systems can be divided into two categories [2][3][4][11]: shunt connected current controllers and series connected voltage controllers. Shunt connected current controllers are shunt connected to a power system with a voltage-fed PWM converter structure. Series connected voltage controllers are connected in series to a power system with a current-fed PWM converter structure. This thesis focuses on a shunt connected current controller and expects that the result will be extendable to series connected voltage controllers. An Active Compensator (shunt connected), a Static Synchronous Compensator (STATCOM), and a Mini-HVDC are some common applications of shunt connected current controllers for power systems. The commonality can be analyzed among various power electronics based controllers in utility power systems. The current controllers can be observed as having the same

fundamental structure. The bidirectional voltage source converter is the basic power electronic module of the shunt connected current controllers. A generalized workflow can represent either a power electronic controller connected in a shunt, or a series configuration, depending on the connection with the system. The generalized workflow of a shunt-connected controller is shown in Figure 2.1.

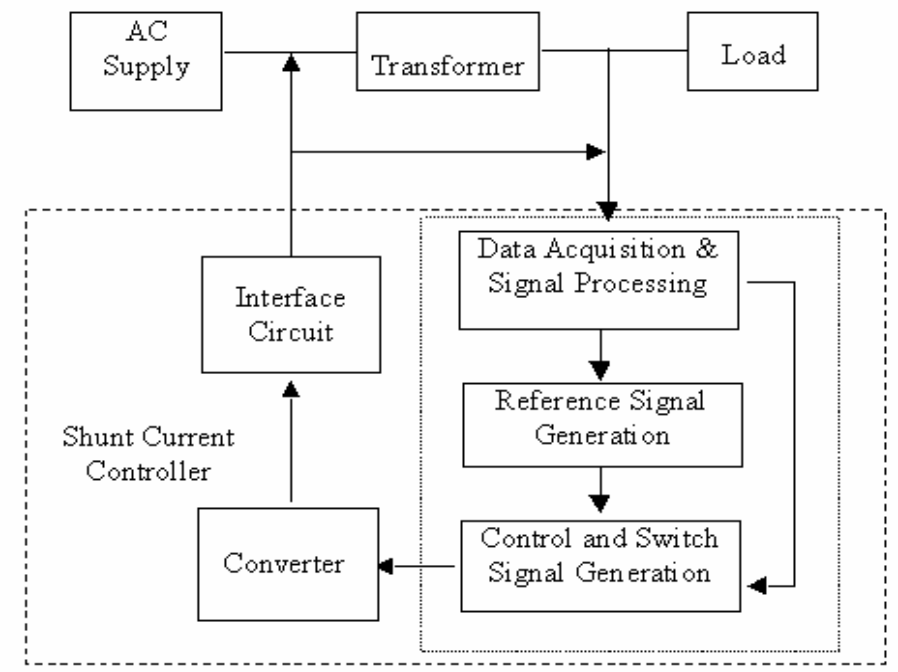


Figure 2.1 Generalized Workflow of a Shunt Current Controller

Usually, the shunt current controller consists of an interface circuit, converter, and the control part. The control part is the heart of the power electronic controller and can be divided into three subsystems. The first subsystem performs data acquisition and signal processing, and the control reference is derived from the measured voltage and current signals. The second subsystem performs reference signal generation; compensating

commands in terms of current or voltage levels are derived based on application goals. The third subsystem is control and switch signal generation. The gating signals for the solid-state devices of the power electronic controller are generated and sent to the converter.

The converter is controlled by the switching signals. Thus, the shunt current controller can generate components in the current according to system states in time. The effectiveness of any shunt current controller relies on the selection of the configuration of the controller and the control strategy. For different applications, configurations with various control strategies have been proposed. The effectiveness of control strategies relies on the method used to obtain the compensating current reference, on the closed loop control strategy applied, and the modulation technique.

In recent years, digital signal processors (DSP) have been widely applied to implement control algorithms for power electronics systems. It is possible to use different control algorithms for improving the dynamic and steady state performance of a shunt current controller.

2.2 Division Between Real Time (RT) Model and Hardware Controller

A real-time system simulator is a tool that will allow the control algorithms of a power electronic controller to be tested without the construction of the associated power electronics and system components. It is very useful for the testing of scenarios where the actual hardware may not be available.

In order to accurately mimic the system response to the digital controller, the system simulation must run on a real-time operating system. A real-time system ensures

that responses will be deterministic, that is, the system calculation and operation will occur within a given time. With non-real-time systems, there is no way to ensure that the response will occur within any given time period, and calculations may finish later or earlier than expected, i.e., hard determinism is required.

In order to perform a hardware-in-the-loop test, the control subsystem can be exported to the DSP, and the power system can be modeled in a real-time system simulation. With channels of Analog-to-Digital Converters (ADC), a DSP can process the measurements from current and/or voltage transducers. With the Pulse Width Modulation (PWM) unit, the DSP generates PWM signals and sends them to the converter. So, the power system with an converter and a data acquisition part is modeled in the Real-time system simulator. Signal Processing (A/D), Reference Signal Generation, and Control and Switch Signal Generation can be performed by a DSP. The interface of a DSP based shunt current controller is shown in Figure 2.2.

In order to model the power electronic converter and simplified section of the power system, several functions of the real-time system simulator have been defined. Generally, they can be divided into three parts as shown in Figure 2.3. “Signal Read” reads the control signals, which could be PWM signals. “System Calculation” determines the response of the power system according to the control signals received. “System Measurement Write” outputs system measurements needed by the DSP.

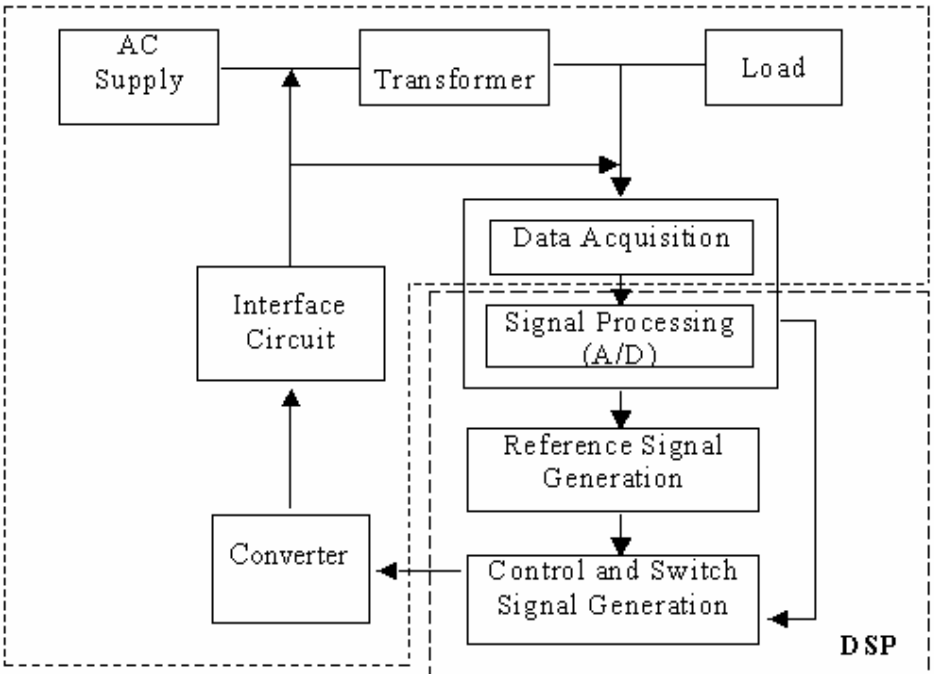


Figure 2.2 Interface of DSP Based Shunt Current Controller

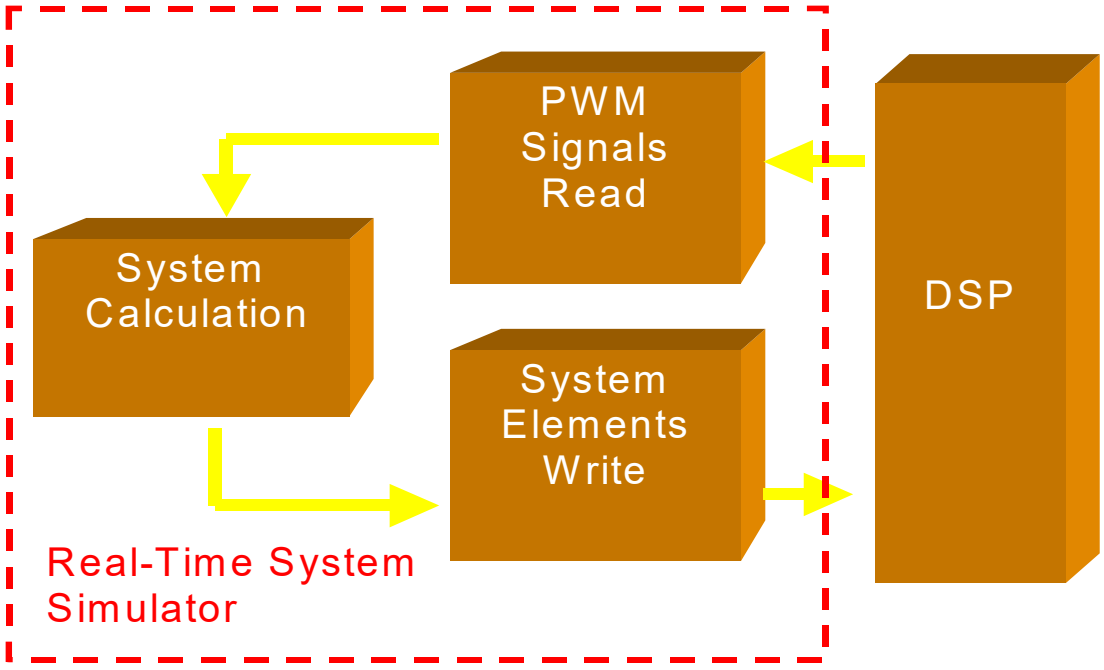


Figure 2.3 Functions and Interface of the Real-time System Simulator

For a power electronic controller, the DSP must obtain system measurements at the start of each switching period and send out the transistor gate signals for the next period. The system simulator must read the PWM pulse in each switching period and send out power system measurement signals at the end of the switching period.

2.3 Assumption of Modulation Type as Space Vector Modulation and Integration Time Step Selection

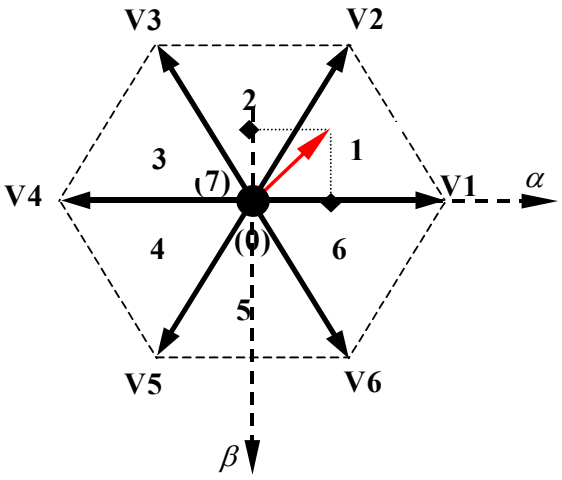
2.3.1 Space Vector Modulation (SVM)

Many modulation techniques are able to produce PWM. Carrier-based Pulse Width Modulation and Space Vector Modulation are two popular techniques for PWM control of three phase converters. In the past, Carrier-based Pulse Width Modulation methods were widely used in most applications [14][15][16][17]. However, since the Carrier-based Pulse Width Modulation is not suitable for the digital control, with the development of microprocessors, Space Vector Modulation has become one of the most important PWM methods. Space Vector Modulation uses the space vector concept to compute the duty cycle of the switches.

In this thesis, the voltage space vector strategy is utilized and detailed analyses are presented in [14][15][16][17]. The sequencing strategy established makes symmetrical use of the states 0 and 7 as shown in Figure 2.4. For example, in sextant 1, the state sequence is 0-1-2-7-2-1-0, and the duration of each state is:

$$t_a = \frac{T}{2}(U_\alpha - \frac{1}{\sqrt{3}}U_\beta)$$
$$t_b = \frac{T}{2}\frac{2}{\sqrt{3}}U_\beta$$
$$t_0 = \frac{T}{2} - t_a - t_b \qquad (2.1)$$

Where T is the switching period and U_α, U_β are the α, β components of the input.



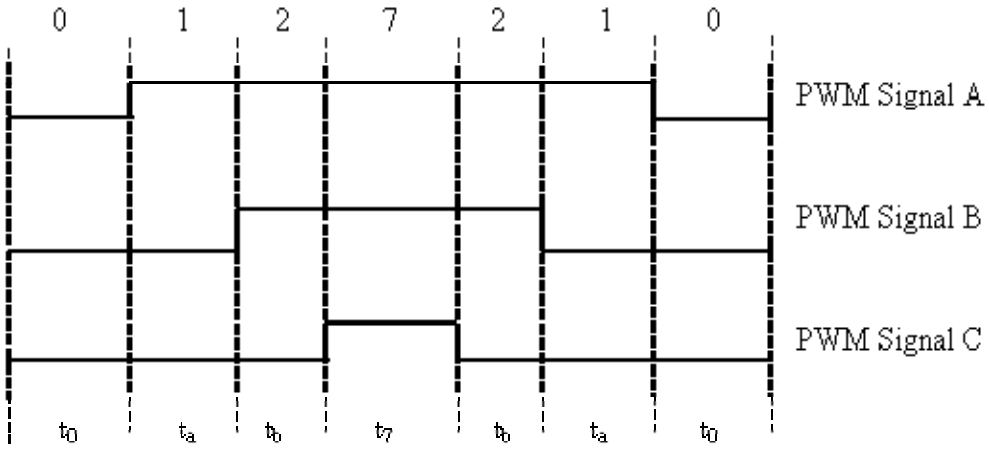


Figure 2.4 Illustration of the Vector PWM a) Shows an Instantaneous Reference Vector in Sextant 1, and b) Shows the Resultant Switching Function

2.3.2 Time Step Selection

After the system model is developed in the real time system simulator, an appropriate integration time-step is needed to solve the system mathematical model. The time-step should be selected to ensure the system solution's accuracy and also meet the time constraints. Small time slicing will slow down the simulation while large time slicing may overlook an important point.

There are many ways to select the time step for the mathematical model. The period can be sliced into hundreds of even slots, or divide according to the states of the three-converter leg's PWM signals. As the duty ratio of the PWM signals varies, to catch the state change of the PWM signal using even slicing, a much smaller time step is needed resulting in longer computation time. Thus, slicing the switch period according to the PWM signal is preferred.

As shown in Figure 2.5, the three PWM signals show that the switch states will change seven times in one cycle. To get accurate simulation results and spend less calculation time, in this test, one switching period is sliced into seven slots. After obtaining ta, tb and tc by "PWM Signal Read", the switch status sequence (0,0,0), (1,0,0), (1,1,0), (1,1,1), (1,1,0), (1,0,0) and (0,0,0) can be deduced in a switching period. Also, the time of each status can be calculated as t1, t2, t3, t4, t5, t6, and t7 respectively.

Using t1, t2, t3, t4, t5, t6, and t7 as the time step for each switch state, the output voltage at the converter terminals can be calculated.

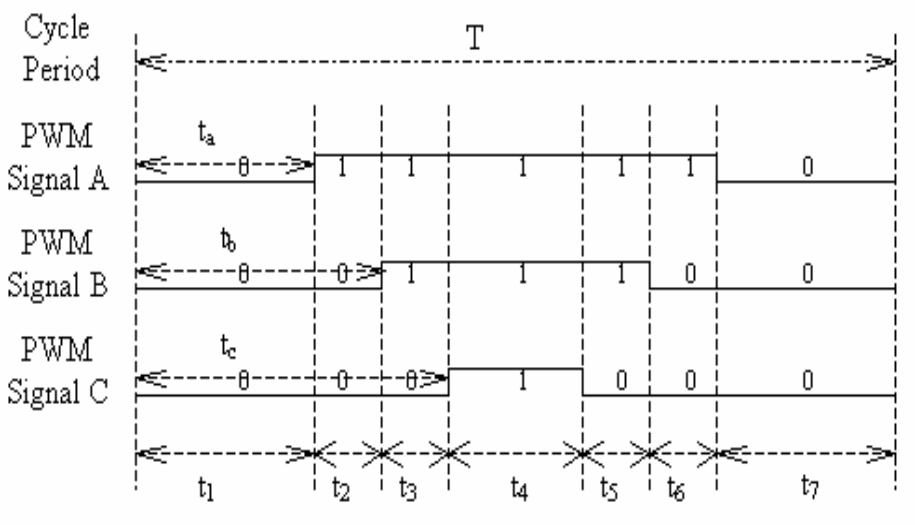


Figure 2.5 Time Step Slicing for PWM Signal

2.3.3 Synchronization of DSP and Real-time System

Precise control of timing and coordination of the DSP and real-time system simulator is needed to ensure that the DSP does not attempt to read the output of the real-time system before it is ready. Synchronization of the DSP and real-time system can be implemented by the addition of a trigger signal.

Since a power electronic controller is a device that usually has a high switching frequency, it puts strict time constraints on the real-time system simulator. For central aligned PWM, a trigger signal is introduced to synchronize the DSP with the system simulation as shown in Figure 2.6.

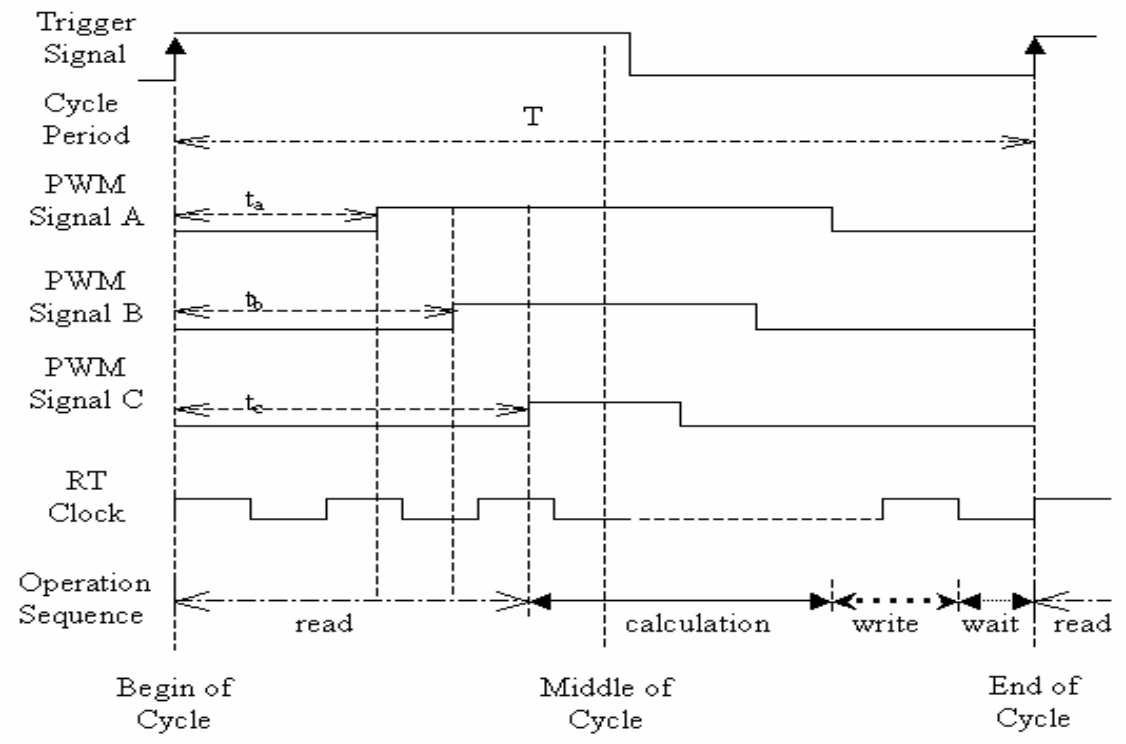


Figure 2.6 Configuration of Power Electronic Controller Design

The trigger signal and PWM pulses are produced by the DSP. The rising edge of the trigger signal is used as a synchronization signal to identify the start of a switching period. When the rising edge of the trigger signal is detected, “Signal Read” blocks begin to count with 20MHz time base until the rising edge of the PWM signals is detected. Thus, ta, tb, and tc for three lower switches of the converter phase legs can be calculated. Then, the power system response to the PWM signals can be calculated and sent back to the DSP. These three functions will occur in sequence and remain coordinated with the DSP I/O. The system outputs are generated before the cycle ends. Then, the real time system simulator waits until the next rising edge of the trigger signal. In this way, one switching period is marked by the rising edge between consecutive trigger signals.

In this chapter, the division between a real-time system model and a hardware controller with interface has been described. According to the interface, the modulation type, integration time step selection, and synchronization between the controller and the real-time system simulation are discussed. Thus hardware-in-the-loop test can be set up based on the determined interface.

CHAPTER III

REAL-TIME SYSTEM SIMULATOR DESIGN

This chapter introduces the real-time system simulator in the Hardware-in-the Loop (HIL) test. In this chapter, the hardware configuration for the real-time system simulator and the software implementation of the simulator will be discussed.

3.1 Real-Time System

A real-time system consists of software and hardware components. In the implementation described in this thesis, the software components include software implemented using the National Instruments LabVIEW platform. The software runs on the LabVIEW real-time operating system called the RT Engine. The hardware components of this real-time system include a host computer and an RT target.

An RT target refers to RT Series hardware that runs the RT Engine and software created with LabVIEW. The RT Engine is a version of LabVIEW that runs on the RT target. The RT Engine runs the Virtual Instruments (VIs) downloaded to the RT target from the LabVIEW on the host computer.

The RT Engine runs on a real-time operating system (RTOS), which ensures that the LabVIEW execution system and other operating system services adhere to real-time operation. When the LabVIEW program is running on the host computer, the program

will be downloaded to the RT target and run on RT target.

3.2 Hardware for the Real-Time System

To fulfill the functions of the real-time system simulator described in chapter 2, the real-time system simulator is required to have the following properties:

- It should have the ability to read PWM signals with acceptable error.
- It needs high-speed digital inputs and analog outputs to match high switching frequency (up to 20KHZ) of the converter.
- In one switching period, the simulator should finish the signal read, system calculation, and measurements write.

Thus, a real-time operating system is required to simulate the power system and communicate with the DSP, such that the task can be executed precisely according to a specified time schedule. Also, high-speed digital and analog data acquisition devices are required.

Figure 3.1 shows the hardware configuration used in the HIL test. The system simulator will simulate the behavior of the shunt current controller along with the entire supply and load system.

To meet the real-time computations needed for this application, a PXI-8186RT controller from National Instruments is used. A PXI-6602 high-speed digital input device is selected to read the PWM signals, and a PXI-6733 high-speed analog output device is used to output the system quantities that are measured by the current controller. These three devices are connected by a PXI chassis and use the same real-time clock. Figure 3.2

shows the devices used for the real-time system simulator, including PXI-8186RT, PXI-6602, and PXI-6733 from National Instruments.

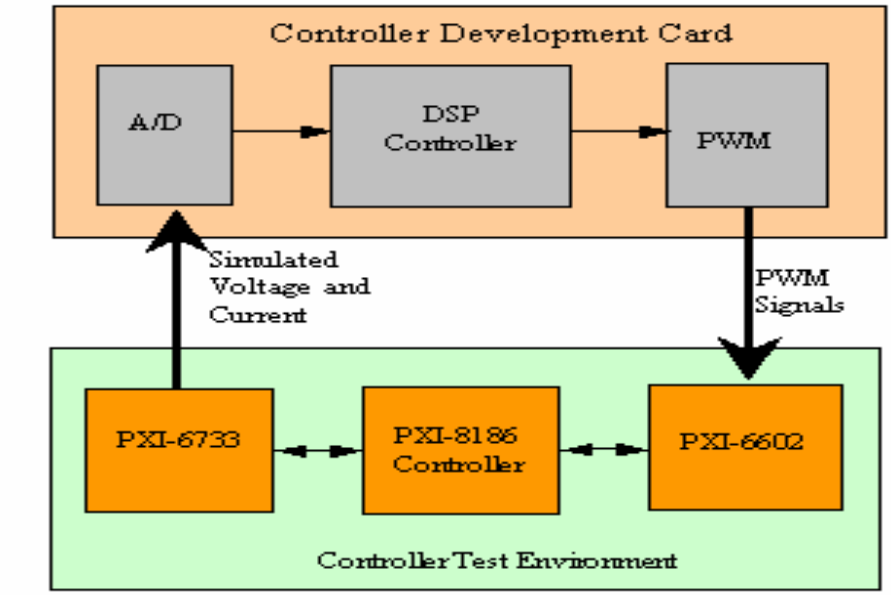


Figure 3.1 Hardware Configuration for Shunt Current Controller HIL

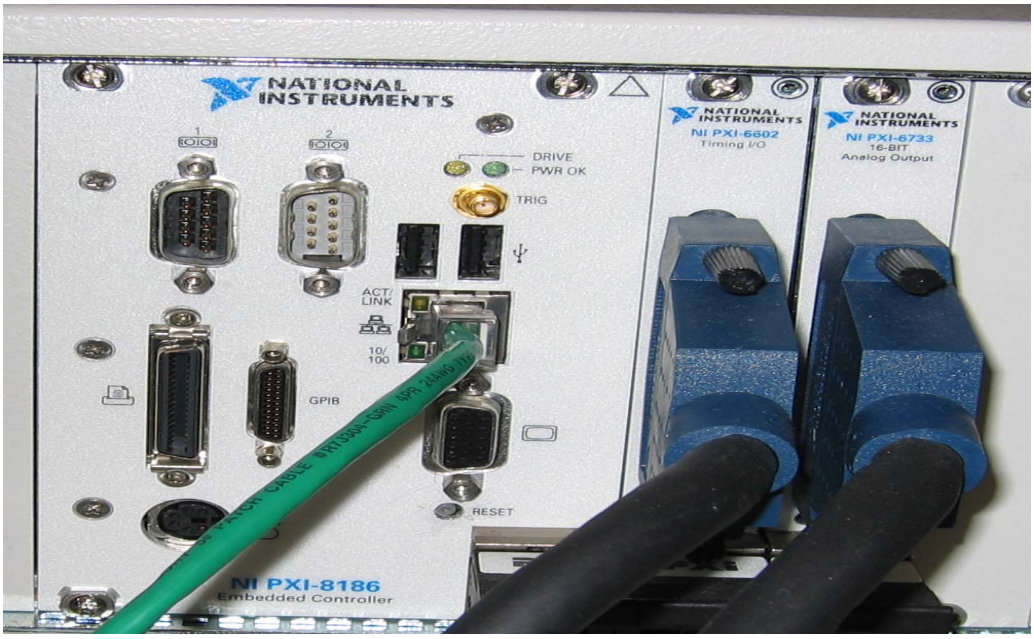


Figure 3.2 NI Devices PXI-8186, PXI-6602 and PXI-6733

3.3 Implementation of real-time system simulator

To validate the function of the current controller, a real-time system simulator is developed to mimic the interaction of a current controller working with a power system. A general workflow of the real-time system simulator is presented in Figure 3.3.

The program flow is divided into two distinct sections. The first section describes the behavior of the power system without the shunt connected current controller, and the second section describes the behavior of the power system with an active shunt connected current controller. When the program begins, it is running in the first section. During that section, the current controller is not placed in operation, and the simulator simulates the power system with a voltage source and load. Thus, the simulator provides the DSP with the values of source voltage and load current through analog output. When the current controller is placed in operation, the program flow proceeds to the second section. During this section, the simulator mimics the power system with a voltage source, load, and current controller. The real-time system simulator will begin by reading the PWM signal, then it will calculate the power system model and output the values of source voltage, load current, current controller injected current, and capacitor voltage through the analog output. Until it reaches the user specified stop time, the simulator will continue running in the loop.

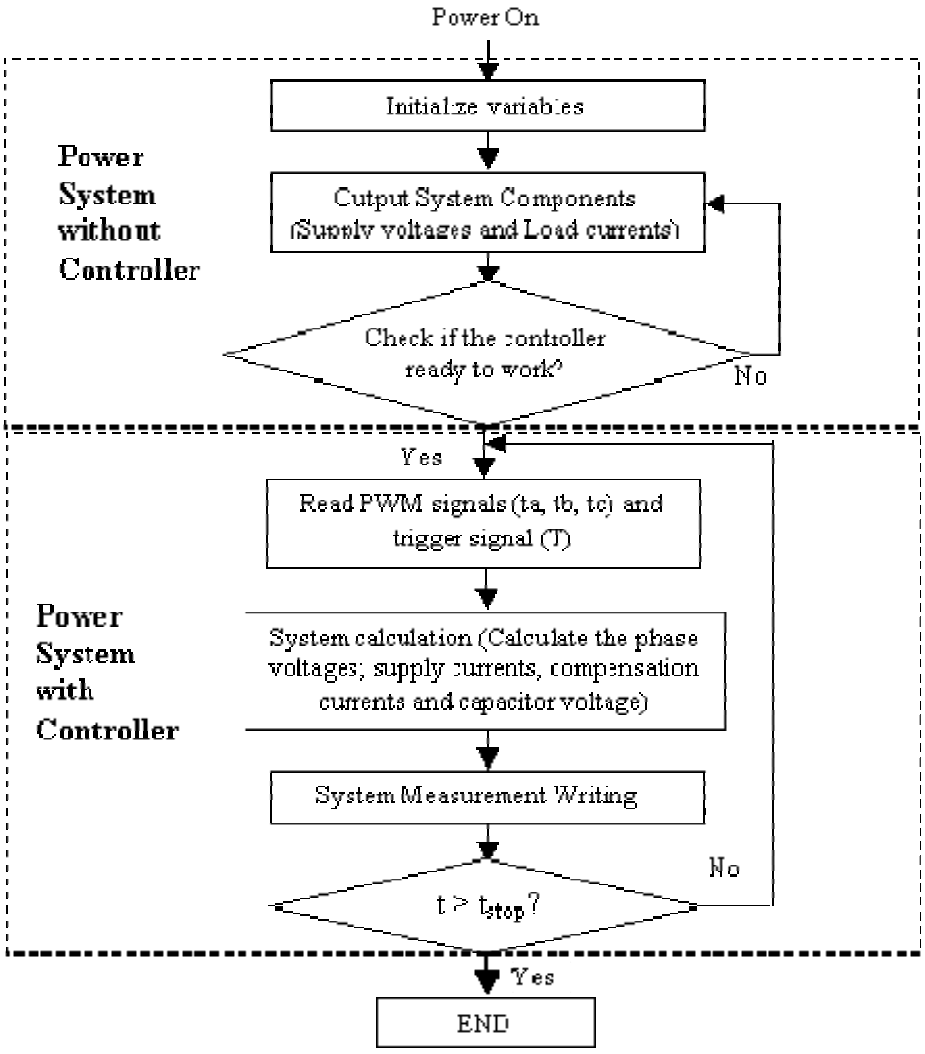


Figure 3.3 Workflow of the Real-time System Simulator

3.3.1 PWM Signal Read

The NI PXI-6602 is used for PWM signal reading. It is a timing and digital I/O module with eight 32-bit counter/timers and 32 lines of TTL/CMOS-compatible digital I/O. Also, it performs a wide variety of counter/timer tasks, including encoder position measurement, event counting, period measurement, pulse width measurement, pulse

generation, pulse train generation, and frequency measurement. The brief product summary of PXI 6602 is available in Appendix A. In this thesis, PXI-6602 counters and digital I/O are configured to realize the function of pulse width measurement of the PWM signal.

Since the signal arrangement as described in Chapter II is implemented, the two-signal edge-separation measurement is used for pulse width measurement. In the two-signal edge-separation measurement, two signals are used to identify the width of the pulse: Trigger Signal (AUX_LINE) and PWM signal (GATE). The counter uses SOURCE, the time base provided by the PXI-8186R, to measure the time between the two signals. Thus, the measurement of PWM signals begins when the rising edge of the trigger signal is detected and ends when the rising edge of the PWM signal is reached. The value of the counter will be latched into the HW Save register. Figure 3.4 shows an example of a two-signal edge-separation measurement.

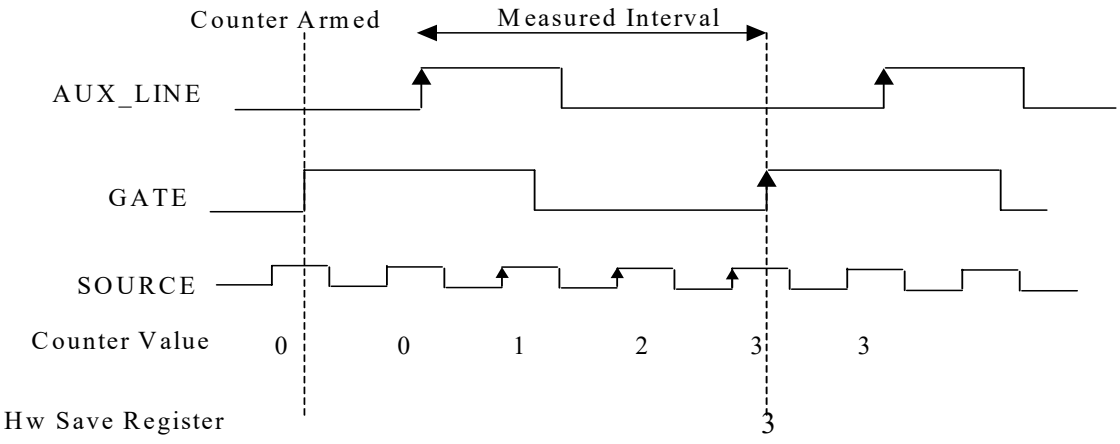


Figure 3.4 Two-Signal Edge-Separation Measurement

When t_1, t_2 and t_3 for three phases are measured in this way, the switch "on" time of the upper leg T_a, T_b, and T_c of the PWM signal can be obtained from:

$$T_a=T-2*t_1;$$

$$T_b=T-2*t_2;$$

$$T_c=T-2*t_3; \quad (3.1)$$

where T is the switching period.

PWM Signal Read Implementation in LabVIEW is shown in Figure 3.5. The node named "CI Two Edge Separation" will fulfill the function of measuring the amount of time between the rising edge of the trigger signal and the rising of the PWM signal. The node "Implicit" will configure the number of samples to acquire for each channel in the task. Then, "DAQmx Start Task" will transmit the task to the running state and begin the measurement. When the task is ready, "DAQmx Read" will read the PWM signal from the channel specified. When reading is finished, this VI stops the task, and if necessary, releases any resources that the task reserved. "DAQmx Clear Task" will clear the task.

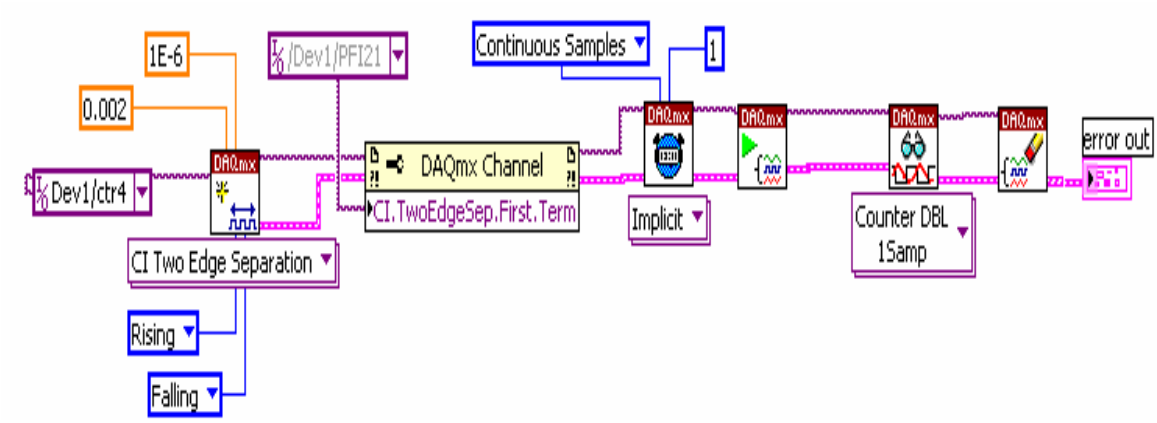


Figure 3.5 PWM Signal Measurement

3.3.2 System Calculation Model Design in LabVIEW

A power system with a shunt current controller is shown in Figure 3.6. The voltage source is assumed to be ideal and balanced, and it is connected with a load. A shunt current controller consists of a three-phase voltage-fed PWM converter using IGBTs, a dc capacitor C_{dc}, three-phase interfacing inductors L_c, and resistors R_c. The current controller is connected in parallel with the load.

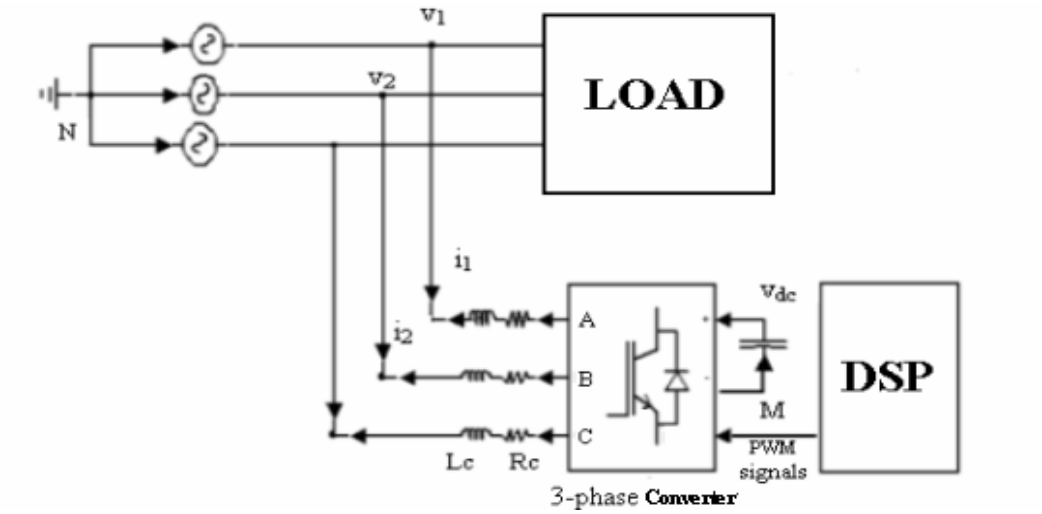


Figure 3.6 Shunt Current Controller Connected to Power System

3.3.2.1 System Model

Using Kirchoff's rule, we can obtain the equations relating the voltage and current at the common connection point of the current controller [1]

$$\begin{aligned} V_1 &= L_c di_1/dt + R_c i_1 + V_{1M} + V_{MN} \\ V_2 &= L_c di_2/dt + R_c i_2 + V_{2M} + V_{MN} \\ V_3 &= L_c di_3/dt + R_c i_3 + V_{3M} + V_{MN} \end{aligned} \tag{3.2}$$

V_1, V_2 and V_3 are the voltages at the connection point, i_1, i_2 and i_3 are the currents fed by the shunt current controller. N is the neutral point of the voltage source and M is the down side of the active filter. As the AC supply voltage is balanced, from Equation (3.2), we can obtain the voltage between point M and point N:

$$V_{MN} = -\frac{1}{3}(V_{1M} + V_{2M} + V_{3M}) = -\frac{1}{3}\sum_{m=1}^{3} V_{mM} \tag{3.3}$$

In the converter, the state of the upper switch and lower switch of each phase leg should be opposite state. So, we take c_k as 1 when the upper switch is closed and the lower switch is open. Otherwise, c_k is 0 when the upper switch is open and the lower switch is closed.

So, when c_k is 1, V_{mM} will be V_{dc} , and when c_k is 0, V_{mM} will be 0. Thus, Equation (3.3) could be written as:

$$V_{MN} = -\frac{1}{3}\sum_{k=1}^{3} c_k V_{dc} \tag{3.4}$$

Also, Equation (3.2) can be modified as:

$$\frac{di_k}{dt} = -\frac{R_c}{L_c} i_k - \frac{1}{L_c}\left[c_k - \frac{1}{3}\sum_{k=1}^{3} c_k\right] V_{dc} + \frac{V_k}{L_c} \tag{3.5}$$

Define the function $d_{nk} = \left[c_k - \frac{1}{3}\sum_{m=1}^{3} c_m\right]$, which is called switching state function.

It could also be presented as:

$$\begin{bmatrix} d_{n1} \\ d_{n2} \\ d_{n3} \end{bmatrix} = \frac{1}{3}\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \tag{3.6}$$

For the converter capacitor, the voltage on the dc side follows the equation:

$$\frac{dV_{dc}}{dt} = \frac{1}{C} i_{dc} = \frac{1}{C} \sum_{m=1}^{3} c_m i_m \tag{3.7}$$

Because the system is balanced and the conversion of c_m to d_{nk} from Equation (3.6), Equation (3.7) can be:

$$\frac{dV_{dc}}{dt} = \frac{1}{C}(2d_{n1} + d_{n2})i_1 + \frac{1}{C}(d_{n1} + 2d_{n2})i_2 \tag{3.8}$$

Thus, the complete model of the current controller in the abc reference frame can be obtained by the application of Equation (3.5) and the Equation (3.8) as:

$$\frac{d}{dt}\begin{bmatrix} i_1 \\ i_2 \\ V_{dc} \end{bmatrix} = \begin{bmatrix} -\frac{R_c}{L_c} & 0 & -\frac{d_{n1}}{L_c} \\ 0 & -\frac{R_c}{L_c} & -\frac{d_{n2}}{L_c} \\ \frac{2d_{n1} + d_{n2}}{C} & \frac{d_{n1} + 2d_{n2}}{C} & 0 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ V_{dc} \end{bmatrix} + \frac{1}{L_c}\begin{bmatrix} V_1 \\ V_2 \\ 0 \end{bmatrix} \tag{3.9}$$

3.3.2.2 Computational Model

A system calculation model can be established based on Equation (3.9). To solve Equation (3.9) with given initial values, there are many numerical methods, such as the Euler method, Trapezoidal method, Runge-Kutta method, Gear's method, and so on. The selection of the numerical method should consider computation time stability and accuracy.

Compared with other methods, the Forward Euler is a single-step, explicit integration method. It is selected for our real-time simulation as long as solver stability (convergence) is guaranteed. Based on using forward Euler to discretize Equation (3.9), we can obtain the difference equation as:

$$\left(\begin{bmatrix} i_1 \\ i_2 \\ V_{dc} \end{bmatrix}_{n+1} - \begin{bmatrix} i_1 \\ i_2 \\ V_{dc} \end{bmatrix}_n\right)/h = \begin{bmatrix} -\frac{R_c}{L_c} & 0 & -\frac{d_{n1}}{L_c} \\ 0 & -\frac{R_c}{L_c} & -\frac{d_{n2}}{L_c} \\ \frac{2d_{n1}+d_{n2}}{C} & \frac{d_{n1}+2d_{n2}}{C} & 0 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ V_{dc} \end{bmatrix}_n + \frac{1}{L_c}\begin{bmatrix} V_1 \\ V_2 \\ 0 \end{bmatrix}_n \quad (3.10)$$

Where h is the time step.

Equation (3.10) can be reorganized as:

$$\begin{bmatrix} i_1 \\ i_2 \\ V_{dc} \end{bmatrix}_{n+1} = \begin{bmatrix} -\frac{R_c}{L_c}h+1 & 0 & -\frac{d_{n1}}{L_c}h \\ 0 & -\frac{R_c}{L_c}h+1 & -\frac{d_{n2}}{L_c}h \\ \frac{2d_{n1}+d_{n2}}{C}h & \frac{d_{n1}+2d_{n2}}{C}h & 1 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ V_{dc} \end{bmatrix}_n + \frac{h}{L_c}\begin{bmatrix} V_1 \\ V_2 \\ 0 \end{bmatrix}_n \quad (3.11)$$

To obtain the stability index of the above equation, the eigenvalue of the iteration matrix in Equation (3.11) needs to be calculated by solving the Equation (3.12).

$$\begin{vmatrix} -\frac{R_c}{L_c}h+1-\lambda & 0 & -\frac{d_{n1}}{L_c}h \\ 0 & -\frac{R_c}{L_c}h+1-\lambda & -\frac{d_{n2}}{L_c}h \\ \frac{2d_{n1}+d_{n2}}{C}h & \frac{d_{n1}+2d_{n2}}{C}h & 1-\lambda \end{vmatrix} = 0 \quad (3.12)$$

Expanding it yields:

$$\begin{aligned} &(-\frac{R_c}{L_c}h+1-\lambda)^2(1-\lambda) - (-\frac{d_{n1}}{L_c}h)(-\frac{R_c}{L_c}h+1-\lambda)(\frac{2d_{n1}+d_{n2}}{C}h) \\ &-(-\frac{d_{n2}}{L_c}h)(\frac{d_{n1}+2d_{n2}}{C}h)(-\frac{R_c}{L_c}h+1-\lambda) = 0 \end{aligned} \quad (3.13)$$

By organizing the Equation (3.13), we can get:

$$(-\frac{R_c}{L_c}h+1-\lambda)((-\frac{R_c}{L_c}h+1-\lambda)(1-\lambda)+\frac{h^2}{L_c*C}(2d_{n1}{}^2+2d_{n1}d_{n2}+2d_{n2}{}^2))=0 \quad (3.14)$$

Thus, eigenvalues can be obtained as:

$$\lambda_1=1-\frac{R_c}{L_c}h; \quad (3.15)$$

$$\lambda_2=1-\frac{\frac{R_c}{L_c}h+\sqrt{(\frac{R_c}{L_c}h)^2-4\frac{h^2}{L_c*C}(2d_{n1}{}^2+2d_{n1}d_{n2}+2d_{n2}{}^2)}}{2}; \quad (3.16)$$

$$\lambda_3=1-\frac{\frac{R_c}{L_c}h-\sqrt{(\frac{R_c}{L_c}h)^2-4\frac{h^2}{L_c*C}(2d_{n1}{}^2+2d_{n1}d_{n2}+2d_{n2}{}^2)}}{2}; \quad (3.17)$$

As $2d_{n1}{}^2+2d_{n1}d_{n2}+2d_{n2}{}^2$ responds to the switch status, when c_1=0,c_2=0,c_3=0 or c_1=1,c_2=1,c_3=1, it equals 0, and at other switch status it equals 2/3. When it equals 0, $\lambda_1=\lambda_2=1-\frac{R_c}{L_c}h, \lambda_3=1$. Equation (3.11) is stable when $|\lambda_1|\le 1$. i.e. $h\le\frac{2L_c}{R_c}$.

When $2d_{n1}{}^2+2d_{n1}d_{n2}+2d_{n2}{}^2$ equals 2/3, If $(\frac{R_c}{L_c}h)^2\ge\frac{8*h^2}{3*L_c*C}$, $\lambda_1,\lambda_2,\lambda_3$ are less than 1. Equation (3.11) is stable if $|\lambda|\le 1$. When

1) $\lambda_1\le 1$, requires $h\le\frac{2L_c}{R_c}$;

2) $\lambda_2\le 1$, requires $h\le\frac{2L_c}{R_c}$;

3) $\lambda_3\le 1$, requires $h\le\frac{4L_c}{R_c}$.

That is, the Equation (3.11) will be stable when $h \le \frac{2L_c}{R_c}$;

If $(\frac{R_c}{L_c}h)^2 < \frac{8*h^2}{3*L_c*C}$, λ_2, λ_3 have the imaginary part. Equation (3.11) is stable when

1) $\lambda_1 \le 1$. So $h \le \frac{2L_c}{R_c}$;

2) $\lambda_2 \le 1$ and $\lambda_3 \le 1$, that is $h \le \frac{3*R_c*C}{2}$

So, when $h \le \frac{2L_c}{R_c}$ and $h \le \frac{3*R_c*C}{2}$ will be stable.

Thus, the stability of the solution is constrained by the time step, the interface parameter L_c, R_c, and the capacitor. When the forward Euler method is selected for modeling, the selected parameters need to be tested in order to ensure that the computation is stable. With the forward Euler method, the truncation error is O (h^2). When the switching period is very small, the error can be neglected.

3.3.2.3 Current Controller Model Implementation

The LabVIEW application was developed with the NI LabVIEW Real-Time Module for Windows and then downloaded to a NI PXI-8186 RT embedded controller via the Ethernet. The brief product summary of PXI-8186 is available in Appendix B. The embedded code executes on a real-time operating system. Thus, all of the powerful, flexible development tools of LabVIEW are used to build reliable real-time solutions.

The Formula Node block in LabVIEW is used to perform mathematical model computation. It is a convenient text-based node used to perform mathematical operation on the block diagram. Thus, the mathematic model of the power system with converter can be realized in LabVIEW. Figure 3.7 shows the computational program for the power system with the current controller in a real-time system simulator.

The program is a two layer "for" loop. The outer "for" loop associates with the number of running iterations, and iteration corresponds to a switch period. The inner "for" loop associates with the time step. When the time step is selected according to PWM signal statuses, there are seven iterations for the "for" loop. The computation process is implemented in the inner "for" loop.

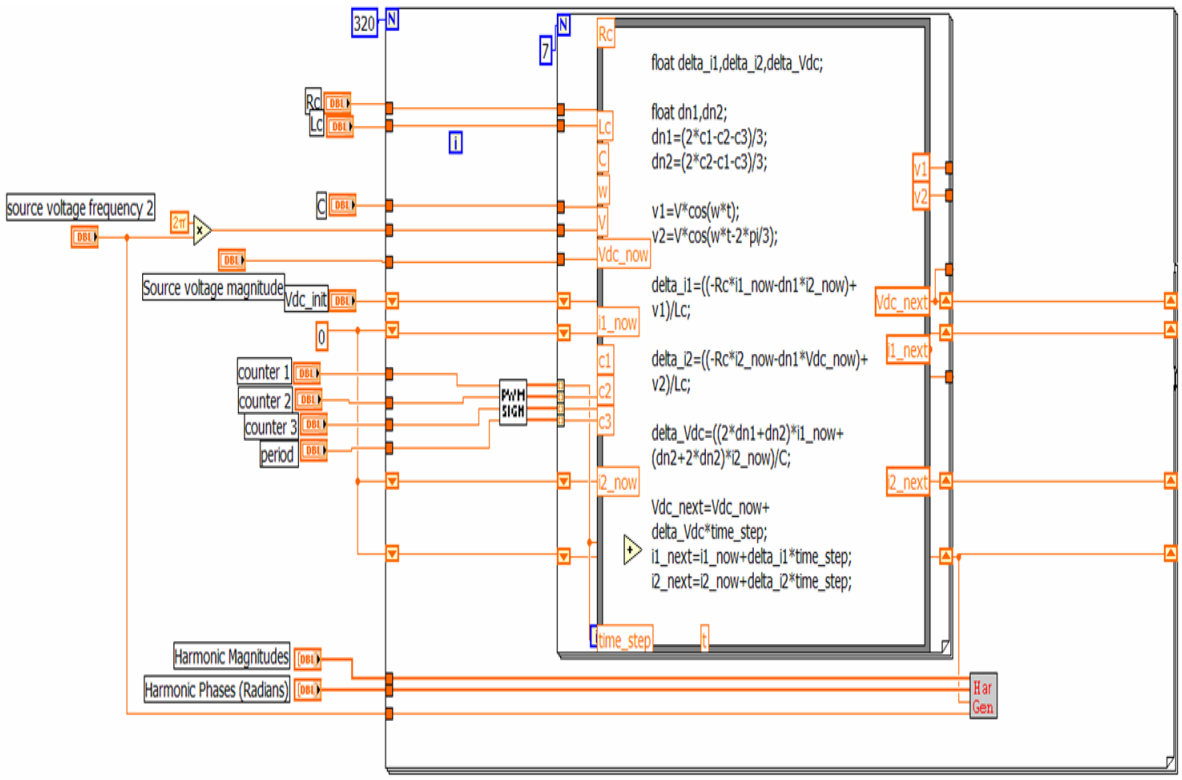


Figure 3.7 Implementation of Power System with Controller in LabVIEW

3.3.3 System Measurements Write

In this thesis, the National Instruments PXI-6733 is selected for system measurement write. It can deliver high-performance, reliable, high-speed analog outputs to meet a wide range of application requirements. This card provides eight analog output channels at 1 MS/s per channel, 16-bit resolution, and digital triggering. The module uses the PXI trigger bus to synchronize with additional data acquisition, motion, and vision products, so that engineers can use this device with LabVIEW Real-Time in order to perform real-time control, including hardware-in-the-loop simulation and rapid-control prototyping. The brief product summary of PXI-6733 is available in Appendix C. In this thesis, PXI-6733 analog output channels will be configured for the output of system measurements to DSP.

The implementation of System Measurements Write in LabVIEW is shown in Figure 3.8. Node "DAQmx Task" is used to create channels to output analog signals. Then, "DAQmx Start Task" will transmit the task to the running state to begin the generation. When the task is established, "DAQmx Write" writes a single sample separately to each channel in a task that contains analog output channels. After the writing is completed, this VI stops the task, and, if necessary, releases any resources the task reserved. "DAQmx Clear Task" will clear the task. To send the system measurements simultaneously, "DAQmx Timing" is used and the property of "SimultaneousAOEnable" is enabled. Thus, channels are enabled to output analog signal simultaneously.

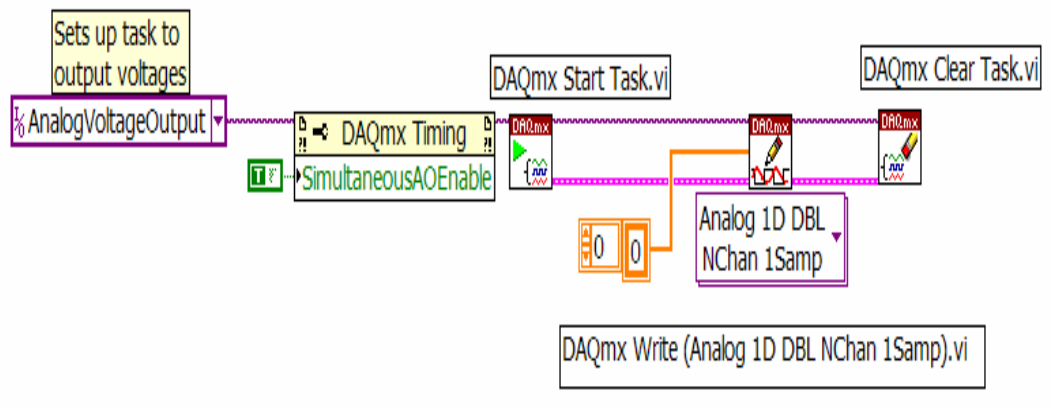


Figure 3.8 System Components Write

In this chapter the real-time system simulator in the Hardware-in-the Loop (HIL) test was introduced. The hardware configuration for the real-time system simulator and the software implementation of the simulator were discussed. Shunt connected active compensator is a common application of shunt connected current controller. As an example, the HIL design of shunt connected active compensator will be described in next chapter.

CHAPTER IV

SHUNT ACTIVE COMPENSATOR TEST SYSTEM

This chapter presents a shunt active compensator test system. The shunt active compensator is one common type of shunt connected current controller. This chapter will give a review of active compensators and describe the implementation of the active compensator used for the test.

4.1 Introduction of Active Compensator

Active compensators for harmonic compensation have become a topic of increasing interest over the past decade. Based on different hardware and control strategies, active compensators can fulfill a combination of functions such as harmonic elimination, unity power factor compensation, unbalance compensation, and neutral current compensation.

Based on converter topology, active compensators can be classified as a series or as a shunt compensator. Figure 4.1 is an example of a shunt active compensator, which is most widely used to eliminate current harmonics, reactive power compensation, and balancing unbalanced currents. Mainly active compensators are used at the load end because nonlinear loads inject current harmonics. An active compensator injects equal compensating currents, opposite in phase, to cancel harmonics or reactive components of

the nonlinear load current at the connecting point.

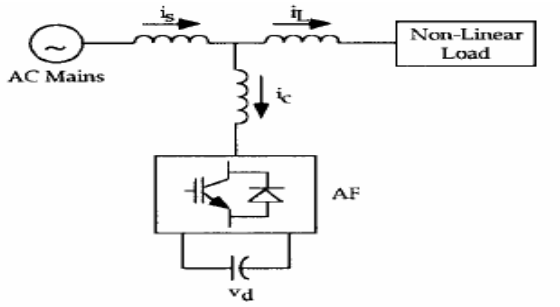


Figure 4.1 Shunt Active Compensator

4.2 Active Compensator Design

When a power system is connected with a nonlinear load, the supply current is distorted. To compensate for the distorted components of the current, a shunt active compensator may be used. In a balanced system, the active compensator acts as a current source connected in parallel with the nonlinear load, and it is controlled to produce the harmonic currents required for the load. In this way, the ac supply needs only to produce the fundamental currents.

In order to compensate the harmonic current, the current control strategy of active compensator is developed. For the test system in this thesis, the control strategy described in [1][5][6][7][8][9] is used. Figure 4.2 shows the global schematic diagram of a shunt active compensator with the nonlinear control. d-q theory is selected for reference generation. PI control will be used in control parameter generation. Space Vector Modulation will be implemented in PWM generation.

Assuming the power system is balanced, dq theory could be used to derive reference with the aim of reducing the control complexity. A nonlinear control technique is used to achieve better performance, while the separation of internal and external loop dynamics is realized. A decoupled current control using PI-type compensators is utilized to let the currents injected by the compensator rapidly track their references. The dc voltage level is regulated using a PI-type nonlinear compensator. Finally, PWM switching signals for the converter are obtained by the space vector modulation. Figure 4.2 illustrates the global schematic diagram of a shunt active compensator with the nonlinear control.

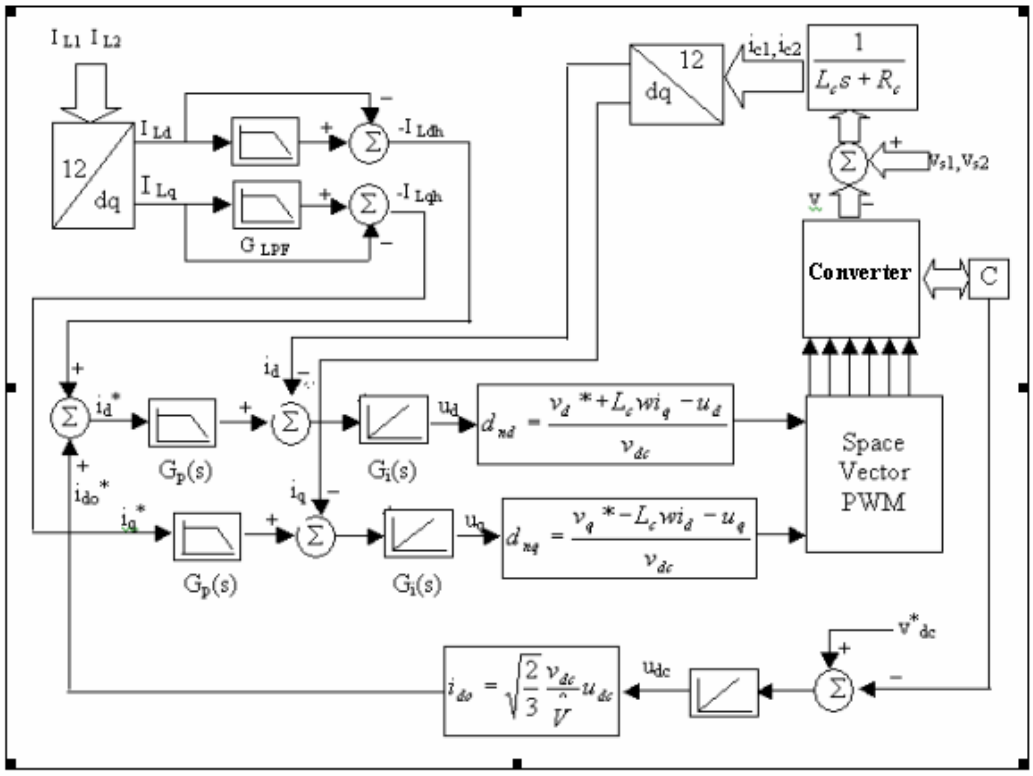


Figure 4.2 Global Schematic Diagram of Shunt Active Compensator with Nonlinear Control

4.2.1 Control Reference Generation

In the synchronous d–q reference frame controllers, voltage and current signals are transformed to a synchronously rotating frame, in which fundamental quantities become dc quantities, and the harmonic compensating commands are extracted. The required compensation harmonic current is obtained by filtering out the fundamental part with a low pass filter.

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = C_{dq}^{12} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} \tag{4.1}$$

The transformation matrix is

$$C_{dq}^{12} = \sqrt{2} \begin{bmatrix} \cos(\theta - \pi/6) & \sin\theta \\ -\sin(\theta - \pi/6) & \cos\theta \end{bmatrix} \tag{4.2}$$

The abc-dq transformation is fulfilled by the function node in Figure 3. As the active compensator is modeled in the stationary abc frame, current reference is derived based on the system measurement i_{L1}, i_{L2}. After d-q transformation, the low pass filter is used to obtain the harmonic. As the fundamental is constant in d-q plan, the cut point of the low pass filter is set at 25 HZ. Thus, the harmonic component is extracted from the system measurement by subtracting the constant from the original current. Figure 4.3 below shows the diagram of reference signal derivation.

As the requirement of the current control, seven system signals will be acquired by the controller: system voltage v_{s1}, v_{s2}, load current i_{L1}, i_{L2}, capacitor voltage v_{dc} and compensation current i_1, i_2.

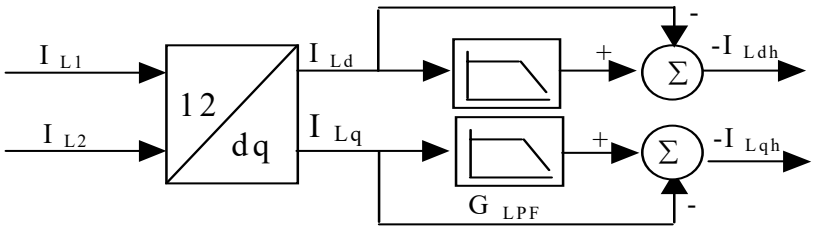


Figure 4.3 Harmonic Current Reference Generation

4.2.2 Control Parameter Generation

A decoupled current control using PI-type compensators are utilized to allow the currents injected by the compensator track rapidly their references. The dc voltage level is regulated using a PI-type nonlinear compensator.

As the d-q transformation, the current reference is extracted in (d, q) plane. So, the coupled dynamics of the currents tracking problem, have been transformed into decoupled dynamics. Hence, the currents i_d and i_q, can be controlled independently by acting upon inputs u_d and u_q, respectively. Furthermore, by using PI-type compensators, a fast dynamic response and zero steady state errors can be achieved.

4.2.2.1 Inner Control Loop

The inner current loop is simplified as: the error, between the reference and the measured inductor current, is amplified by the current controller G_i, which will produce the control voltage. According to the control voltage, PWM produces the switching signal at the switching frequency f_s.

When the harmonic current reference is generated, the error can be obtained by comparing the references with the actual compensation current. In the simulation, the

current controller using PI-type compensators is utilized to allow the currents injected by the active compensator track their reference rapidly.

The transfer function of the PI compensators is given by:

$$G_i(s) = k_p \frac{s + k_i / k_p}{s} \tag{4.3}$$

where:

$K_p = 2\xi\omega_{ni} L_c - R_c$ and $k_i = \omega_{ni}^2 L_c$;

The transient response of the current will be affected by the presence of the zero in transfer function. In order to eliminate the zero in the close loop transfer function, a prefilter is added

$$G_p(s) = \frac{1}{1 + (k_p / k_i)s} \tag{4.4}$$

Below is the figure of the inner control loop of the current i_q:

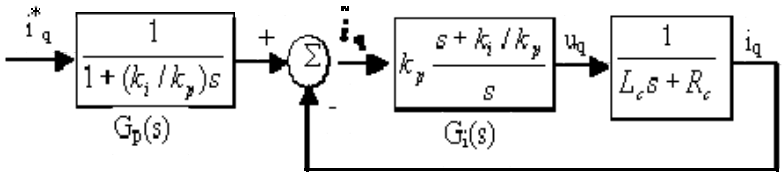


Figure 4.4 The Inner Control Loop of Current i_q

4.2.2.2 Outer Control Loop

The dc capacitor voltage v_{dc} on the dc side of the active compensator is detected and then compared with the dc voltage reference V_{dc}^*. The difference signal between v_{dc}

and V_{dc}^{*} is amplified by a dc voltage controller. Consequently, a small amount of active power is absorbed or released to the dc capacitor so as to regulate v_{dc}.

The transfer function of the PI compensators is given by:

$$G_v(s)=k_1\frac{s+k_2/k_1}{s} \tag{4.5}$$

where:

$k_1=2\xi\omega_{nv}C$ and $k_2=\omega_{nv}^{2}C$

Figure 4.5 shows the outer control loop of the dc voltage.

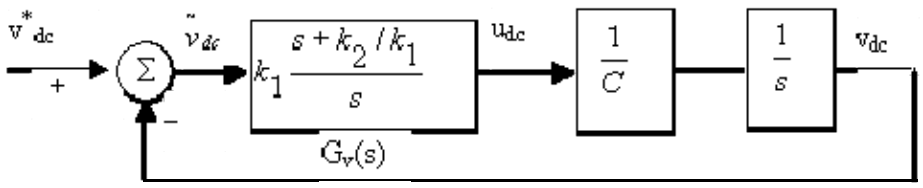


Figure 4.5 The Outer Control Loop of dc Voltage

Thus, with the structure of a fast inner loop, current tracking loop, and a slow outer loop, and a dc voltage regulation loop, a fast dynamic response of harmonic compensation is obtained. As the d-q transformation, the current reference is extracted in the (d, q) plane. So, the coupled dynamics of the currents tracking problem have been transformed into decoupled dynamics. The currents i_d and i_q can be controlled independently by acting upon inputs u_d and u_q, respectively.

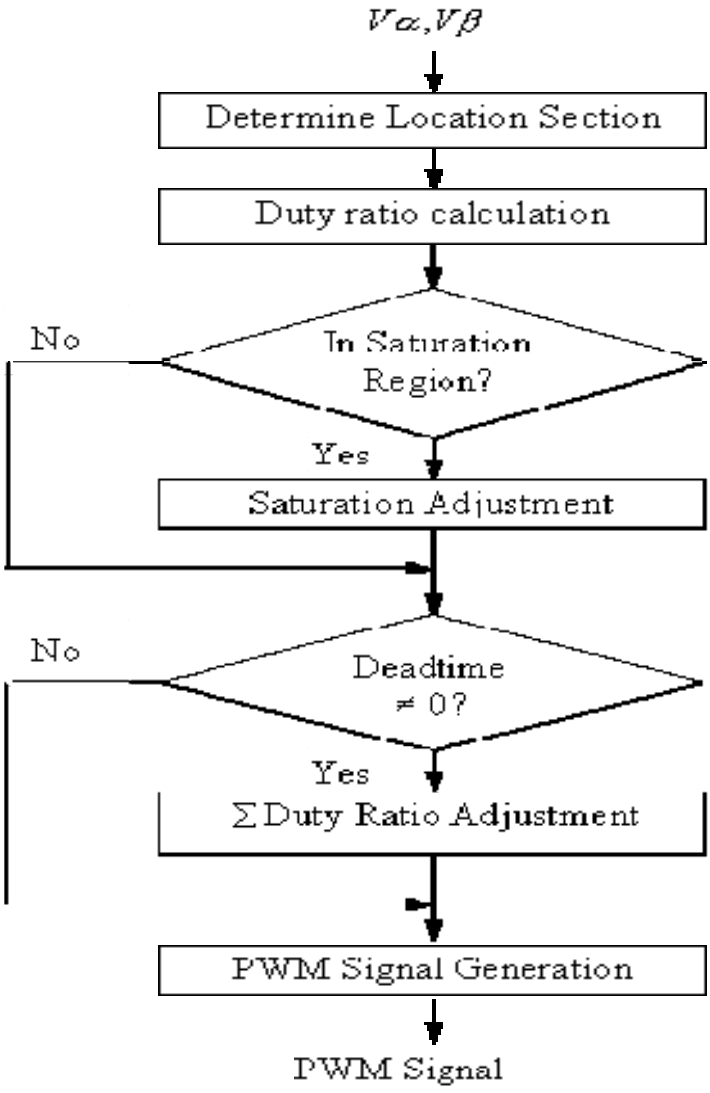


Figure 4.6 SVM Function Flow Chart

4.2.3 Space Vector Modulation

After the reference terminal voltage is obtained, it is used to generate the PWM signal by the SVM algorithm as required by the constraints outlined in Chapter II. Furthermore, the space vector modulation strategy suits the digital implementation well.

Also, some physical device will be selected as the switch of the converter, and its physical characters must be considered. When the insulated-gate-bipolar transistor (IGBT) converter is used, the turnoff characteristic of the IGBT model is approximated by two segments, the fall time (Tf) and the tail time (Tt). These two time parameters determine the transition time needed for the converter to switch from a leg. To avoid the dead time of an IGBT, it is necessary to adjust the duty ratio of a PWM signal. Thus, the SVM function flow chart in application can be modified as shown as in Figure 4.6.

Shunt active compensator is one common type of shunt connected current controller. In this chapter a shunt active compensator test system was presented. The control algorithm implemented in shunt connected current controller was described. The functionality of the compensator will be validated in next chapter.

CHAPTER V

SHUNT ACTIVE COMPENSATOR TEST RESULTS

This chapter presents the test results of a shunt active compensator. Test results include the results in MATLAB simulation, hardware-in-the-loop test, and the results of a physical hardware test.

5.1 System Parameters

A simple power system with a shunt current compensator is shown in Figure 5.1. The voltage source is assumed to be ideal and balanced. The nonlinear load is a 6-pulse converter with an inductor and a resistor. A shunt current controller, acting as an active compensator, is connected and consists of a three-phase voltage-fed PWM converter using IGBT, a dc capacitor C, three-phase interfacing inductors Lc, and resistors Rc.

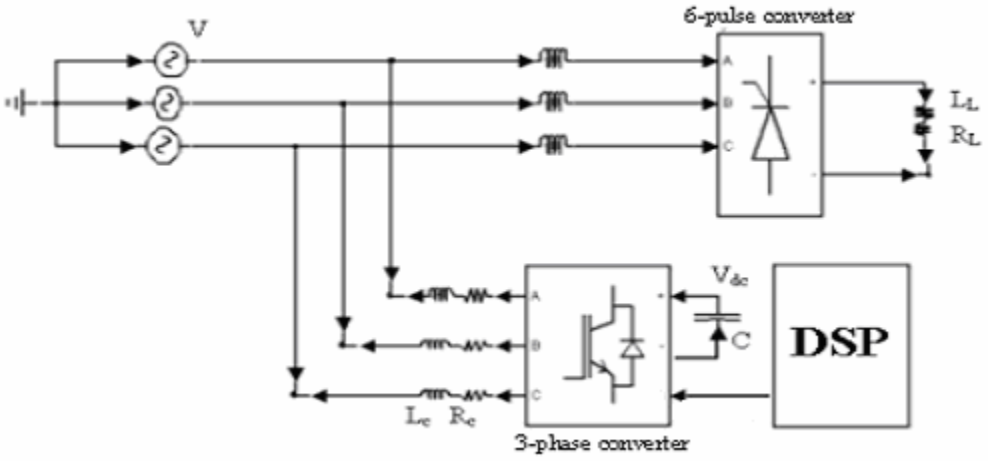


Figure 5.1 Power System with a Shunt Current Compensator

The system parameters are listed in Table 5.1. First, the shunt active compensator is simulated in the MATLAB, and then a hardware-in-the- loop test is performed to validate the performance of the hardware control part for the active compensator. Finally, the shunt active compensator is performed in the real plant.

Table 5.1 System Parameters

V	R_L	f	L_L	L_c	C	R_c	V_{dc}
121v	12Ω,	60Hz	0.1mh	1 mh	820uF	0.1 Ω	500 v

This system implements the d-q synchronous reference frame method to derive the reference, uses PI control to track the reference, and uses SVM to generate the PWM pulses. The control parameters in the control loop are listed in Table 5.2.

Table 5.2 Control Parameters for Shunt Active Compensator

K_p	K_i	K_1	K_2
12.8	12000	0.1408	1.8

5.2 Active compensator simulation in MATLAB

The system was built and simulated in the MATLAB/ Simulink environment shown in Figure 5.2.

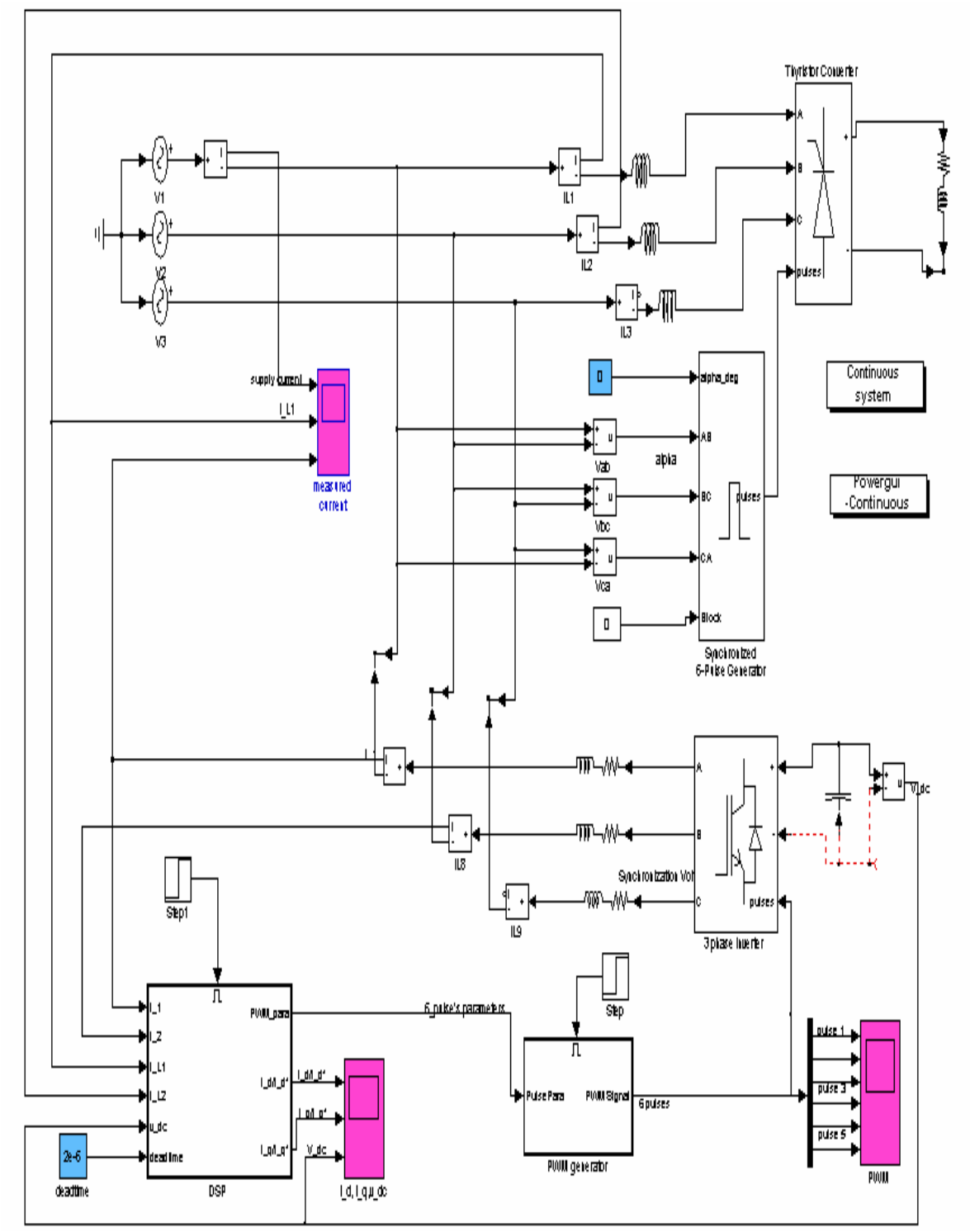


Figure 5.2 Configuration of Active Compensator in MATLAB/Simulink

Figure 5.3 shows the supply current and load current of the power system and the compensation current of the active compensator. Before compensation, the supply current should be as same as the load current. After compensation, the supply current is near sinusoid, meaning that the compensation current has compensated most of the harmonic current. Current spikes, which are due to the limited transient response of PI controllers, might be observed at supply current waveform.

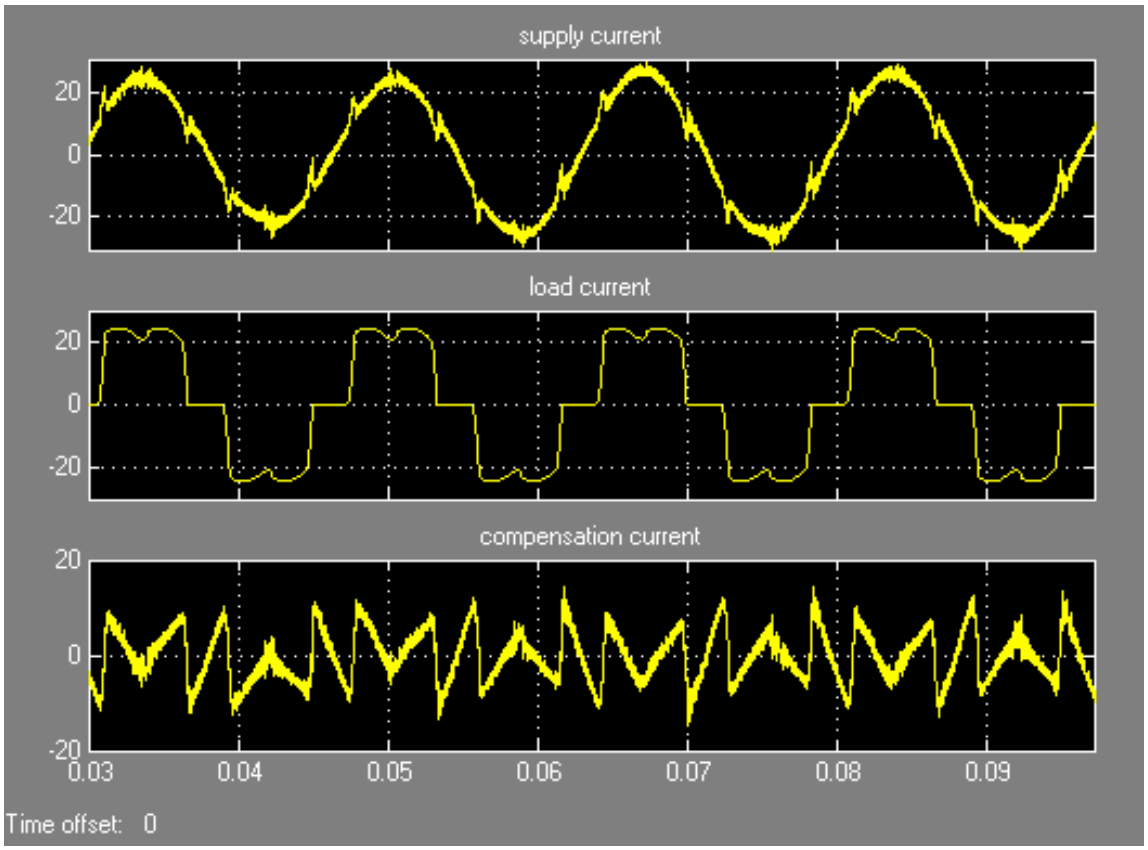


Figure 5.3 Simulated Active Compensator Behavior with α=0

In steady state operation, the total harmonic distortion (THD) of the non-linear load is 27.69%. After compensation, the THD of the supply current is 6.92%. Table 5.3

illustrates the individual amplitude of low order harmonics in the supply currents in percentages of the fundamental component (load firing angle is 0 degrees).

Table 5.3 Harmonic Contents in Supply Currents

Harmonic Order	1	5	7	11	13
Before Compensation	100	22.30	10.70	7.94	7.93
After Compensation	100	3.82	2.58	3.19	2.15

The control algorithms used in the shunt active compensator have been validated in simulation. The control algorithms will be implemented in a DSP, and the functionality of that DSP will be validated in HIL test.

5.3 Hardware-in-the-Loop Test

Figure 5.4 shows equipment used for the shunt current controller HIL, including a DSP, a real-time system simulator and a host computer. A PXI-8186RT controller from National Instruments was selected as the RT target. A PXI-6602 and a PXI-6733 were selected as data acquisition facilities. Also, the high-performance Motorola DSP56F807 shown in Figure 5.5 was used in the Hardware-in-the-loop test. The control algorithm was coded in C++ language, using the Metrowerks Code Warrior IDE and implemented in the DSP as part of an associated project [13].

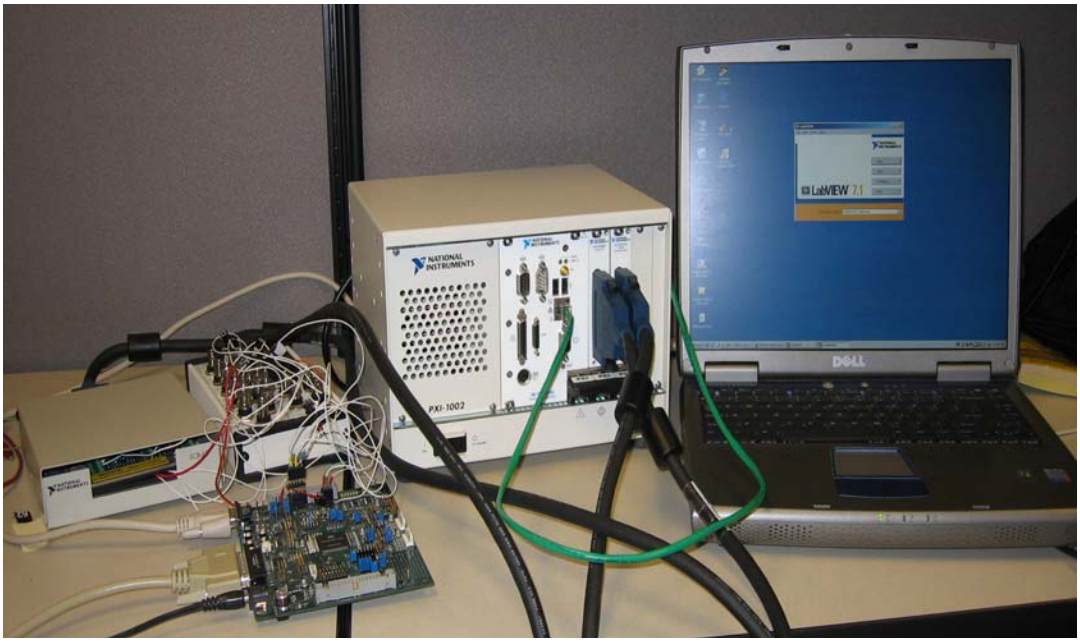


Figure 5.4 Hardware for Shunt Current Controller HIL

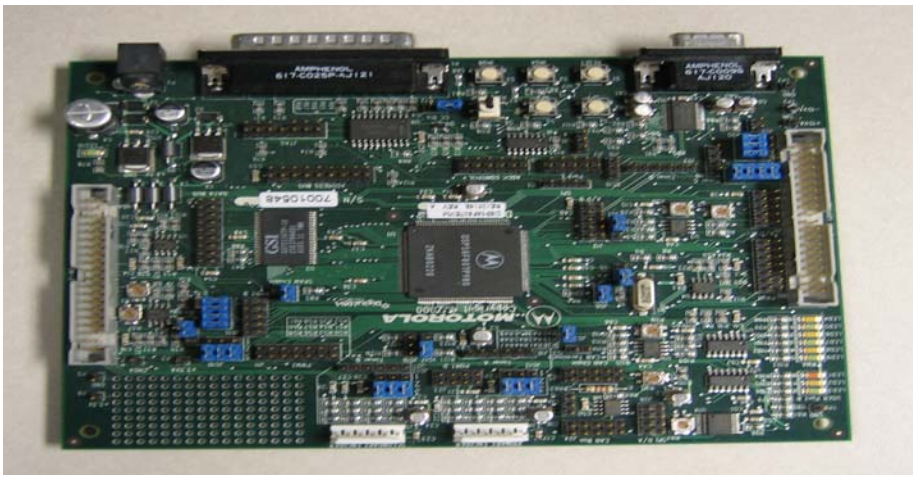


Figure 5.5 DSP56F807

Figure 5.6 shows the LabVIEW program for the HIL test. In the program, three channels of PXI-6602 are used for PWM signal reading. Seven channels of PXI-6733 are selected to output system components. Also, the mathematic model of the shunt current

compensator is performed in for loop. The detailed program for the mathematic model is listed in the Appendix D.

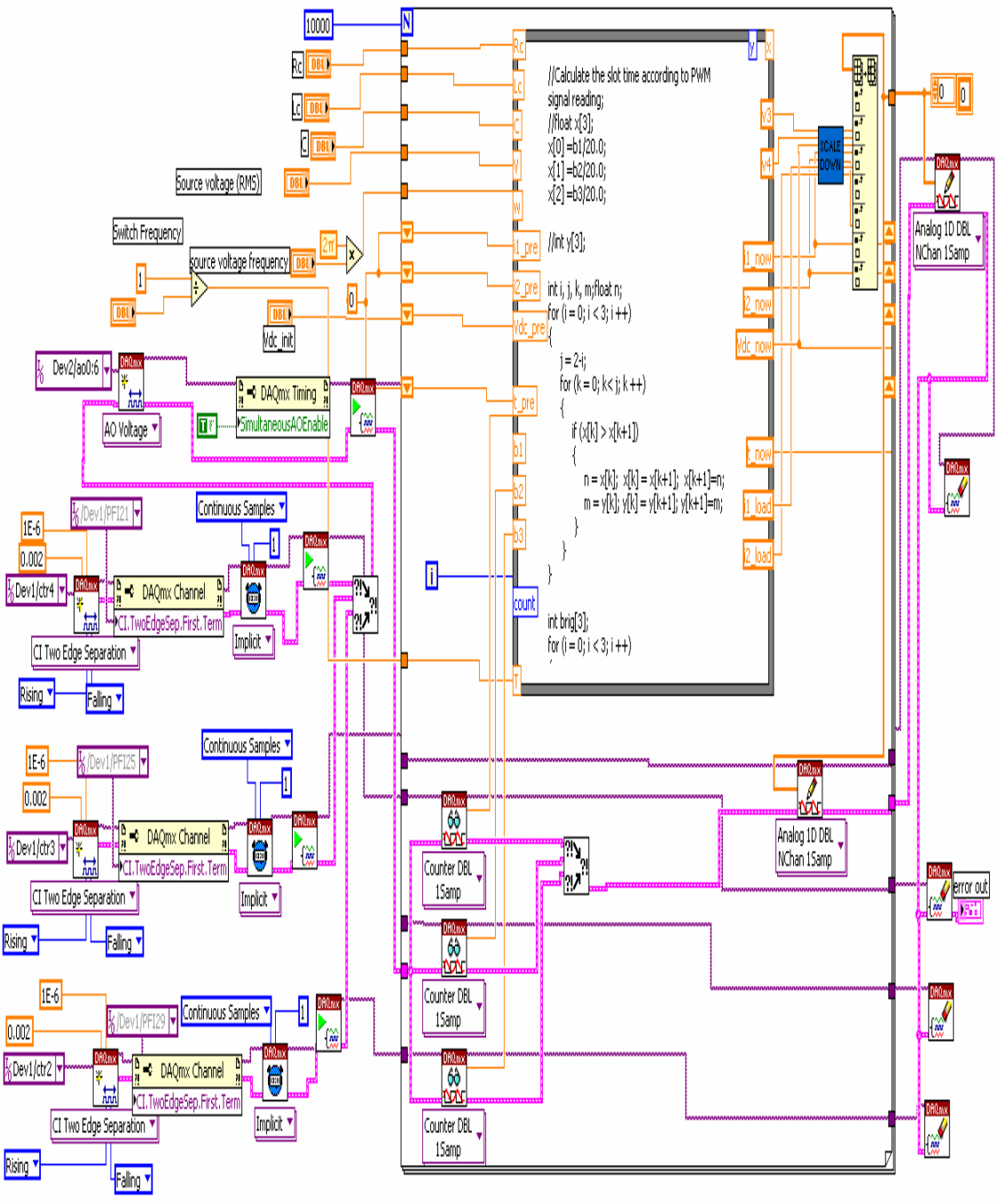


Figure 5.6 LabVIEW Program for HIL Test

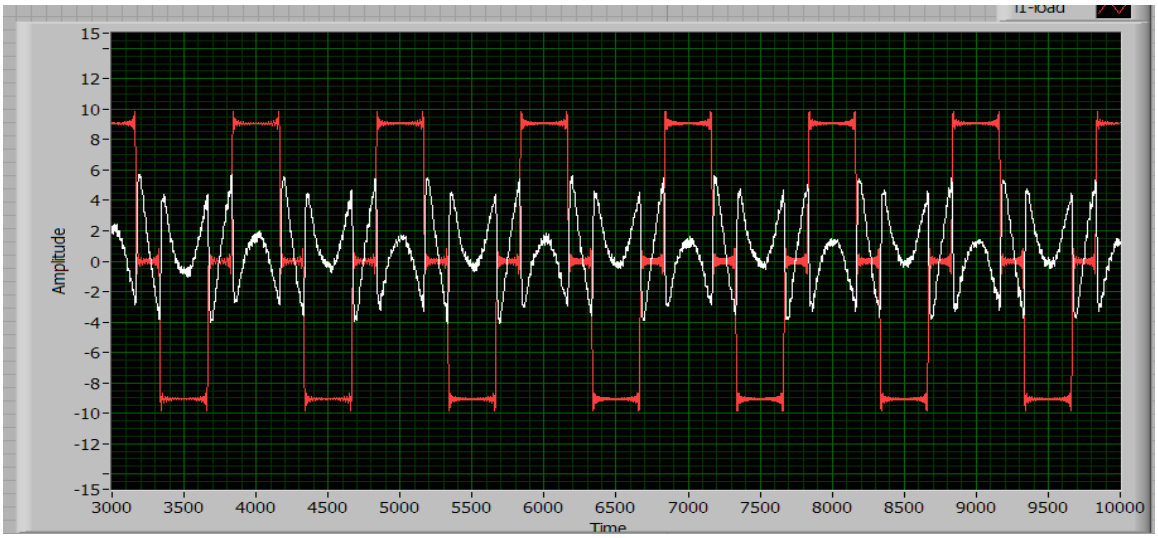


Figure 5.7 Load Current (red) and Compensation Current (white)

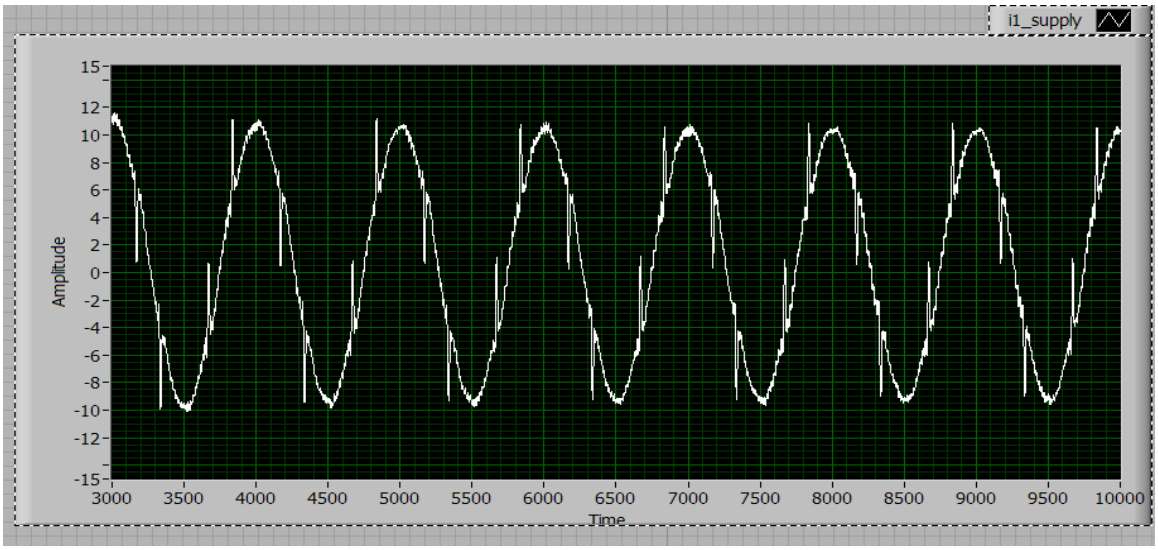


Figure 5.8 Supply Current

Figure 5.7 shows the load current (red) and compensation current (white), and Figure 5.8 shows the supply current. Although there are some high frequency components in the supply current, the supply current is very close to the sinusoid. With the control algorithm implemented in the DSP, the functionality of DSP has been

validated in the HIL test. Now, the DSP is ready to be inserted into the hardware of the real plant for the final test.

5.4 Hardware Test

The power circuit of the converter is adopted from the commercial PEBB PM1000, developed by American Superconductor. The 240V three-phase industrial power supply is connected in series with the load. To check the performance of the compensator, the commercial 6RA70 thyristor control rectifier (TCR) from SIEMENS was employed. Figure 5.9 shows the laboratory experimental setup.

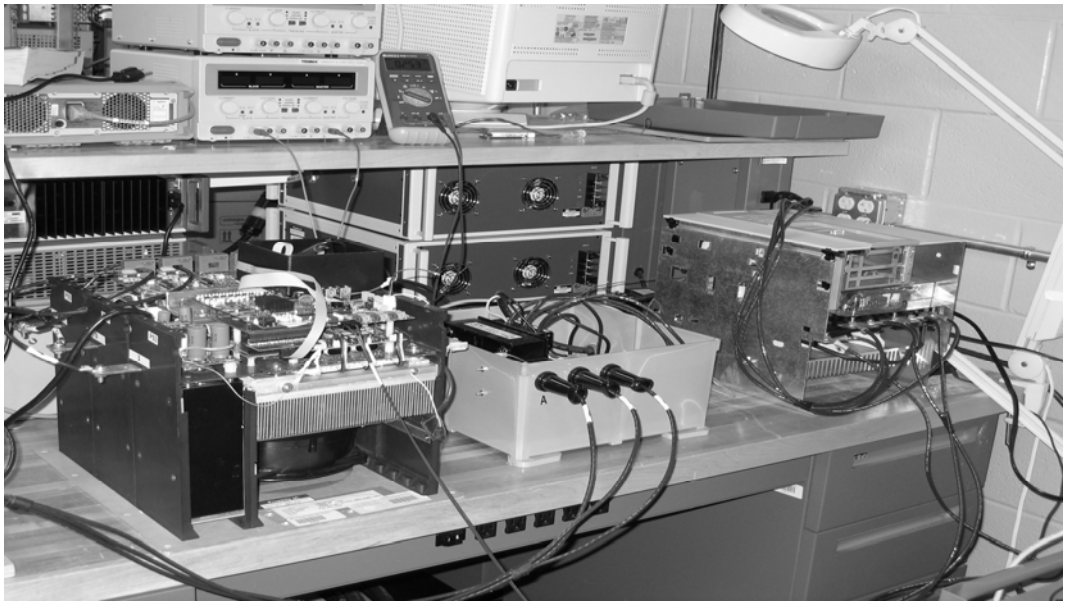


Figure 5.9 Experimental Setup. (PM1000-left, TCR load-right) [13]

Experimental results for the compensation current, supply current and load current [13] are presented in Figure 5.10.

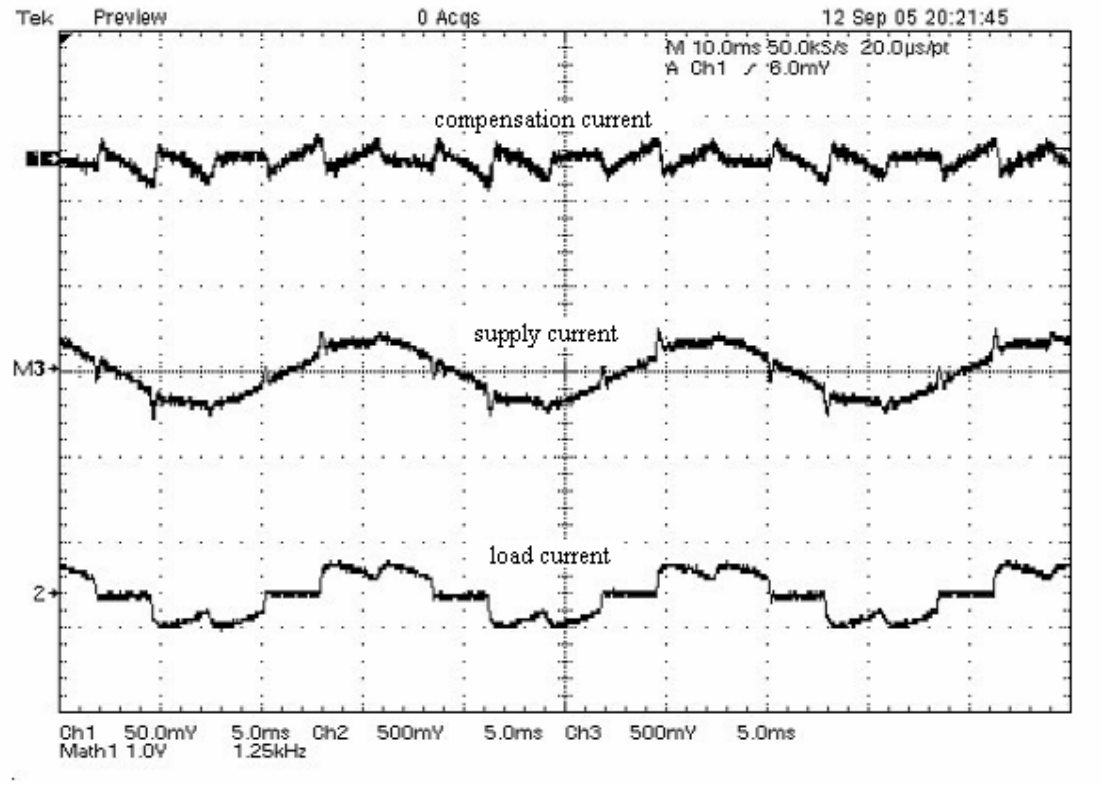


Figure 5.10 Experimental Results for the Compensator Current, Supply Current and Load Current [13]

Experimental results show that with the control algorithm applied in DSP, the shunt active compensator can compensate the harmonics current caused by nonlinear load. With the formal procedure, the design of the shunt current compensator is finished.

CHAPTER VI

CONCLUSION

This thesis presents a real-time controller-in-the-loop test system for a digital controller design in power system applications. A formal procedure to develop a shunt current controller is presented from software simulation to hardware implementation. To efficiently evaluate the various digital controller designs for a shunt current controller, a controller-in-the-loop simulator for the DSP controller is beneficial. Requirements for a real-time system simulator for a shunt current controller are discussed, and a hardware implementation is presented. The results demonstrate the following contributions of this work:

1) A formal procedure to develop a digital controller is given.
2) A hardware-in-the-loop simulation is provided for the digital controller in order to evaluate control algorithms without the voltage source converter and power system
3) Simulation models for the example system are presented, along with an implementation of that system in the real-time system simulator. An active compensator prototype is first developed in MATLAB/Simulink. Then, following a formal design procedure, the power system is modeled in a digital simulator, and the controller is implemented in a digital controller board. Finally, a hardware-in-the- loop test is performed to validate the

performance of the hardware controller for the active compensator.

4) Experimental results show that following the formal procedure, it is able to verify the design of digital controller prior to full hardware development.

In the future, the performance comparison of the hardware-in-the-loop test with the physical system test can help to refine the models in the digital simulator. The load current is simplified and needs to be simulated by more appropriate mathematical models that reflect real devices.

The models also need to be validated in other applications with other different environments, such as different loads and controls. As mentioned before, this research work has simplified the model of the power circuit to achieve fast simulation speed. Obviously, in order to facilitate the test with different power system configurations, a comprehensive common model is needed to describe every aspect in the power system. This work needs to be extended to other components of power systems. Finally, considering some new features provided by MATLAB/Simulink to transfer the power system model in Simulink to LabVIEW callable code, it might be possible to build any complex power system in Simulink and make it executable in a digital simulator. This is a future goal that would make the real-time system simulation more flexible.

REFERENCES

[1] N. Mendalek, and K. Al-Haddad, "Modeling and Nonlinear Control of Shunt Active Power Filter in the Synchronous Reference Frame", *Proceedings of Ninth International Conference on Harmonics and Quality of Power*, vol.1, Oct. 2000, pp. 30-35.

[2] B. Singh, K. Al-Haddad, and A. Chandra, "A review of active filters for power quality improvement ", *IEEE Transactions on Industrial Electronics*, vol.46, no.5, Oct. 1999, pp.960-971.

[3] M. El-Habrouk, M.K. Darwish, and P. Mehta, "Active power filters: a review", *IEEE Proceedings on Electric Power Applications*, vol.147, no.5, Sep. 2000, pp.403-413.

[4] H. Akagi. "Trends in Active power line conditioners", *IEEE Transactions on Power Electronics*, vol. 9, no. 3, May 1994, pp. 263 –268.

[5] J. Nastran, R. Cajhen, M. Seliger, and P. Jereb, "Active power filter for nonlinear AC loads", *IEEE Transaction on Power Electronic*, vol. 9, Jan. 1994, pp. 92-96.

[6] V. B. Bhavaraju, and P. N. Enjeti, "Analysis, design of an active power filter for balancing unbalanced loads", *IEEE Transactions on Power Electronic*, vol. 8, Oct. 1993, pp. 640-647.

[7] V. B. Bhavaraju, and P. Enjeti, "A novel active line conditioner for a three-phase system", *Proceedings of IEEE Industry Applications Society Annual Meeting*, vol.2, Oct. 1993, pp. 979-985.

[8] A. Campos, G. Joos, P. D. Ziogas, and J. F. Lindsay, "Analysis, design of a series voltage unbalance compensator based on a three-phase VSI operating with unbalanced switching functions", *IEEE Transactions on Power Electronic*, vol. 9, May 1994, pp. 269-274.

[9] C. Donghua, and X. Shaojun, "Electric Utility Deregulation, Restructuring and Power Technologies", *Proceedings of the 2004 IEEE International Conference*, vol.2, Apr. 2004, pp.666-670.

[10] A. Monti, E. Santi, R. A. Dougal, M. Riva, "Rapid prototyping of digital controls for power electronics", *IEEE Transactions on Power Electronics*, pp.915-923, vol.18, May 2003

[11] F. Wang, S. Rosado, and D. Boroyevich, "Open modular power electronics building blocks for utility power system controller applications", *Proceedings of IEEE Power Electronics Specialist Conference*, vol. 4, June 2003, pp.1792 – 1797.

[12] S. Rosado, F. Wang, D. Boroyevich, and R. Wachal, "Control interface characterization of power electronics building blocks (PEBB) in utility power system applications", *Proceedings of IEEE Power Engineering Society General Meeting*, vol. 3, July 2003, pp. 1355.

[13] H. Qinghua, K. Borisov, and H. L. Ginn III, "PEBB-based shunt active power filter for shipboard power systems", *Proceedings of IEEE Electric Ship Technologies Symposium*, July 2005, pp.393 – 399.

[14] Z. Keliang, and W. Danwei, "Relationship between space-vector modulation and three-phase carrier-based PWM: a comprehensive analysis [three-phase inverters]", *IEEE Transactions on Industrial Electronics*, vol. 49, Feb. 2002, pp.186 – 196.

[15] D. Rathnakumar, J. LakshmanaPerumal, and T. Srinivasan, "A new software implementation of space vector PWM", *Proceedings of IEEE Southeast Conference*, April 2005, 8-10 pp.131 – 136.

[16] A.E. Nava-Segura, J.Linares-Flores, and G. Mino-Aguilar, "Vector active filter without current hysteresis controllers", *Proceedings of the 2000 IEEE International Symposium on Industrial Electronics*, vol.1, Dec. 2000, pp.84 – 89.

[17] D.Shen, and P.W. Lehn, "Fixed-frequency space-vector-modulation control for three-phase four-leg active power filters", *Proceedings of IEEE Electric Power Applications,* vol.149, July 2002, pp.268 – 274.

APPENDIX A

NATIONAL INSTRUMENTS PXI-6602

The NI PXI-6602 is a timing and digital I/O module with eight 32-bit counter/timers and 32 lines of TTL/CMOS-compatible digital I/O. Eight digital I/O lines are dedicated and the remaining 24 are shared with the counter/timers. a wide variety of counter/timer tasks can be performed with the NI PXI-6602, including encoder position measurement, event counting, period measurement, pulse width measurement, pulse generation, pulse train generation, and frequency measurement. A specification of NI PXI-6602 related in the thesis experiment is summarized as below. For detail specification of NI PXI-6602, please check the website of ni.com

NI PXI-6602

Timing I/O

General-Purpose Up/Down Counter/Timers

Number of channels..... 8 up/down counters
Counter size/number of bits..... 32 bits

Digital logic levels

Level	Minimum	Maximum
Input low voltage	-0.3 V	0.8 V
Input high voltage	2.0 V	5.25 V

Base clocks..... .....100 kHz, 20 MHz, and 80 MHz
Base clocks accuracy..... ±0.005%

Data Transfers

Transfer modes..... DMA, interrupts, programmed I/O

Transfer rates

	DMA		Interrupt	
Continuous Operation	Buffer Size (MS/s)	Rate (ks/s)	Buffer Size (MS/s)	Rate (ks/s)
	50	28	50	28

DMA channels..... 3

Digital I/O

Number of channels..... Up to 32 input/output
Compatibility .....5 V TTL/CMOS

Digital logic levels

Level	Minimum	Maximum
Input low voltage	-0.3 V	0.8 V
Input high voltage	2.0 V	5.25 V

PXI Trigger Bus

Trigger lines..... 6
Star trigger 1

APPENDIX B

NATIONAL INSTRUMENTS NI PXI-8186

The National Instruments PXI-8186 is a high-performance Pentium 4 embedded controller for use in any PXI or CompactPCI system. The NI PXI-8186 is ideal for applications requiring intensive analysis or software development. A PXI-8186 embedded controller in PXI chassis offers a compact, high-performance PC platform for modular instrumentation and data acquisition applications. The feature of NI PXI-8186 Embeded controller is listed below:

NI PXI-8186

CPU	2.2 GHz Pentium 4-M
On-die cache	512 KB
DDR RAM	256 MB, standard 1 GB, maximum
Hard drive	30 GB, minimum
100 Base TX^2 Ethernet	Yes
GPIB {IEEE 488.2} interface	Yes
Serial ports	2
Parallel port	Yes
USB 2.0 ports	2
PXI trigger bus input/output	Yes
Operating system	Windows 2000/XP, LabVIEW Real-Time

APPENDIX C

NATIONAL INSTRUMENTS PXI-6733

The National Instruments PXI-6733 delivers high-performance, reliable, high-speed analog outputs to meet a wide range of application requirements. It provides eight analog output channels at 1 MS/s per channel, 16-bit resolution, and digital triggering. In addition, the module can perform high-speed digital pattern generation/detection up to 10 Mwords/s. The module uses the PXI trigger bus to synchronize with additional data acquisition, motion and vision products, so users can create custom measurement solutions to test innovative designs. The NI PXI-6733 is ideal for applications such as stimulus-response tests, including acoustic distortion testing, and open-loop simulation, including 3-phase power simulation. This device can also be used with LabVIEW Real-Time to perform real-time control, including hardware-in-the-loop simulation and rapid-control prototyping. A specification of NI PXI-6602 related in the thesis experiment is summarized as below. For detail specification of NI PXI-6602, please check the website of ni.com

NI PXI-6733
Analog Output
Output Characteristics
Number of channels ..... 8 voltage outputs
Resolution.....16 bits, 1 in 65,536

Maximum Update Rate of NI PXI-6733

Number of Channels	Using Local FIFO (ks/s)	Using Host FIFO (ks/s)
1	1000	1000
2	1000	1000
3	1000	1000
4	1000	1000
5	1000	1000
6	952	1000
7	833	869
8	740	769

FIFO buffer size.....16,384 samples

Voltage Output
Ranges..... ±10.0 V, ±AO EXT REF
Output coupling ..... DC
Protection ..... Short-circuit to ground

Digital I/O

Number of channels..... 8 input/output
Compatibility ..... 5 V TTL/CMOS
Power-on state ..... Input (high-impedance)
Data transfers ..... Programmed I/O, DMA, interrupts
Input buffer..... 2048 bytes
Output buffer ..... 2048 bytes
Transfer rate ..... 10 Mwords/s

Timing I/O

General-Purpose Up/Down Counter/Timers
Number of channels..... 2
Resolution..... 24 bits
Compatibility ..... 5 V TTL/CMOS
Digital logic levels

Level	Minimum	Maximum
Input low voltage	0 V	0.8 V
Input high voltage	2 V	5 V
Output low voltage (Iout = 5 mA)	–	0.4 V
Output high voltage (Iout = -3.5 mA)	4.35 V	–

Base clocks available..... 20 MHz and 100 kHz

Digital Trigger

Purpose
Analog output ..... Start trigger, gate, clock
General-purpose counter/timers ..... Source, gate
Source..... PFI <0...9>, RTSI <0...6>
Slope..... Positive or negative; software selectable
Compatibility ..... 5 V TTL/CMOS

APPENDIX D

LabVIEW PROGRAM IN FORMULA NODE FOR HIL TEST

Formula Node in LabVIEW is used to perform computation of active compensator mathematical model. Active compensator mathematical model is set up with Equation 3.9. For formula node, it gets variables values from input terminals for calculation and sends out the values of variables through output terminals.

For the formula node, input terminals are Rc, Lc, C, V, w, i1_pre, i2_pre, Vdc_pre, t_pre, b1, b2, b3, T. Where Rc is the interface resistance; Lc is the interface inductance; C is the capacitor capacity; V is the magnitude of the source voltage; w=2* Pi*f, where f is the source frequency; i1_pre, i2_pre are the compensation currents for phase a, b at previous time step; Vdc_pre is the capacitor voltage at previous time step; t_pre is the time of at previous time step; b1, b2, b3 are the measurements of PWM signals; T is time step.

Output terminals of the formula node are i1_now, i2_ now, Vdc_ now, t_ now, v3, v4, i1_load, and i2_load. Where i1_now and i2_ now are the compensation currents for phase a, b at current time step; Vdc_ now is the capacitor voltage at current time step; t_ now is the time of at current time step; v3 and v4 are the source voltages of phase a and b at current time step; i1_load, i2_load are the load currents of phase a, b at current time step.

The program in the formula node is as follows:

```
//Calculate the slot time according to PWM signals reading;
float x[3];
x[0] =b1/4.0;
x[1] =b2/4.0;
x[2] =b3/4.0;

int y[3];
```

```
int i, j, k, m;
float n;
for (i = 0; i < 3; i ++)
{
    j = 2-i;
    for (k = 0; k< j; k ++)
    {
        if (x[k] > x[k+1])
        {
            n = x[k];  x[k] = x[k+1];  x[k+1]=n;
            m = y[k]; y[k] = y[k+1]; y[k+1]=m;
        }
    }
}

int brig[3];
for (i = 0; i < 3; i ++)
{
   m = y[i];
   brig[m] = i;
}

int c[3][7];
for (i = 0; i < 3; i ++)
{
   for (j= 0; j < 7; j ++)
   {
      c[i][j] = z[brig[i]][j];
   }
}

float t[7];
t[0]=t[6]=x[0];
t[1]=t[5]=x[1]-x[0];
t[2]=t[4]=x[2]-x[1];
t[3]=T-x[2]*2;

//according to the slot time, calculate the system elements;

float delta_i1,delta_i2,delta_Vdc,v1,v2;
float dn1,dn2;

for (i = 0; i < 7; i ++)
{
```

```
    dn1=(2*c[0][i]-c[1][i]-c[2][i])/3;
    dn2=(2*c[1][i]-c[0][i]-c[2][i])/3;

    v1=sqrt(2)*V*cos(w*t_pre);
    v2=sqrt(2)*V*cos(w*t_pre-2*pi/3);

    delta_i1=((-Rc*i1_pre-dn1*Vdc_pre)+v1)/Lc;
    delta_i2=((-Rc*i2_pre-dn2*Vdc_pre)+v2)/Lc;
    delta_Vdc=((2*dn1+dn2)*i1_pre+(dn1+2*dn2)*i2_pre)/C;

    Vdc_pre=Vdc_pre+delta_Vdc*t[i];
    i1_pre=i1_pre+delta_i1*t[i];
    i2_pre=i2_pre+delta_i2*t[i];

    t_pre=t_pre+t[i];
}

// program does not calculate at the first period to avoid dynamic state;
if  (count <1000)
   {
  i1_now=0;
  i2_now=0;
 Vdc_now=300;
}
else
{
  i1_now=i1_pre;
  i2_now=i2_pre;
 Vdc_now=Vdc_pre;
}
 t_now=t_pre;

v3=sqrt(6)*V*cos(w*t_now+pi/6);
v4=sqrt(6)*V*cos(w*t_now-pi/2);

 i1_load=0.5*(20*cos(w*t_now)-4*cos(5*w*t_now)+20/7*cos(7*w*t_now)-
20/11*cos(11*w*t_now)+20/13*cos(13*w*t_now)-
20/17*cos(17*w*t_now)+20/19*cos(19*w*t_now)-
20/23*cos(23*w*t_now)+20/25*cos(25*w*t_now)-
20/29*cos(29*w*t_now)+20/31*cos(31*w*t_now)-
20/35*cos(35*w*t_now)+20/37*cos(37*w*t_now)-
20/41*cos(41*w*t_now)+20/43*cos(43*w*t_now)-
20/47*cos(47*w*t_now)+20/49*cos(49*w*t_now)-
20/53*cos(53*w*t_now)+20/55*cos(55*w*t_now));
```

```
i2_load=0.5*(20*cos(w*t_now-2*pi/3)-4*cos(5*w*t_now-
10*pi/3)+20/7*cos(7*w*t_now-14*pi/3)-20/11*cos(11*w*t_now-
22*pi/3)+20/13*cos(13*w*t_now-26*pi/3)-20/17*cos(17*w*t_now-
34*pi/3)+20/19*cos(19*w*t_now-38*pi/3)-20/23*cos(23*w*t_now-
46*pi/3)+20/25*cos(25*w*t_now-50*pi/3)-20/29*cos(29*w*t_now-
58*pi/3)+20/31*cos(31*w*t_now-62*pi/3)-20/35*cos(35*w*t_now-
70*pi/3)+20/37*cos(37*w*t_now-74*pi/3)-20/41*cos(41*w*t_now-
82*pi/3)+20/43*cos(43*w*t_now-86*pi/3)-20/47*cos(47*w*t_now-
94*pi/3)+20/49*cos(49*w*t_now-98*pi/3)-20/53*cos(53*w*t_now-
106*pi/3)+20/55*cos(110*w*t_now));
```