FERRORESONANCE IN CAPACITIVE VOLTAGE TRANSFORMER (CVT) DUE TO BREAKER OPENING

SHAKIL AHAMED KHAN

DISSERTATION SUBMITTED IN FULFILMENT
OF THE REQUIREMENT
FOR THE DEGREE OF MASTER OF PHILOSOPHY

INSTITUTE OF GRADUATE STUDIES
UNIVERSITY OF MALAYA
KUALA LUMPUR

2015

ORIGINAL LITERARY WORK DECLARATION

Name of the candidate: **Shakil Ahamed Khan**

Registration/Matric No: **HGF 120006**

Name of the Degree: **Master of Philosophy**

Title of Project Paper/Research Report/Dissertation/Thesis (This Work):
Ferroresonance in Capacitive Voltage Transformer (CVT) Due To Breaker Opening

Field of Study: **Power Electronics**

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this work;

(2) This work is original;

(3) Any use of my work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the work and its authorship have been acknowledged in this work;

(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;

(5) I hereby assign all and every rights in the copyright to this work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;

(6) I am fully aware that if in the course of making these works I have infringe any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date

Subscribe and solemnly declare before,

Witness Signature Date

Name :

Designation :

ABSTRACT

Ferroresonance incidences in electrical power system have been commonly regarded as unexplained phenomenon due to its relatively rare frequency of occurrence which is not critical by the utility engineers. As a result, research conducted in this area is limited and the awareness on ferroresonance is relatively low amongst the utility engineers. However, as the electrical system evolves, its complexity increases in line with the increasing risk of ferroresonance. Ferroresonance gained prominence only in the recent decade, when it has been reported to cause damaging consequences to power equipment. Several literatures had concerning reported practical encounters of ferroresonance which led to equipment failures and electrical blackout. It must be noted that most of the literatures concentrated on ferroresonance in power transformer only. In contrast, this research will place ferroresonance in capacitive voltage transformer (CVT) as the main focus. It is demonstrated in this research that ferroresonance can also occur in CVT due to circuit breaker switching. Various ferroresonance suppression techniques have since been proposed as ferroresonance mitigation solutions in CVT. This research presents a new technique for detection and mitigation of the ferroresonance phenomenon in CVT. In addition, the transient performance of CVT and ferroresonance mitigation performance with the proposed new technique is also compared with other existing ferroresonance suppression techniques. EMTP-RV simulation results demonstrate that, the transient response for a CVT with the proposed ferroresonance suppression circuit (FSC) is much better than conventional active and passive FSCs. The accuracy of the proposed ferroresonance detection and mitigation technique is verified through comparison of the laboratory test (Hardware-in-the-Loop (HIL) real-time simulations) results and with those obtained from EMTP-RV simulation results. Closed-loop testing is performed using real time digital simulator (RTDS). The experimental results demonstrate that the developed technique can accurately detect the

phenomenon of ferroresonance in CVT and can suppress ferroresonance faster than other conventional techniques.

ABSTRAK

Insiden ferroresonance dalam sistem kuasa elektrik telah biasa dianggap sebagai fenomena yang tidak dapat dijelaskan kerana kekerapan yang agak jarang berlaku dan yang tidak kritikal kepada jurutera utiliti. Akibatnya, penyelidikan yang dijalankan dalam bidang ini adalah terhad dan kesedaran pada ferroresonance agak rendah di kalangan jurutera utiliti. Walau bagaimanapun, perkembangan sistem elektrik meningkatkan kerumitan sejajar dengan risiko ferroresonance. Kebelakangan ini, ferroresonance menjadi terkenal apabila ia telah dilaporkan menyebabkan impak kerosakkan pada peralatan sistem kuasa voltan tinggi. Beberapa kajian lepas telah melaporkan mengenai penemuan praktikal ferroresonance yang membawa kepada kegagalan peralatan elektrik. Kebanyakan kajian yang lepas tertumpu pada ferroresonance dalam pengubah kuasa sahaja. Sebaliknya, tumpuan utama kajian ini adalah pada ferroresonance yang berlaku dalam pengubah voltan kapasitif (CVT). Kajian lepas menunjukkan yang ferroresonance juga boleh berlaku dalam CVT disebabkan oleh litar pemutus pensuisan. Pelbagai teknik penindasan ferroresonance telah pun dicadangkan sebagai penyelesaian untuk memitigasikan ferroresonance dalam CVT. Kajian ini membentangkan satu teknik yang baharu untuk mengesan dan mengurangkan fenomena ferroresonance dalam CVT. Di samping itu, prestasi CVT dan prestasi mitigasi ferroresonance dengan satu teknik yang baharu akan dicadangkan juga dibanding dengan teknik-teknik penindasan ferroresonance sedia ada. Keputusan simulasi EMTP - RV menunjukkan bahawa, respon transient untuk CVT dengan litar penindasan ferroresonance (FSC) yang dicadangkan adalah jauh lebih baik daripada FSCs aktif danpasif. Ketepatan pengesanan ferroresonance dan teknik mitigasi yang dicadangkan telah disahkan melalui perbandingan ujian makmal “Hardware- in-the – Loop” (HIL) menggunakkan simulasi yang nyata dan keputusan yang diperolehi daripada hasil simulasi EMTP – RV dan ujian gelung tertutup dilakukan menggunakan

simulator digital masa sebenar (RTDS). Keputusan eksperimen menunjukkan bahawa teknik yang dibangunkan dapat mengesan fenomena ferroresonance dalam CVT dengan tepat, dan boleh menyekat ferroresonance lebih cepat daripada teknik konvensional yang lain.

ACKNOWLEDGEMENT

First of all, I am I am really grateful to Allah in blessing me with the knowledge, giving me the courage to tackle all problems and helping me in every step of my life.

I would like to express my gratitude to my supervisor Prof. Dr. Nasrudin Bin Abd Rahim for motivating and guiding me during my thesis work. It has been a pleasure to collaborate with him and I hope to continue. His wise experience in the field of electrical power engineering has enlightened me throughout the project.

I would like to express my indebted gratitude to my supervisor Dr. Ab Halim Bin Abu Bakar for his outstanding support, contribution and invaluable assistance in the achievement and development of my MPhil thesis.

I cannot find the right words to express the admiration and sincere gratitude towards Dr. Tan Chia Kwang for helping me. His suggestions were always valuable and his technical comments lead to the completion of the project.

I am greatly indebted to my father, mother and younger brother for their continuous loving support, inspiration and encouragement.

I also express my gratitude to all UMPEDAC staff for helping me directly or indirectly to carry out my research work. I gratefully acknowledge the privileges and opportunities offered by the University of Malaya.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

AC	Alternating Current
FSC	Ferroresonance suppression circuit
FFT	Fast Fourier transform
DC	Direct Current
AFFC	Active ferroresonance suppression circuit
PFFC	Passive ferroresonance suppression circuit
EFFC	Electronic ferroresonance suppression circuit
EMTP	Electromagnetic transient program
CB	Circuit breaker
CVT	Capacitor voltage transformer
HV	High Voltage
MV	Medium Voltage
EHV	Extra High Voltage
UHV	Ultra High Voltage
EMTP-RV	Electro Magnetic Transient Program-Restructure Version
IEEE	Institute of Electrical and Electronics Engineers
OC	Over Current
OV	Over Voltage
pu	per unit
kV	kilo Volts
kA	Kiloamperes
Hz	Hertz
ms	millisecond
MOV	Metal Oxide Varistor
CT	Current Transformer
PT	Potential Transformer

SDT	Step Down Transformer

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

1.1 Background

The power community recognized that any form of disturbance to the perfect sine wave of voltage and current is undesirable due to the stringent demand from the sensitive loads (Bollen, 2000; Dugan, 2003). Ferroresonance is one of the events which also distorts the voltage and current sine waves. However, the awareness of ferroresonance amongst utility engineers has been relatively low due to the rare occurrence and difficulties in detection. In addition, the absence of direct impact to the consumers further contributed to the perception that ferroresonance is insignificant. In contrast, the voltage sags and harmonics are always being paid the most attention due to the ease in detection and obvious consequences to customer loads.

Notwithstanding the fact that power quality events are much more popular than ferroresonance, it must be noted that ferroresonance was actually discovered even before power quality terms were formalized, agreed and accepted into the standards. The authors in (Ta-Peng & Chia-Ching, 2006) suggested that works related to ferroresonance was first carried out on analysis in transformer and was published in 1907 (Valverde, 2011). Subsequently, the word ‘ferroresonance’ was introduced by Boucherot in his analysis on complex resonance oscillation in series RLC circuit with nonlinear inductance (Bethenod, Nov. 30, 1907).

Few literatures published over the years of research had contradicted the common perception that ferroresonance will not have a significant impact on the power network. It was reported that the distorted waveforms in ferroresonance will not directly affect the sensitive loads. Instead, the literatures reported that ferroresonance causes failures in power equipment, which

would indirectly affect the consumers and eventually lead to even greater consequences of major blackouts.

These literatures reported practical encounters of the damaging consequences in power equipment due to ferroresonance. The authors in (Dugan, 2003) presented three practical experiences of ferroresonance in transformers. It was reported in all three cases that the transformers produced loud noise during ferroresonance and signs of heating were observed on the transformers. It was reported in (Simha, 2008) that a lightning arrestor failed catastrophically at the moment single phase cutouts were closed one after another. The root cause was then traced to ferroresonance. The authors in (Tanggawelu, 2003) reported several incidents of overvoltages in Malaysian distribution network, which led to equipment damages. The overvoltages are then traced to ferroresonance as well. In contrast, the authors in (Escudero, 2004) experienced ferroresonance in voltage transformers during commissioning of a new 400kV substation in Ireland. Subsequently, the authors repeated the same switching operations and consistent overvoltage waveforms were captured. It was verified that the consistent overvoltages were caused by ferroresonance. Finally, ferroresonance had also been identified as the root cause to a flashover which led to blackout in Taiwan on 18th March 2001 (Ta-Peng & Chia-Ching, 2006). The incident started with the tripping of a 345kV system, which causes the reactor cooling pump motor with flywheel to supply reverse power flow to the system. This reverse power flow acted as the ac source and interacted with the system inductance and capacitance to produce ferroresonance. The combination of high overvoltages and low frequency in the ferroresonance then causes insulation breakdown in 4.16kV air circuit breaker of the nuclear power station, which led to flashover and consequently major blackout.

A large majority of the research conducted till now concentrated mainly on investigation of ferroresonance in power transformer only. However, the latest research revealed that ferroresonance can also occur in capacitive voltage transformer (CVT). As such, it is vital for a research to be conducted to investigate the risk of ferroresonance occurrence and mitigation techniques in CVT. It must also be noted that the presence of Ferroresonance suppression circuit (FSC) will affect the transient performance of the CVT due to the non-linear components in the FSC (Badrkhani Ajaei, Sanaye-Pasand, Rezaei-Zare, & Iravani, 2009; Graovac, Iravani, Wang, & McTaggart, 2003; Sakamuri & Yesuraj, 2011; Sanaye-Pasand, Rezaei-Zare, Mohseni, Farhangi, & Iravani, 2006). As such, it is of utmost importance for a study to be conducted on the ferroresonance suppression and transient response for the different types of FSCs.

1.2 Problem Statement

Various ferroresonance suppression techniques have been proposed as mitigation solutions for ferroresonance in CVT. The most common ferroresonance suppression techniques are the active FSC and passive FSC. It must also be noted that the presence of active FSC and passive FSC will affect the transient performance of the CVT due to the non-linear components in the FSC (Badrkhani Ajaei et al., 2009; Graovac et al., 2003; Sakamuri & Yesuraj, 2011; Sanaye-Pasand et al., 2006). Furthermore, study (Mahdi Davarpanah, 2012) has concluded that the active FSC suppress ferroresonance faster than passive FSC. Therefore it has been extensively implemented in CVTs modeling. However, active FSC cannot mitigate fundamental frequency ferroresonance oscillations in CVTs considerably during the auxiliary voltage transformer (AVT) ferroresonance (Mahdi Davarpanah, 2012). Thus, the durable overvoltages due to fundamental frequency ferroresonance may damage the capacitor elements, compensating

reactor in CVT, and the devices connected to the CVT secondary side. On the other hand, passive ferroresonance suppression circuit requires longer time to mitigate ferroresonance compare to active ferroresonance suppression circuit. Recently a new electronic type FSC is proposed in (Sakamuri & Yesuraj, 2011) as ferroresonance mitigation solution in CVT. However, ferroresonance detection system is not designed to operate the circuit during ferroresonance. Furthermore, the circuit configuration is complex and it needs two driver circuits to operate during positive and negative half cycle of line voltage. In contrast, a new electronic FSC design which is simpler will be proposed in this research along with ferroresonance detection method. The proposed FSC design can detect and mitigate all modes of ferroresonance in CVT. In addition, the proposed circuit does not contain any energy storage element. As a result, transient performance of CVT is improved compared to the active and passive FSC based CVT.

1.3 Research Objectives

Considering the importance of improving transient performance and ferroresonance mitigation performance of CVT, the main objectives of this research are as follows:

1. To propose a new ferroresonance suppression circuit and algorithm to detect and mitigate ferroresonance in CVT.
2. To improve transient performance of CVT and analysis of ferroresonance suppression performance and transient response of CVT with different types of ferroresonance suppression circuits.

3. To develop hardware of the ferroresonance detection system and to test using real time digital simulator in order to evaluate the performance of proposed ferroresonance suppression circuit.

1.4 Scope of Research

In order to achieve the above mentioned objectives, following methodology is adopted.

1. Modeling of a 132kV Capacitive Voltage Transformer in EMTP-RV.
2. Initiate ferroresonance in CVT through circuit breaker switching for a typical substation configuration.
3. Propose new ferroresonance mitigation circuit and algorithm to detect and mitigate ferroresonance in CVT – Electronic type FSC
4. Comparison of ferroresonance suppression performance of a CVT while active, passive or electronic type FSC is used in the modeling of a CVT.
5. Construction of a test system in EMTP-RV to evaluate the transient performance of CVT while different type of FSC is used in the modeling of a CVT.
6. Comparison of transient response of CVT while active, passive and the proposed electronic type FSC is used in the modeling of a CVT.

7. Initiating ferroresonance phenomenon in CVT using RTDS and investigation of ferroresonance suppression performance of the proposed technique in the developed hardware.

1.5 Thesis Outline

This thesis report is organized into five chapters. A brief summary of these five chapters is given in this section

Chapter 2: Literature Review

This chapter gives an overview of ferroresonance phenomenon, graphical solution for ferroresonance circuit, characteristic of ferroresonance modes. Ferroresonance occurrence in transformer, causes of ferroresonance, impact of ferroresonance and mitigation of ferroresonance will be discussed in this chapter. The main objective of this chapter is to discuss ferroresonance event in CVT. The modelling of a CVT, initiation of ferroresonance in CVT will be discussed in this chapter. Chapter 2 will also cover the conventional techniques to mitigate ferroresonance in CVT and transient response of CVT.

Chapter 3: Research Methodology

This chapter covers initiation of ferroresonance in CVT due to circuit breaker switching operation. In this chapter, a new ferroresonance mitigation circuit along with ferroresonance detection algorithm is proposed. The main objective of this chapter is to discuss about experimental design of new ferroresonance detection circuit testing. This chapter also gives an overview of real time digital simulator (RTDS) and hardware-in-loop (HIL) testing.

Chapter 4: Results and Discussion

Chapter 4 presents the performance investigation of ferroresonance suppression techniques to mitigate ferroresonance in CVT. A comparison for ferroresonance mitigation performance of the FSC techniques is also made in this chapter. This chapter will cover the transient performance of CVT with different types of FSCs. In addition, this chapter explains the ferroresonance detection algorithm implementation in DSP microcontroller to detect ferroresonance in CVT. Hardware- in –loop (HIL) testing using RTDS is also explained in this chapter.

Chapter 5: Conclusion and Future Work

This chapter presents research conclusion and other proposed future work.

CHAPTER 2: LITERATURE REVIEW

2.1 Ferroresonance Phenomenon

The power community recognized that any disturbances in the fundamental power frequency waveforms of voltage and current will pose a danger risk to the electricity utility's operation. As such, these disturbances have been commonly agreed as power quality problems with the definition - 'Any power problem manifested in voltage, current or frequency deviations that result in failure or misoperation of customer equipment (Dugan, 2003). The main power quality problems include voltage sag, voltage swell, transients and harmonic distortions. Voltage sag has been given the most attention due to the higher frequency of occurrence as well as the huge financial implications compared to other power quality problems (Bollen, 2000; Dugan, 2003). Harmonic distortions are also given special attention due to the implications which include losses, heating damages and mal-operation of power electronic devices capable of halting the entire processing plant (Bollen, 2000; R. C. Dugan, 2003). The interest for the rest of the power quality problems are relatively low. However, it must be noted that ferroresonance incidences, which distorts the voltage sine wave are commonly not given sufficient attention in most power quality literatures. This explains the relatively low level of awareness among the utility engineers on the topic of ferroresonance. Due to the very rare frequency of occurrence coupled with difficulties in detection, utility engineers commonly regard them grossly as transient events due to switching operations, which do not pose any risk to the power equipment. However, several literatures had proven otherwise by establishing a link between equipment failures to ferroresonance (Abbasi Fordoei, Gholami, Fathi, & Abbasi, 2013; Corporation, May 29, 2002; Hassan, Vaziri, & Vadhva, 2011; Lacerda Ribas, Lourenco, Leite, & Batistela, 2013; Moses, Masoum, & Toliyat, 2011).

Ferroresonance phenomenon in electric power systems has been recognized and investigated in numerous technical literatures as early as first decades of the twentieth century (Lacerda Ribas et al., 2013; Hamid Radmanesh & Gharehpetian, 2013). The term was first documented by P.Boucherotin 1920, describing the unusual coexisting operating points and oscillations in a series circuit with nonlinear inductance (Akinci, Ekren, Seker, & Yildirim, 2013; Moses et al., 2011). It has been extensively analyzed with different approaches, spanning nearly a century of accumulated research but it still remains a challenge due the complexity of factors that can lead to the phenomenon (Lacerda Ribas et al., 2013). The occurrence of ferroresonance in electrical power systems can cause energy quality and security problems. Nowadays the occurrence of ferroresonance is more frequent with the growth, expansion and complexity of power systems which can cause subsequent catastrophic damage to electrical equipment affecting the reliability of power networks (Milicevic & Emin, 2013).

According to ANSI/IEEEC37.100 Standard, ferroresonance is defined as "an electrical resonance condition associated with the saturation of a ferromagnetic device, such as a transformer through capacitance" (Lacerda Ribas et al., 2013). This phenomenon generally appears on a series circuit consisting of a nonlinear inductor with abnormal temporary transient behavior. The authors in (Akinci et al., 2013) have explained ferroresonance as a jump phenomenon. This phenomenon is characterized by an abrupt jump from one normal steady-state response to another ferroresonance steady-state response due to a small perturbation introduced to a system parameter. Ferroresonance, is defined in (Moses et al., 2011), as a complex oscillatory interaction of energy exchange between system capacitances and nonlinear magnetizing inductances of ferromagnetic cores. Ferroresonance systems are considered as a nonlinear dynamic system due to nonlinear nature of this phenomenon, thus linear methods cannot be applied to analyze ferroresonance system (Abbasi Fordoei et al.,

2013).

Ferroresonance can be defined generally as a nonlinear series resonance involving a nonlinear inductor in series with a capacitor and excited at or near its natural frequency. This phenomenon occurs when components reach critical values in the series circuit (Bakar, Rahim, & Zambri, 2011). When nonlinear inductance contributed by a saturated transformer matches with the circuit capacitance, the circuit can be subjected to ferroresonance condition. Unlike linear resonance, ferroresonance can occur for a wide range of capacitance values (Ferracci, 1998; Simha & Wei-jen, 2008). When the core of a nonlinear circuit is driven into saturation, the circuit can exhibit multiple values of inductances as the value of a nonlinear inductor is different for current magnitudes above the saturation point. It means that, multiple values of capacitances can potentially lead a circuit into ferroresonance condition at a given frequency (Corporation, May 29, 2002). The main characteristic of ferroresonance phenomenon is that, there can be several possible stable steady state responses for a given configuration and similar network parameters. This phenomenon is characterized by a sudden jump from a stable steady state condition to another ferroresonant steady state condition with harmonic distortion and very high sustained overvoltage that can cause severe damage to network equipment. The authors in (Ferracci, 1998) reported that, initial conditions such as remanent flux in the core of transformers, initial charge on capacitors and switching point on the wave will determine the resultant steady state response.

Ferroresonance circuit connection contains at least an alternating source, a saturable non-linear inductor and a capacitor. The origin of capacitance can typically be from capacitor voltage transformers, shunt and series capacitor banks, reactive power compensation capacitor bank, lumped stray capacitance in transformer windings, circuit breaker grading capacitor, metalclad

substations, bushings, busbars and feeders. The origin of saturable inductor can be formed from single phase or three phase power transformers, shunt reactors and inductive voltage transformers (Giordano et al., 2009; Moses et al., 2011; Sanaye-Pasand et al., 2006; Tseng & Cheng, 2011). In general ferroresonance can occur in the power network consisting of series and shunt capacitances interacting with the magnetizing inductances. A typical single-phase ferroresonance circuit in power network is shown in Fig. 2.1

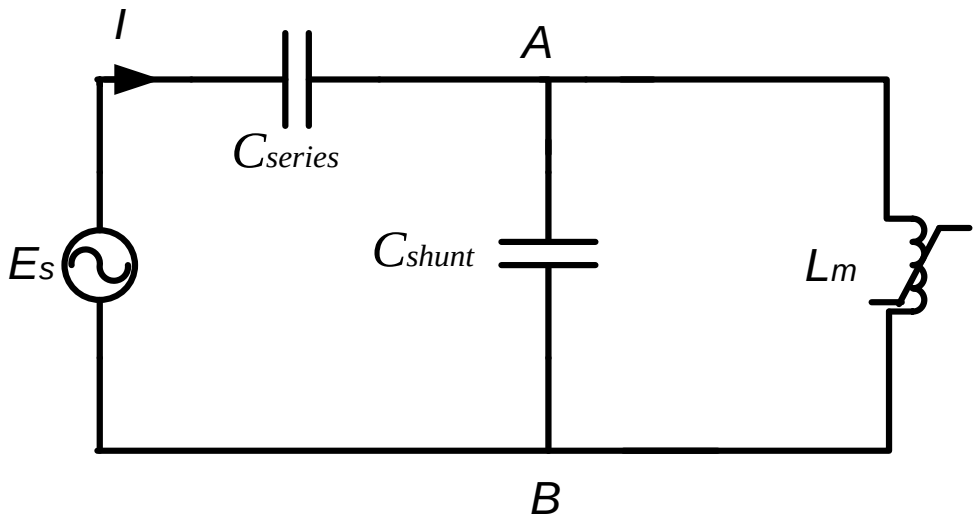


Figure 2.1: Single-phase ferroresonance circuit in power network (Ang, 2010)

2.2. Ferroresonant circuit

For convenient analysis of ferroresonance effect, an LC series equivalent circuit can be simplified from the circuit shown in Fig. 2.1 by using Thevenin's theory. The simplified series ferroresonance circuit is shown in Fig. 2.2 (Ang, 2010).

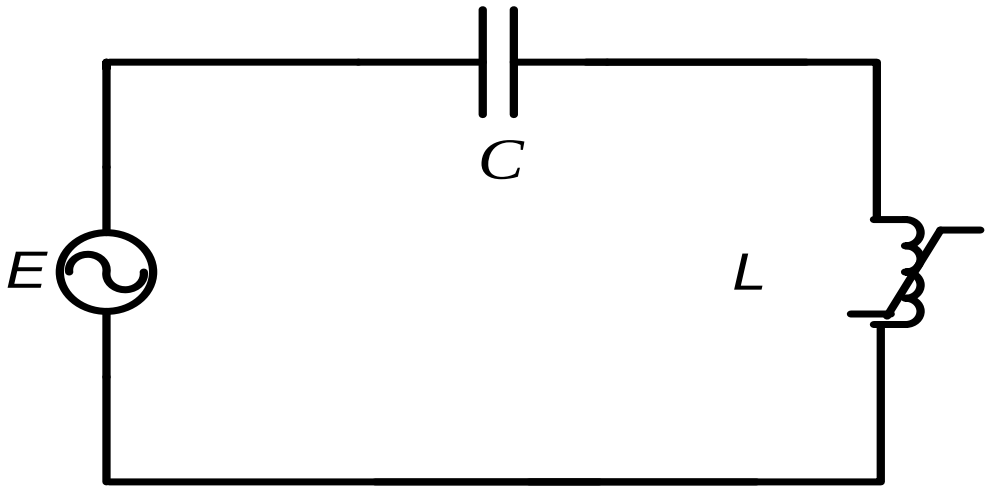


Figure 2.2: Series ferroresonant circuit

The Thevenin's voltage at terminals A-B, is given by Eqn. (2.1) (Ang, 2010).

$$E = E_s \frac{C_{series}}{C_{series}+C_{shunt}} \tag{2.1}$$

Thevenin's capacitance at terminals A-B, is given by Eqn. (2.2)

$$C = C_{series} + C_{shunt} \tag{2.2}$$

The equation for the series ferroresonance circuit as shown in Fig. 2.2 is given as (Ta-Peng & Chia-Ching, 2006)

$$V_L = E + V_C \tag{2.3}$$

Where

$$V_C = -\frac{1}{C}\int i(t)dt \tag{2.4}$$

$$V_L = L\frac{di(t)}{dt} \tag{2.5}$$

Authors in (Shein, Zissu, & Schapiro, 1989; Ta-Peng & Chia-Ching, 2006) demonstrated a series ferroresonance circuit in a graphical solution. Since the capacitive reactance is linear, its slope gives the capacitive reactance which is plotted as a straight line in Fig. 2.3. The straight line represents the V-I characteristic of capacitor across the saturable inductor. On the other hand the inductive reactance is represented by the saturation curve of a magnetic iron core. The curve V_L represents the V-I magnetizing characteristic of the core. The possible operation points of the ferroresonance behavior are the intersections of the straight line and transformer's magnetization curve. The straight line rotates around the point "p", so its slope decreases as the capacitance is increased. As it can be seen in Fig. 2.3, unlike resonance state, ferroresonant state is possible for a wide range of capacitance values at a given frequency (Corporation, May 29, 2002). For a given value of capacitance, there are three possible operating points of this circuit.

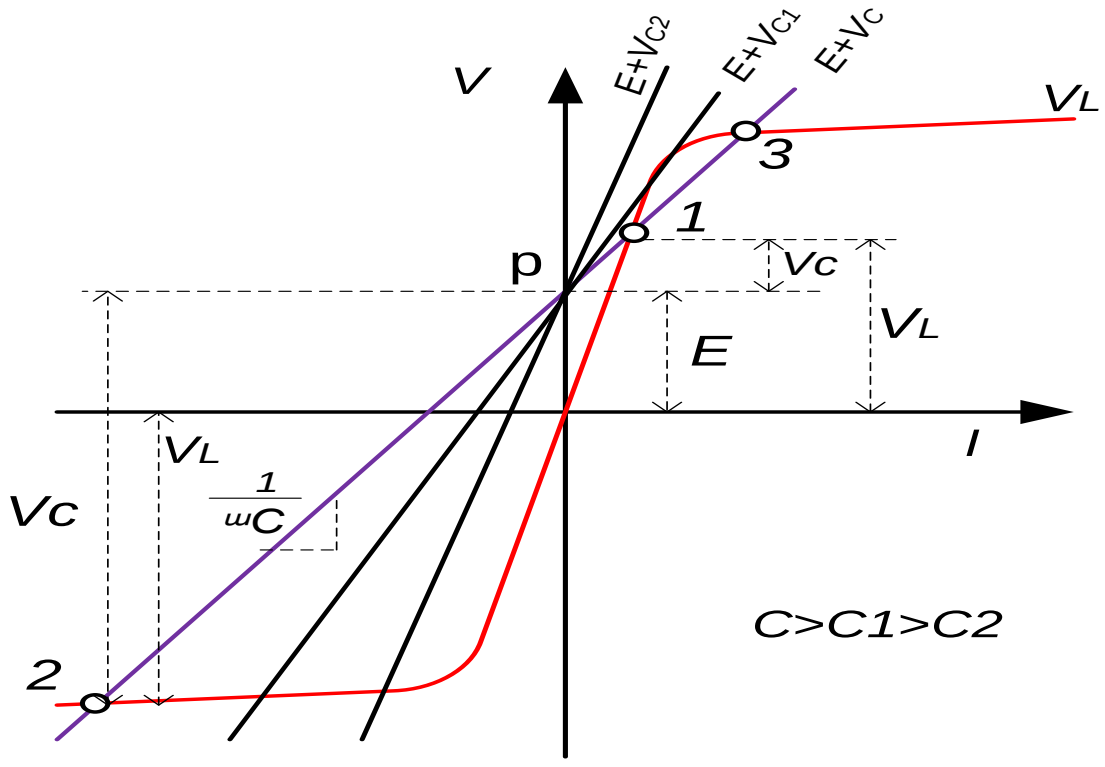


Figure 2.3: Graphical view of series ferroresonance circuit (Ta-Peng & Chia-Ching, 2006)

Intersection point “1” corresponds to normal operation in the linear region of the magnetizing characteristics of the saturable inductor, where excitation current and flux are within the design limit. At this point the inductor voltage (V_L) is higher than capacitance voltage (V_C). It is a non-ferroresonant stable operation point where steady state voltage appears across the inductor terminals. Intersection point “2” is a ferroresonant stable operation point in the saturation zone where V_L is lower than V_C. This is a capacitive situation, ($X_{L\text{-}sat} < X_C$), where the flux densities and excitation current are beyond the design limit. In contrast, the intersection point “3”, in the positive region, is unstable operating point since the source voltage increase follows a current decrease, thus violating the equilibrium theory (Shein et al., 1989; Ta-Peng & Chia-Ching, 2006).

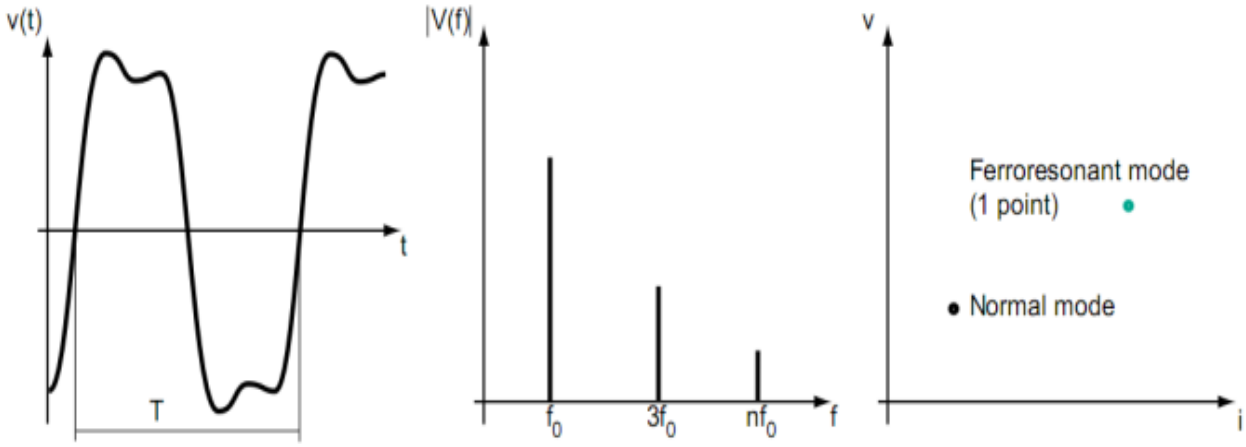
The key elements of ferroresonance phenomenon are excitation of saturable inductor connected in series with capacitor. Inductance (X_L) of a nonlinear inductor is usually related to the magnetization curve of iron core of the inductor. Inductance of a coil is proportional to the slope of the magnetization curve, which indicates that the inductance has high value before the saturation point and the inductance of the coil changes rather suddenly to lower value beyond the saturation point as the voltage in the ferromagnetic coil increases. Under normal operating condition, capacitive reactance (X_C) is smaller than inductive reactance (X_L). However, any switching event in power system may cause the voltage to increase across a transformer, which may push the transformer core into saturation and inductance (X_L) is lowered. This saturated inductive reactance (X_L) may equal capacitive reactance (X_C) of the system capacitance, which will form a series resonant circuit known as ferroresonance (Jazebi, Farazmand, Murali, & de Leon, 2013; Milicevic & Emin, 2013).

2.3 Ferroresonance modes

As a result of the reconfiguration of a circuit into a ferroresonance circuit, the system will jump from one normal steady state condition to another ferroresonant steady state condition. There are several steady state responses for a given circuit due to nonlinearity of ferroresonance circuit and may cause an abrupt jump between two different steady states, following a transient or small variations of a system parameter values. Based on numerical simulations, experiments and waveforms captured in the power systems, ferroresonance circuit can possibly have four types of steady-state responses (Ferracci, 1998). They are the fundamental mode, subharmonic mode, quasi-periodic mode and chaotic mode. Fast Fourier Transform (FFT) and Poincarè map are normally used to illustrate the different categories of ferroresonance modes. All possible modes of ferroresonance and its appearances are shown in Fig. 2.4 to Fig. 2.7 (Ferracci, 1998).

2.3.1 Fundamental mode

The signals (voltage and current) have a distorted waveform, but periodic waveforms with the same period as the power system. The signal spectrum is a discontinuous spectrum consists of the fundamental frequency of the power system and followed by its harmonics (2^{nd}, 3^{rd}.... n^{th}) (Ferracci, 1998). The fundamental mode is shown in Fig. 2.4.

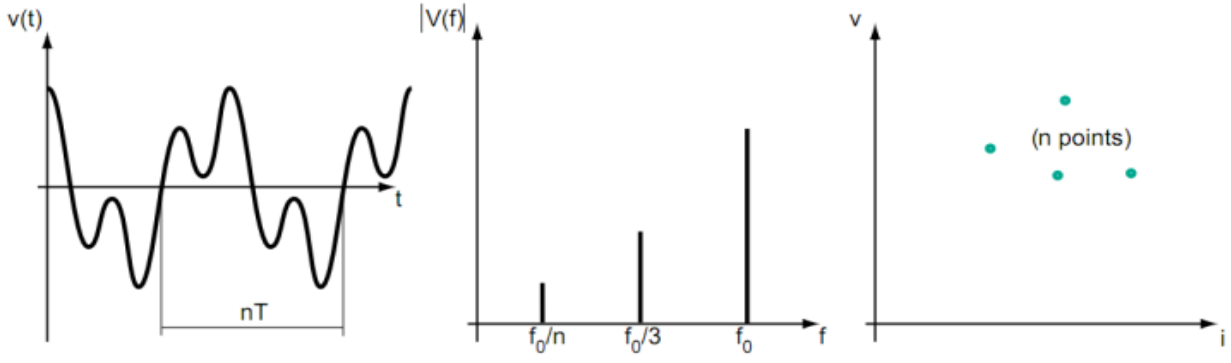


(a) Periodic signal (b) Frequency spectrum (c) Stroboscopic diagram

Figure 2.4: Fundamental mode ferroresonance (Ferracci, 1998)

2.3.2 Sub-harmonic mode

The signals are periodic, with a period multiple of the source period. The frequency contents are described having a spectrum of frequencies equal to f_0/n (where f_0 is the source frequency and n is an integer) (Ferracci, 1998). The sub-harmonic mode is shown in Fig. 2.5.

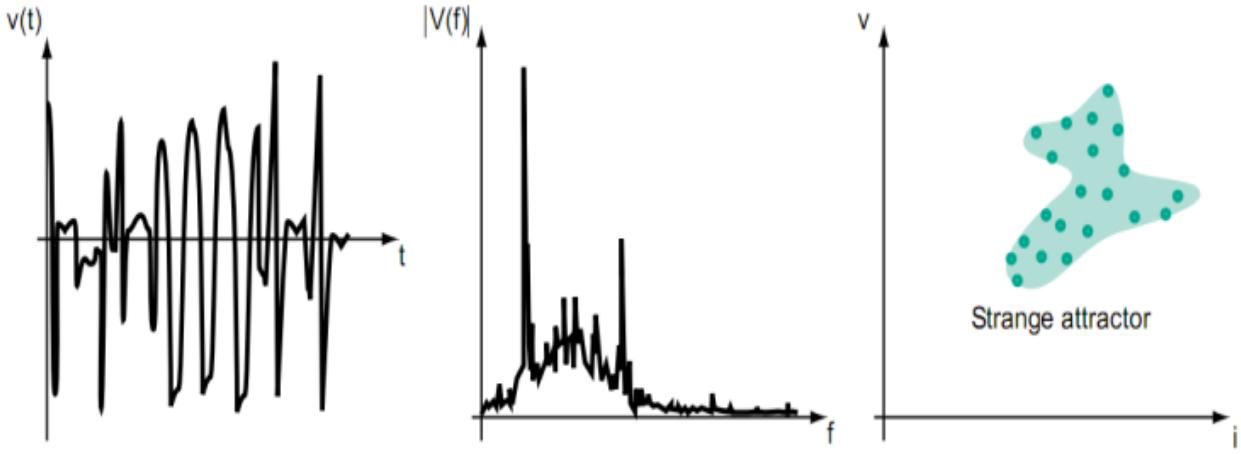


(a) Periodic signal (b) Frequency spectrum (c) Stroboscopic diagram

Figure 2.5: Subharmonic mode ferroresonance (Ferracci, 1998)

2.3.4 Chaotic mode

In chaotic mode, the signals show an irregular and random behavior. This mode has a signal showing non-periodic frequency spectrum and the corresponding spectrum is continuous (Ferracci, 1998). The chaotic mode is shown in Fig. 2.6.

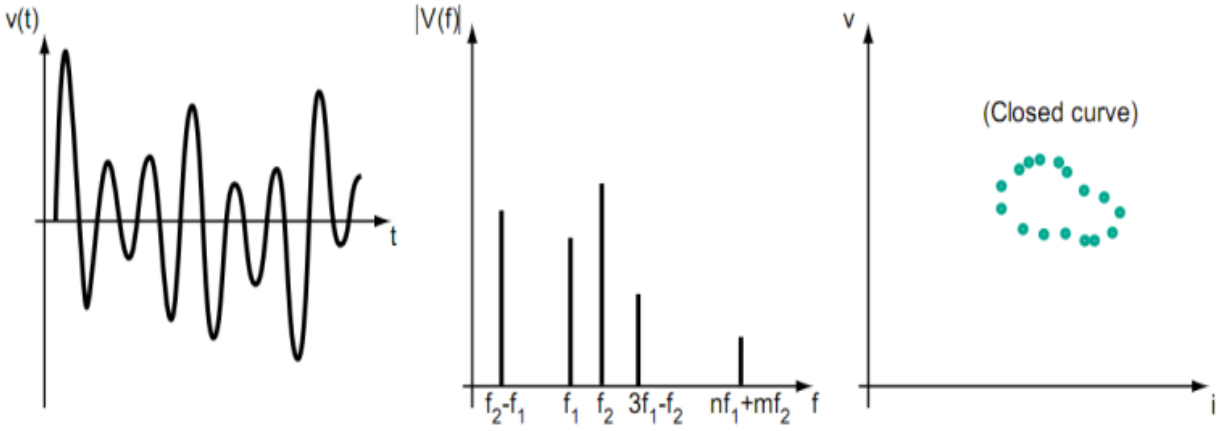


(a) Periodic signal (b) Frequency spectrum (c) Stroboscopic diagram

Figure 2.6: Chaotic mode ferroresonance (Ferracci, 1998)

2.3.3 Quasi-periodic mode

Quasi-periodic mode is also known as pseudo-periodic mode. In this mode, the signals are non-periodic waveforms with a discontinuous frequency spectrum, whose frequencies are expressed in the form: nf_1+mf_2 (where m and n are integers and f_1/f_2 an irrational real number) (Ferracci, 1998). The quasi-periodic mode is shown in Fig. 2.7.



(a) Periodic signal (b) Frequency spectrum (c) Stroboscopic diagram

Figure 2.7: Quasi-periodic mode ferroresonance (Ferracci, 1998)

Several methods have been proposed for the analysis of ferroresonance, such as the bifurcation theory, nonlinear theory, fussy theory, chaos theory and Newton Raphson algorithm (Kieny, 1991; Liu, Li, Liu, Cao, & Zeng, 2011; Mork & Stuehm, 1994; Naidu & De Souza, 1997; Valverde, Buigues, Fernandez, Mazon, & Zamora, 2012). The four different modes of ferroresonance have been investigated by numerous authors. Various plots have been used to illustrate the ferroresonance modes such as poincare´ map, bifurcation diagram, hysteresis formations and phase-plane trajectories (Moses et al., 2011). The different ferroresonance modes in electrical power systems have been investigated in (Hamid Radmanesh & Gharehpetian, 2013), and it was reported that ferroresonance modes are sensitive to core saturation and variations in the control parameter values. In the simulation the capacitance of the system was ramped both upward and downward by the authors to simulate different ferroresonance modes. In contrast, the authors in (Ben Amar & Dhifaoui, 2011) demonstrated that various ferroresonant modes can be achieved by changing the physical parameters of the network such as line length, impedance and supply voltage. Besides that, (Akinci et al., 2013; Saravanaselvan & Ramanujam, 2012) reported that ferroresonance is highly sensitive to the

change of initial conditions and operating conditions, while the authors in (Lacerda Ribas et al., 2013; Lamba, Grinfeld, McKee, & Simpson, 1997; Ferracci, 1998) reported that ferroresonant modes are sensitive to the remanent magnetic flux in a core of a power transformer, initial charge on capacitors and switching instant. The authors in (Radmanesh, Hosseinian, & Fathi, 2012; Saravanaselvan & Ramanujam, 2012) investigated the influence of iron core saturation characteristics on the occurrence of harmonic modes and observed that highly saturated iron core increases the possibility of chaotic mode. On the other hand, the behavior of chaotic ferroresonance has been reported in (Abbasi Fordoei et al., 2013) to depend on the voltage source amplitude, core loss, initial condition, capacitance and resistance of a system. The authors in (Rezaei-Zare, Iravani, & Sanaye-Pasand, 2009) reported that, type of ferroresonance oscillations depends on magnetization characteristic and the core loss.

2.4 Causes of Ferroresonance

It has been addressed from previous sections that the trigger mechanism for ferroresonance is switching events that reconfigure a circuit into ferroresonance circuit. Typical arrangement of these circuits in power system comprises of an excited grading capacitance of a circuit breaker in series with an unloaded inductive voltage transformer. Unbalanced switching with series and shunt capacitances increases the risk of ferroresonance. Besides that, basic circuit parameters have a great impact to the initiation of characteristic ferroresonance states. The degree of influence of the supply voltage, the losses and the circuit capacitance on the ferroresonance phenomenon has been studied in (Ben Amar & Dhifaoui, 2011). Based on measurements and simulation results in (Milicevic & Emin, 2013) the authors observed that ferroresonance occur at higher values of rms source voltage as the coil nominal voltage of

transformer is increased. The authors in that paper also reported the possibility of ferroresonance initiation increases if circuit capacitance increases, because higher capacitance increases the impact of initial conditions on the occurrence of ferroresonance. The influence of circuit parameters, magnetic losses, and supply conditions on the development of ferroresonance has been investigated in (Barbisio, Bottauscio, Chiampi, Crotti, & Giordano, 2008).

The authors in (Lacerda Ribas et al., 2013) described several events leading to ferroresonance occurrence in power systems, which include asymmetrical phase switching, incidence or removing of faults, energization or de-energization of transformers or inductive elements, and manual or automatic single-phase switching. (Hamid Radmanesh & Gharehpetian, 2013) (Shipp, Dionise, Lorch, & MacFarlane, 2011) reported that switching operation or phase opening causes ferroresonance oscillations in power networks. Several case studies have also been performed for different combinations of the switches at the end of the transmission line in (Akinci et al., 2013) and it was reported that removing loads can cause ferroresonance phenomena. Ferroresonance experienced in (Val Escudero, Dudurych, & Redfern, 2007) was due to the switching events that have been carried out during the commissioning of a new 400-kV substation. On the other hand (Pattanapakdee, 2007) reported that ferroresonance occurred due to the switching operations by first opening the circuit breaker, followed by opening the disconnector switch located at the riser pole surge arrester in a station service transformer of a 12-kV substation. The researchers in (McDermit, Shipp, Dionise, & Lorch, 2013; McDermit, Shipp, Dionise, & Lorch, 2012) reported that, several potential transformers failed catastrophically due to occurrence of ferroresonance associated with opening and closing the vacuum circuit breakers.

2.5 Impact of Ferroresonance

Ferroresonance occurrence in electrical power systems can cause undesirable effects on power system components. (Lacerda Ribas et al., 2013) reported that the presence of ferroresonance phenomena can lead the magnetic apparatus to a catastrophic operating condition. It was reported that the implication of ferroresonance include problems in protection systems, overheating in transformers and reactors, and excessive sound which would lead to equipment explosion. It can also cause thermal danger to insulators as well as problems in transmission and distribution systems (Abbasi Fordoei et al., 2013). Ferroresonance in a power system can also result in misoperation of protective devices (Corporation, May 29, 2002). The authors in (Moses et al., 2011) also reported that ferroresonance oscillation can cause distorted over voltage and currents in power networks, leading to excessive heating and insulation failure in transformers. The oscillation that operates a solid dielectric system above its normal stress level for an extended period can shorten equipment lifespan. Ferroresonance can also cause surge arrester failure due to thermal heating and it is a common victim in power system (Hassan et al., 2011). The authors in (Tanggawelu, 2003) reported several incidents of overvoltages in Malaysian distribution network, which led to equipment damages. The overvoltages are then traced to ferroresonance as well.

2.6 Mitigation of Ferroresonance

The initiation of ferroresonance phenomena is of special importance to power network utilities due to the catastrophic impacts of over voltages and over currents on the electrical equipment. Various methods have been proposed to suppress ferroresonance in power networks. Most of

the methods include avoiding switching operations that reconfigure a circuit into ferroresonance circuit and introducing burden such as resistive losses into the affected transformer to damp the ferroresonance.

Mitigation of ferroresonance inside a voltage transformer has been investigated through simulations, laboratory and field tests in (Huang & Hsieh, 2013). In the proposed strategy, a group of resistors were inserted in parallel and then the resistors are detached step by step. This strategy successfully suppressed ferroresonance oscillations. The researchers in (McDermit et al., 2013; McDermit et al., 2012) experimentally proved that, implementation of a snubber circuit can greatly reduce the transient overvoltage and oscillation at the primary winding of potential transformer and power transformer. They verified the analysis using high speed switching transient measurements and proved the effectiveness of snubber circuit and arresters. The snubber circuit can damp out the oscillation in potential transformer by acting as a damping source. It was reported that the resistor in the snubber circuit successfully mitigated the ferroresonance to within acceptable levels. The authors in (Tseng & Cheng, 2011)verified the effectiveness of a ferroresonance mitigation technique by simulation and field tests for potential transformers. Successful ferroresonance mitigation was implemented with damping reactors and the maximum sustained ferroresonance duration was just up to 3.9s. To control these oscillations the authors in (Shein et al., 1989) inserted temporary damping resistors in the secondary of a voltage transformers. A one ohm damping resistor was connected in parallel to the secondary and it was reported that this technique is a reliable solution for suppressing sustained ferroresonance. (Piasecki, Florkowski, Fulczyk, Mahonen, & Nowak, 2007) reported a new method of protecting the voltage transformers against ferroresonance. The researchers developed a compact active load which was connected to the open-delta arranged auxiliary windings and observed that ferroresonance initiated by cable switching was

damped out within 0.2s. (Huang & Hsieh, 2013) discussed the effect of bus capacitance on ferroresonance occurrence and observed that this oscillations can be avoided by increasing the bus capacitance. The authors in(Saravanaselvan & Ramanujam, 2012) also reported that high capacitance to ground can be the subharmonic solutions of a de-energized line. To mitigate ferroresonance oscillations inside a potential transformer, the researchers in (Li, Han, & Zhang, 2012) inserted zero sequence resistance to the secondary coil and suppression of ferroresonance was successfully verified under two working states via a practical gas insulated substation.

2.7 Ferroresonance in CVT

Knowledge on ferroresonance occurrences has been improved since its discovery. Today, ferroresonance sources, impacts and suppression techniques are well documented in various literatures. However, almost all of these knowledge are related to ferroresonance in power transformer. Although some authors reported CVT failures due to ferroresonance incidences, the knowledge remains relatively shallow and unexplained. CVTs are widely used throughout the high voltage and extra high voltage power system to transform the line voltages to designated low voltages levels through a sequence of a capacitive potential divider circuit and a step down transformer (Ajaei & Sanaye-Pasand, 2008; Costello & Zimmerman, 2012; Lucas, McLaren, Keerthipala, & Jayasinghe, 1992; Siregar & Setiawan, 2012). The output signal of a CVT is used for monitoring, controlling the high voltage system and as input sources to protective relays to preserve power system stability and minimize damage to equipment. The performance of relays relies on the signals produced by the CVTs. The signals produced by the CVTs may not exactly track the power system voltages and currents due to the internal energy storage elements and magnetic saturation of nonlinear components of CVT. The output signal of a CVT depends on the transient response of the CVT and different

type of power system transient (Bakar et al., 2011; Zare, Mirzaei, & Abyaneh, 2012). In steady state, the secondary voltage of a CVT mirrors the primary voltage. However, under transient conditions caused by faults, lightning, capacitive switching and system energization, the transient output is no longer a replica of the primary (Graovac et al., 2003; Zare et al., 2012).

Due to transients in power system, voltage changes abruptly either in the primary or the secondary system of CVT that can saturate the step down transformer core. Ferroresonance oscillations may initiate if the system capacitances resonate with the value of saturated non-linear inductance. These oscillations can make noticeable deviation of CVT response and transfer undesired information to protective relays. This phenomenon generates high voltage and current across the components and causes a CVT to explode (Badrkhani Ajaei et al., 2009; Fernandes Jr, Neves, & Vasconcelos, 2007; Graovac et al., 2003; Lucas et al., 1992; Sakamuri & Yesuraj, 2011).

2.7.1 CVT structure

A generic CVT consist of a capacitive voltage divider (CVD), compensating reactor (SR), a step down transformer (SDT) and ferroresonance suppression circuit (FSC). The function of capacitor voltage divider is to step down the line voltage to designated voltage level and it is typically 5kV to 15kV. This voltage level is further reduced to relaying voltage level through a step down transformer. The function of the compensating reactor is to prevent any phase shift between primary and secondary voltages due to capacitive divider network. Compensating reactor cancels the capacitive reactance contributed by capacitive divider network at the system frequency (Costello & Zimmerman, 2012; Daqing & Roberts, 1996; Davarpanah, Sanaye-Pasand, & Badrkhani Ajaei, 2012; Zare et al., 2012). Fig. 2.8 shows a schematic

diagram of the arrangement (Bakar, 2011). Table II shows the parameters of the CVT diagram based on the CVT parameters proposed in (Bakar, 2011).

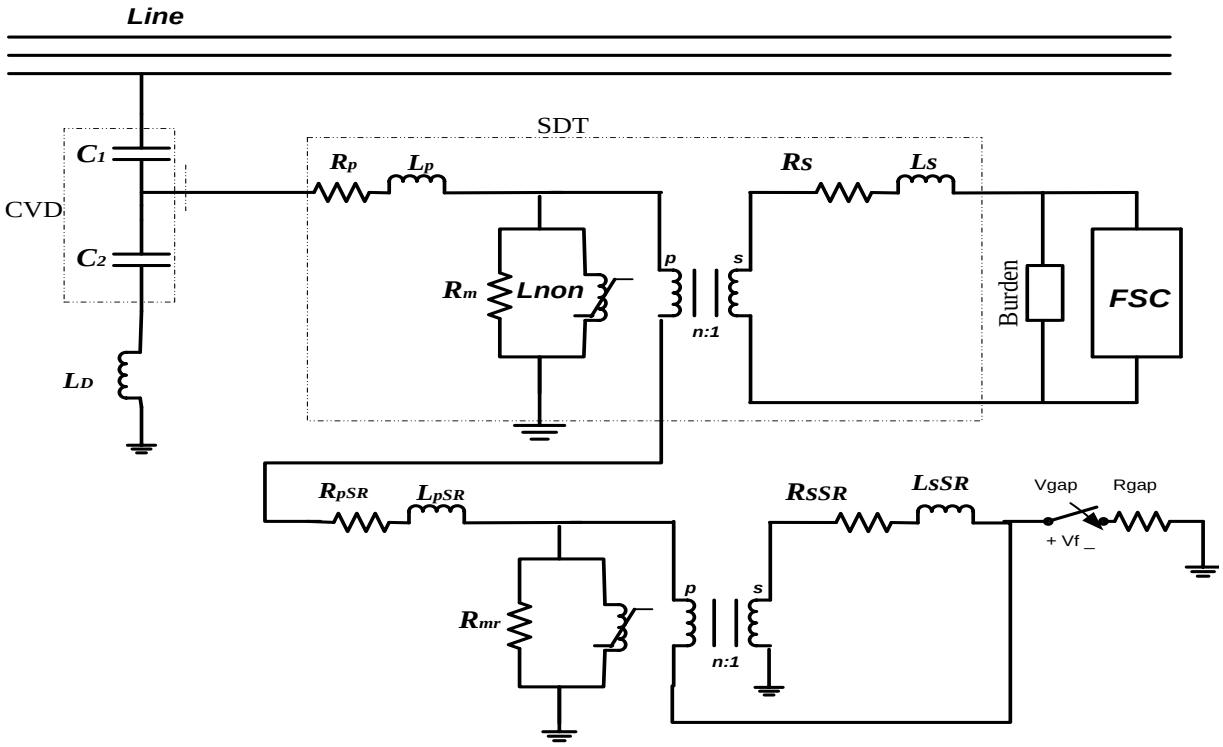


Figure 2.8: Schematic diagram of CVT (Bakar, 2011)

Table 2.1: Data description for capacitive voltage transformer

Description	Parameters
System voltage	132 kV
Frequency	50 Hz
Capacitive voltage divider	*C1* = 5348.8 PF *C2* = 76666.6PF Drain coil [*LD*] = 10 mH
Step down transformer (SDT)	*Rp* = 220 Ω *Lp* = 1.745H *Rm* = 6500000 Ω *Rs* = 0.04 Ω *Ls* = 0.007 mH *Trans ratio* = 78.74
Compensating series reactor (SR)	*RpSR* = 220 Ω *LpSR* = 1.745H *RsSR* = 6500000 Ω *LsSR* = 8841H *Rmr* = 0.04 Ω Trans ratio = 28

2.7.2 Imposing ferroresonance in CVT through simulation

To study the ferroresonance events in CVT, ferroresonance have been intentionally imposed on the CVT in simulation software (Bakar, Lim, & Mekhilef, 2006; Graovac et al., 2003; Sakamuri & Yesuraj, 2011). The authors in (Bakar et al., 2006) imposed ferroresonance by short circuiting the secondary of CVT and then opening the switch after 1 second, while the authors in (Graovac et al., 2003) open the switch after 7 cycles. In contrast, ferroresonance is initiated by the authors in (Sakamuri & Yesuraj, 2011) based on the test recommended by IEC60044-5 ("IEC International Standard on Instrument transformers Part 5: Capacitor Voltage Transformers," 2004). This recommendation includes temporary short circuiting the secondary side of the step down transformer for a maximum period of 100ms and opening the shorted terminal while the CVT is kept energized. Simulation studies are conducted to identify

the modes of ferroresonance in the EMTP-RV environment using the CVT configuration and parameters adopted from (Bakar et al., 2011). The circuit arrangement in EMTP software is shown in Fig. 2.9. The magnetic behavior of the transformer core of CVT is represented by a non-linear inductor (*Lnon*) to model the saturation effect.

Short-circuit across secondary winding is imposed by a switch SW in series with a resistor R. Switch SW and series resistor R are not part of the CVT system. They are introduced in the model to impose transients on the CVT and to initiate ferroresonance oscillation inside the CVT.

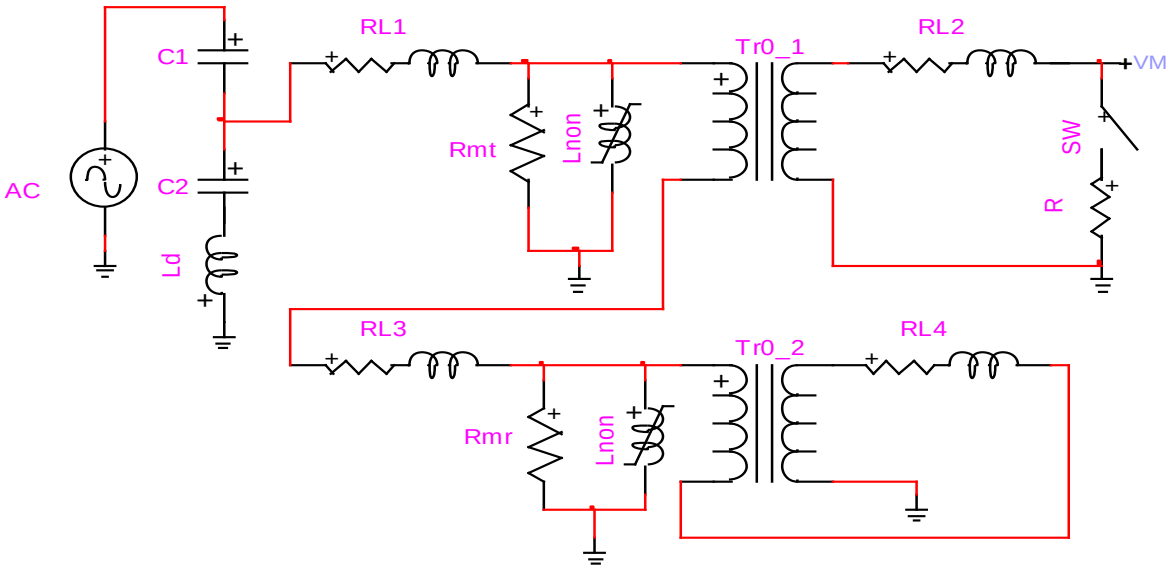
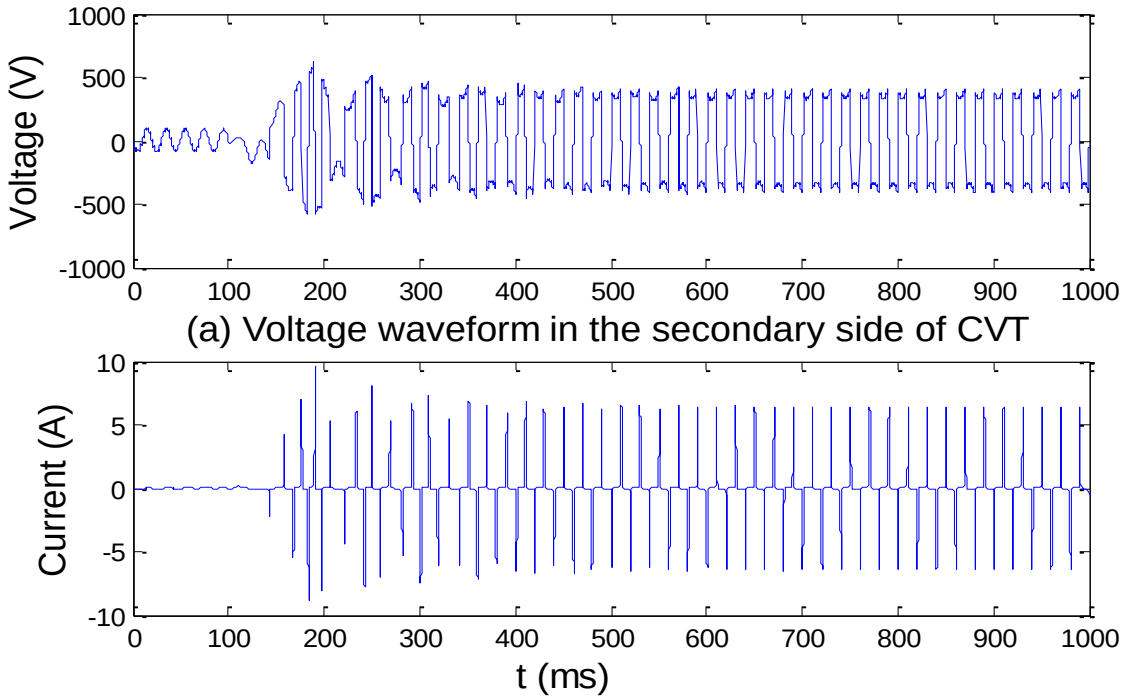


Figure 2.9: Ferroresonance test circuit in EMTP software

To initiate ferroresonance in CVT, step down transformer magnetic core need to be saturated. Remanent flux in the step down transformer core contributes to the core saturation. The switch

is connected at the instant of maximum core flux and disconnected when primary voltage across the step down transformer is at minimum value. Following the disconnection of switch, the step down transformer core is saturated and ferroresonance is initiated in the CVT.



(b) Current waveform in the primary side of CVT

Figure 2.10: CVT secondary voltage and primary current waveform

The voltage and current waveforms in the CVT produced from the simulation is depicted in Fig. 2.10. The frequency spectrum for the CVT voltage of Fig. 2.10 is shown in Fig. 2.11. It is known as the sustained fundamental ferroresonant mode. It resonates at 50 Hz frequency with a sustainable amplitude of 4 per unit. The magnitude of this kind in CVT is of serious concern because of possible damage to the CVT. In addition, the frequency content consists of the fundamental frequency component as well as the existence of higher order frequency components such as 3^{rd}, 5^{th} and 7^{th} harmonics.

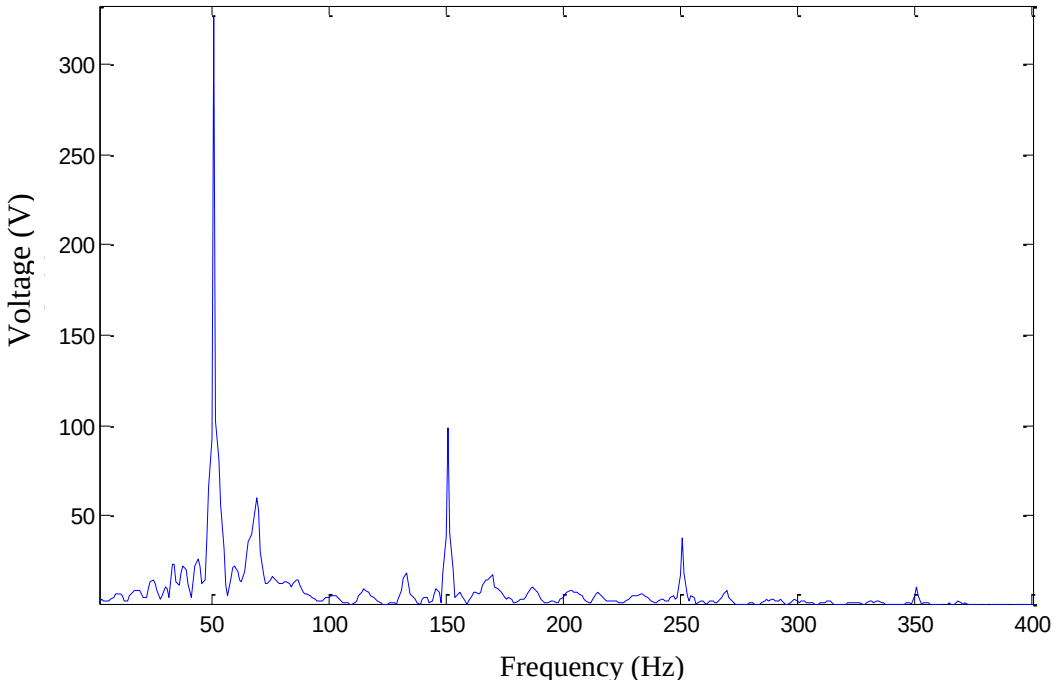


Figure 2.11: FFT for voltage waveform in the secondary side of CVT

2.7.3 Mitigation of ferroresonance in CVT

To avoid or damp out ferroresonance in capacitor voltage transformers, various FSC had been proposed. The ferroresonance suppression methods including saturable reactor type FSC, fundamental frequency blocking filter type FSC and electronic type FSC have been studied and their performances to suppress ferroresonance have been investigated by numerous authors. The frequency domain analysis showed that saturable reactor type FSC is more desirable compared to resonance filter type FSC because CVT frequency response is adversely affected by the resonance filter and causes error in the output signal in the case of higher order harmonics or fast changes in the system voltage (Shahabi, Shirvani, and Purrezagholi, 2009; Sanaye-Pasand et al., 2006). On the other hand, the time domain simulation showed that electronic type FSC with damping resistor is more effective to damp out ferroresonance than

other methods (Shahabi, Gholami, andTaheri, 2009; Sakamuri & Yesuraj, 2011). Also, the transient response of CVT with electronic type FSC is much better than the other two FSCs. The authors in (Shahabi, Shirvani, and Purrezagholi, 2009) also reported that the addition of a surge arrester will significantly reduce the damping time and limit the overvoltage which is experienced in CVT during the first cycle. The impacts of active FSC, passive FSC and over voltage protection devices for fast CVT ferroresonance suppression have been studied in (Badrkhani Ajaei et al., 2009). It was reported that the presence or the absence of over voltage protection devices, the active FSC can mitigate ferroresonance within 2 cycles which is faster than the passive FSC and the output fidelity of active FSC based CVT is less dependent on the burden as compared to passive FSC. The authors in (Graovac et al., 2003) reported that properly tuned triac/spark-gap over voltage protection is capable of mitigating the ferroresonance within two source period. The authors also investigated the performance of passive ferroresonance suppressor where metal oxide varistors (MOV) was implemented as a part of the protection and it was reported that ferroresonance mitigation capability may not be as effective compared to triac/spark-gap type protection but it improves CVT ferroresonance response noticeably.

2.7.3.1 Active ferroresonance suppression circuit

The active FSC is also known as power frequency blocking filter. It consists of inductors *La1* and *La2* with mutual coupling of *Ma,* a capacitor *Ca,* a damping resistor *Ra.* The filter is tuned to the power system fundamental frequency with a high Q factor. The Active FSC acts like a band-pass filter. The filter circuit shows high impedance at the fundamental frequency. When the frequency deviates from the fundamental frequency of power system, the impedance of the

active FSC gradually approaches the resistance of the damping resistor. The ferroresonance oscillations are damped using the resistor *Ra* in the active FSC. Fig. 2.12 shows the circuit diagram of active FSC. The impedance characteristic of active FSC is shown in Fig. 2.13 (Badrkhani Ajaei et al., 2009; Graovac et al., 2003; Sanaye-Pasand et al., 2006).

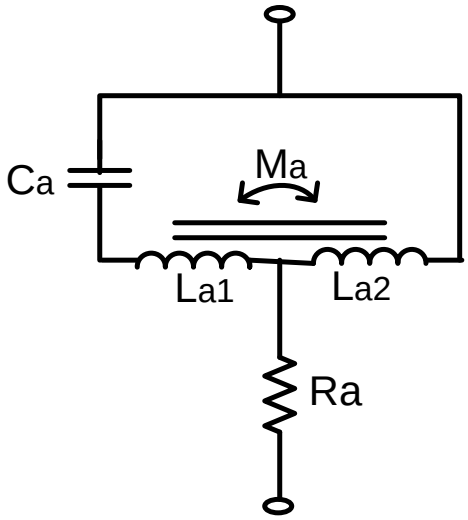


Figure 2.12: Circuit diagram of active FSC.

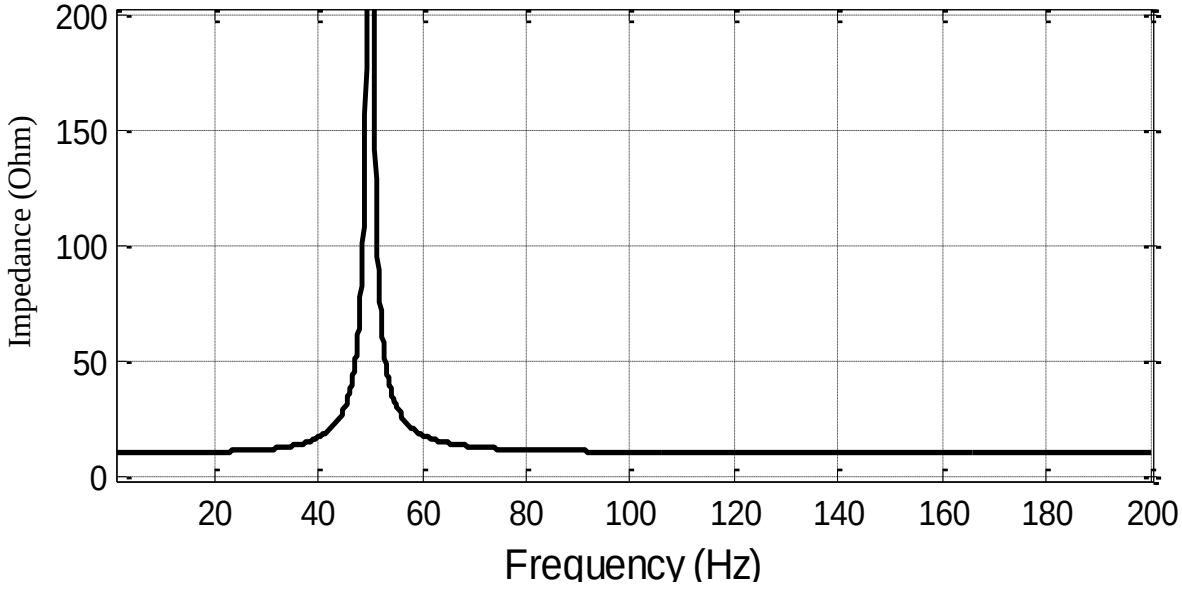


Figure 2.13: Impedance characteristic of active FSC

2.7.3.2 Passive Ferroresonance Suppression Circuit

The passive ferroresonance damping circuit consists of a loading resistor *Rr* which is connected permanently in parallel with a saturable reactor (*Rn* and *Ln*). To mitigate a sustained ferroresonance condition, the saturable reactor *Ln* is designed to saturate at about 150% of the normal voltage. The over voltage during ferroresonance saturates the reactor *Ln.* As a result, the series resistance *Rn* effectively adds additional load and mitigate the ferroresonance oscillation. Fig. 2.14 shows the circuit diagram of passive FSC. The passive FSC impedance versus voltage characteristic is shown in Fig. 2.15 (Badrkhani Ajaei et al., 2009; Graovac et al., 2003; Sanaye-Pasand et al., 2006).

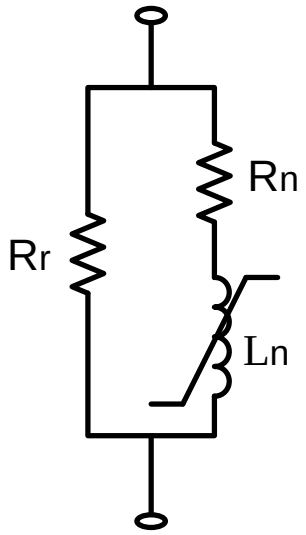


Figure 2.14: Circuit diagram of passive FSC.

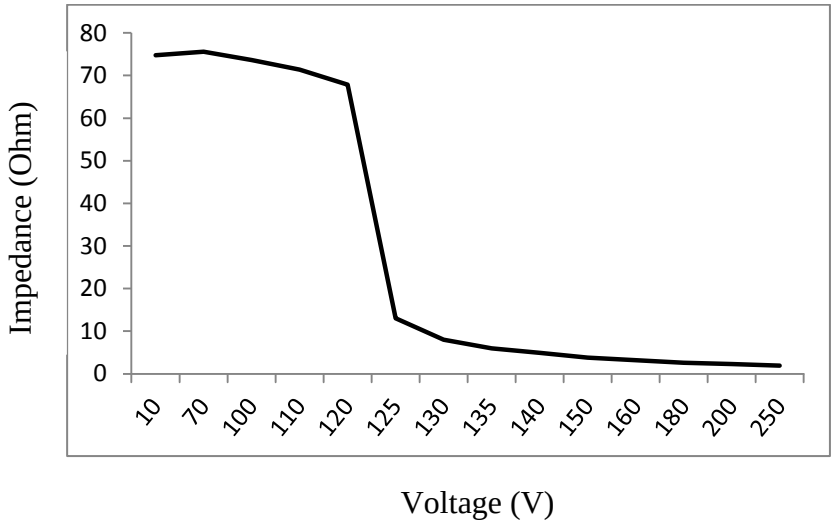


Figure 2.15: Passive FSC impedance magnitude versus voltage.

2.7.3.3 Electronic Ferroresonance Suppression Circuit

The third approach is to use the electronic FSC proposed in (Sakamuri & Yesuraj, 2011) where two switches in a circuit controls the connection of resistor across the CVT secondary. The configuration consists of a damping resistance R_d, two back-to-back thyristor. During ferroresonance or transient condition the switch is turned on for a fixed time interval. If ferroresonance still exits then the switch is kept switched on until ferroresonance duration is damped out. This technique does not contain any bulky inductor or capacitor, which allows reduction in its size. Fig. 2.16 shows the circuit diagrams for a typical electronic FSC. However ferroresonance detection technique is not explained in that paper and did not perform any real time test to prove the effectiveness to mitigate ferroresonance in CVT. In addition, a total of two external gating circuits are required in this FSC design.

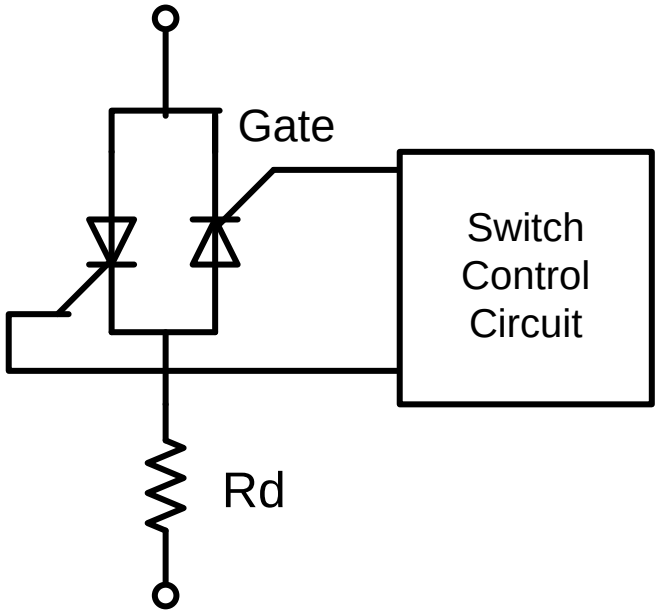


Figure 2.16: Conventional electronic ferroresonance suppression circuit (Sakamuri & Yesuraj, 2011)

2.8 Transient Response of CVT

The transient response of a capacitive voltage transformer is the ability to reproduce rapid changes in the primary voltage. It's defined as the remaining secondary voltage after a specific time due to a short circuit on the primary voltage (Hedding, 2012).

CVT consists of a capacitive voltage divider network, a step down transformer and other connected equipment. Series and parallel connection of capacitor with the power system inductances can form resonant circuit and leads to RC time constant that can cause error in reproducing transmission voltage on the CVT secondary terminal. The CVT transient will not create any problem for the operation of electromechanical relays. However, transient is a problem for solid state and microprocessor relays and needed attention. CVT transient response is different compare to inductive voltage transformer because CVT circuit consists of

energy storage elements such as inductors and capacitors. When a short circuit occurs on the primary circuit, the stored energy in the energy storage elements will discharge and create transient oscillation on the secondary terminal of CVT. This transient oscillation consists of high frequency and low frequency component. The high frequency component can be from 600Hz to 4 kHz and damp out within 10ms. However, low frequency component can be from 2 Hz to 15 Hz and can exist for longer period. This transient response of CVT can create problem for distance relay operation because it depends on the CVT secondary voltage to generate tripping decisions.

During fault condition, if the calculated impedance by distance relay is within its reach setting then it will generate a tripping decision. Therefore, the mirror of the transmission line voltage is necessary from CVT secondary for a distance relay for impedance calculation. CVT transients cause incorrect information to be presented to the relay for a short period of time. Since zone 2 and zone 3 timers are much longer than the CVT transient period, zone 2 and zone 3 elements are not affected by CVT transients. Zone 1 elements operate with no intentional delay. Therefore, their operation is affected by the CVT transient. Several factors influence the transient response of a CVT. The factors that influence the transient response of a CVT are the equivalent capacitance of the stack, the tap voltage, the connected burden, and the type of ferroresonant suppression circuit (Hedding, 2012).

CHAPTER 3: RESEARCH METHODOLOGY

3.1 Initiation of Ferroresonance in CVT

Ferroresonance in power transformer have been observed for many years. Simulation have been conducted for ferroresonance in CVT due to circuit breaker (CB) switching in a typical substation configuration. The network diagram as shown in Fig. 3.1 has been adopted for this study. Supply comes from 132kV transmission line and is connected to a circuit breaker, which is subsequently connected to a busbar with 3 outgoing feeders. A bus CVT is connected to the busbar for voltage measurement. Simulations for circuit breaker switching operation have been conducted in EMTP-RV software and the voltage waveform at the CVT secondary is monitored. The simulation model is shown in Fig. 3.2. A 132kV circuit breaker is connected between the source and Bus 1. The circuit breaker consists of a switch (SW) in parallel with grading capacitor (C_b). Series connected interrupting chambers of circuit breakers are used to provide better breaking ability. To improve the balance of voltage distribution across the chambers, grading capacitor is employed. A CVT is connected to Bus 1. The capacitor (C_g) represents the ground capacitance (the capacitance present between the busbar-to-ground with air as an insulation medium).

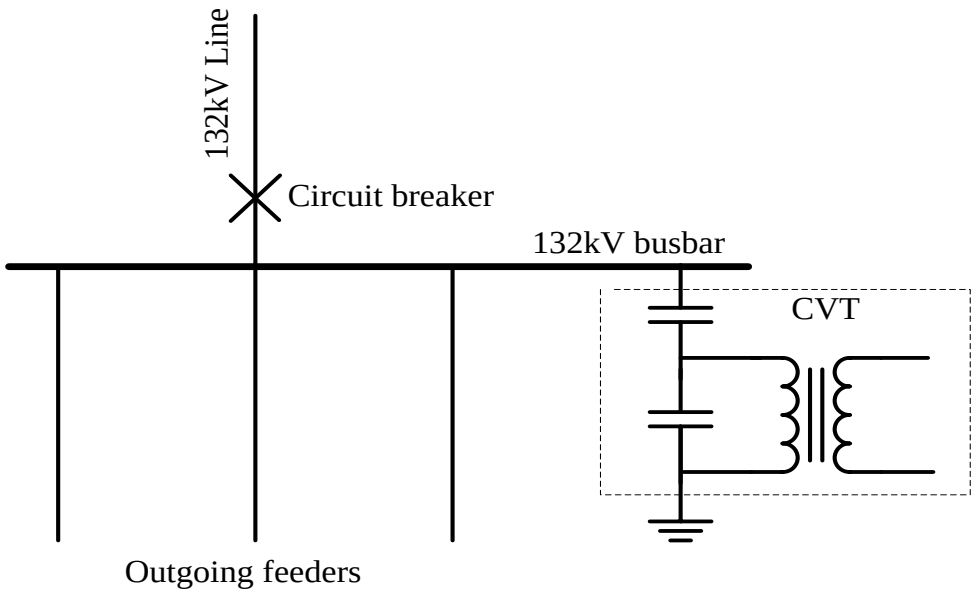


Figure 3.1: Substation configuration

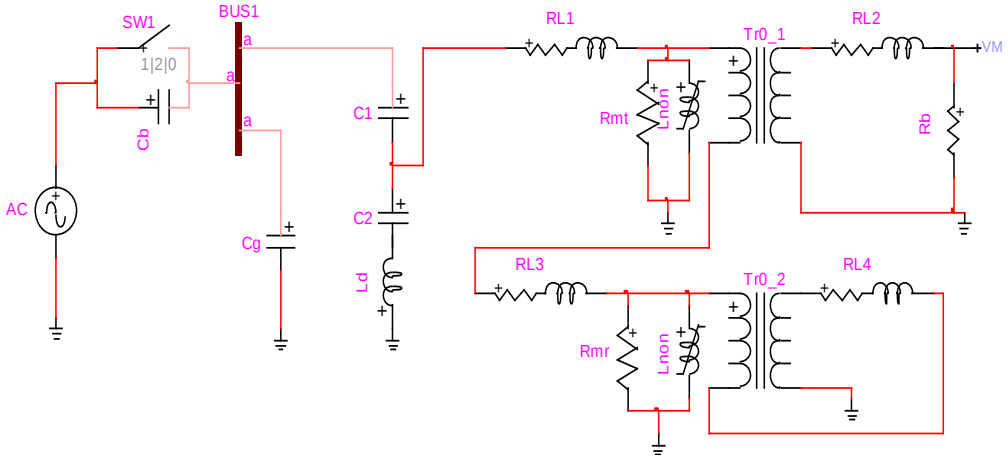
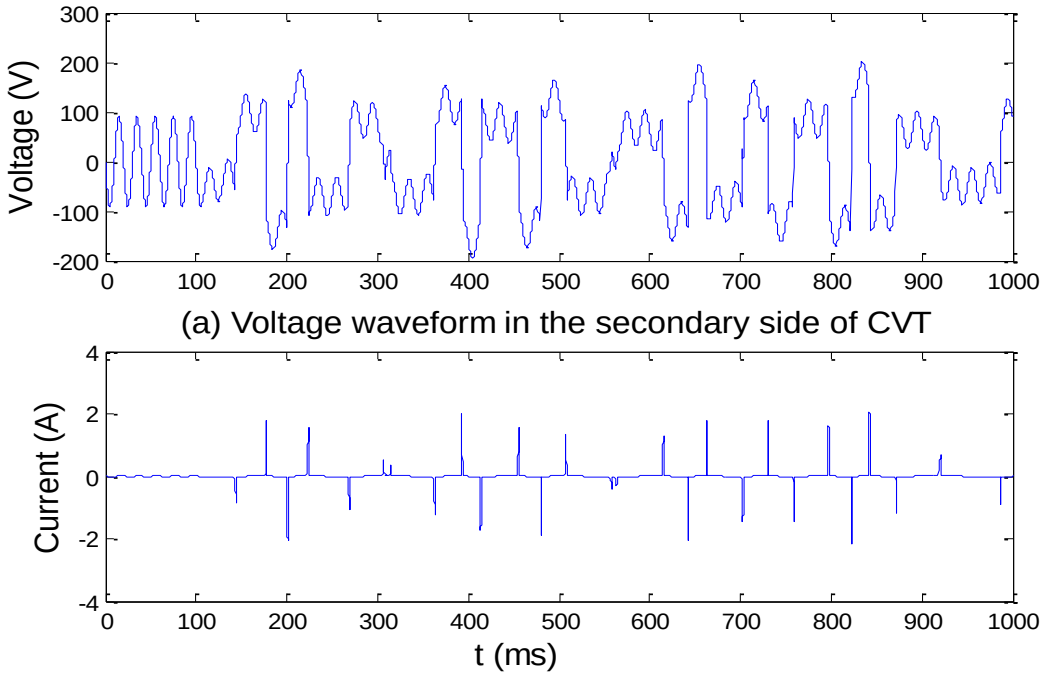


Figure 3.2: Switching simulation of CVT in EMTP software

Opening of circuit breaker initiates the ferroresonant phenomenon, which occur during the de-energization of 132kV bus. This phenomenon involves a nonlinear inductive core of the CVT, circuit breaker grading capacitance and ground capacitance. The value of the grading capacitance, C_b of 8nF and the ground capacitance, C_g of 6nF are used for the circuit to induce a steady state ferroresonance response, following the opening of the circuit breaker. The voltage and current waveforms in the transformer produced from the simulation is depicted in

Fig. 3.3. Fig. 3.3(a) shows the voltage waveform across the CVT secondary terminal and Fig. 3.3(b) shows the current waveform in the primary of the step down transformer of CVT. The frequency spectrum of the steady-state part voltage of Fig. 3.3(a) is shown in Fig. 3.4, which has been identified as the sustained subharmonic ferroresonant mode because the frequency contents are described having a spectrum of frequencies equal to $f_0/3$ (where f_0 is the source frequency). It resonates at 50 Hz frequency with a sustainable amplitude of 1.8 per unit. The magnitude of this kind in CVT is of serious concern because of possible damage to the CVT. In addition, the frequency content consists of the fundamental frequency component as well as the existence of lower order frequency components.



(b) Current waveform in the primary side of CVT

Figure 3.3: CVT secondary voltage and primary current waveform

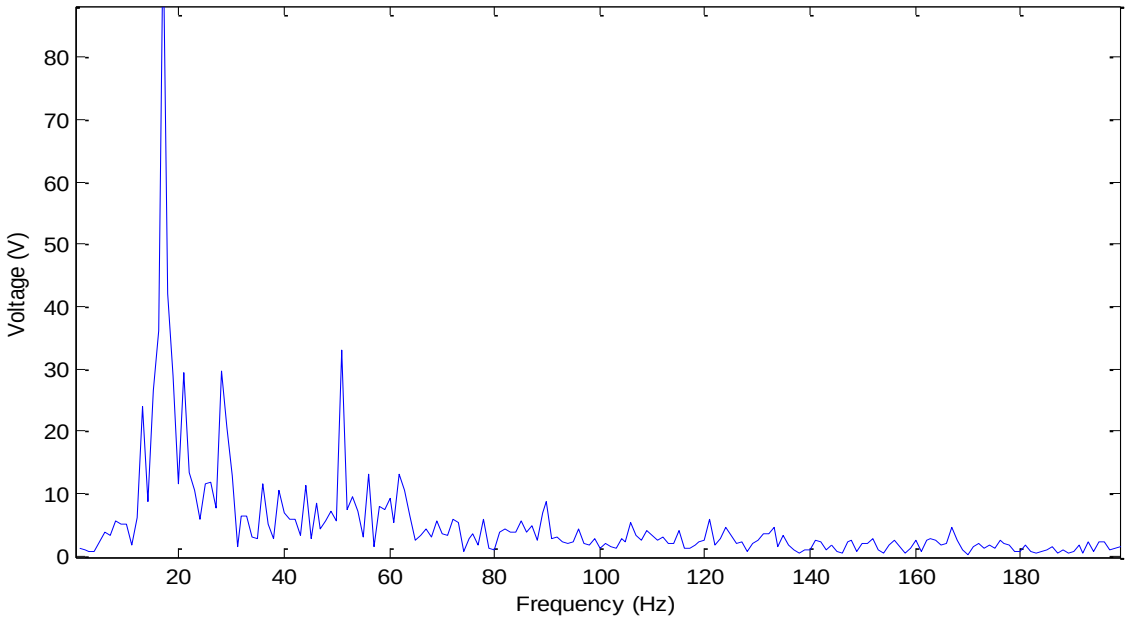


Figure 3.4: FFT for voltage waveform in the secondary side of CVT

3.2 Proposed Electronic Ferroresonance Suppression Circuit

Various ferroresonance suppression techniques in CVT have been proposed as mitigation solutions. The most common ferroresonance suppression techniques are the active FSC and passive FSC. Recently a new electronic FSC is proposed. In general, the ferroresonance suppression circuit can be categorized into passive, active and electronic FSC. Electronic type FSC does not contain any energy storage device. As a result, transient response of CVT will be improved. However ferroresonance detection technique was not explained and did not perform any real time test to prove the effectiveness to mitigate ferroresonance in CVT. In addition, a total of two external gating circuits are required in this FSC design. Therefore a more accurate study is needed on proposed ferroresonance mitigation circuits in CVT. However, it was found that a simpler form of electronic FSC can be achieved. As such, a new electronic FSC is proposed in this research. The configuration consists of a damping resistance

R_a, a power transistor Q and four diodes as shown in Fig. 3.5. This new configuration requires only one driver circuit which is much easier to implement as compared to two external gating circuits required for conventional electronic FSC shown in Figure 2.16.

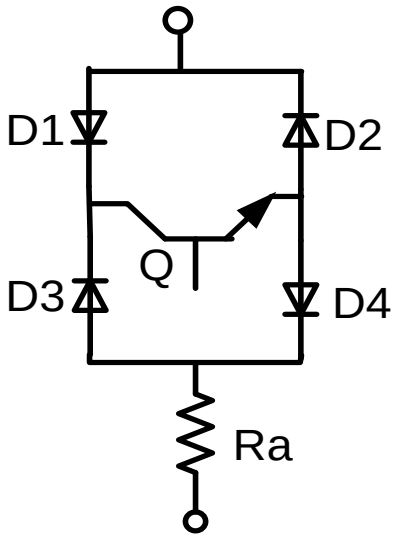


Figure 3.5: Proposed electronic type FSC.

3.2.1 Ferroresonance detection in CVT

CVTs have two output terminals. The first is the power system frequency (50Hz) output terminal, used as an input signal source to protective relays and meters in substations. The second one is a high frequency output terminal normally used for power line carrier communication (PLC) applications. In this research, ferroresonance detection method is implemented based on the waveform analysis of these two output terminal voltages.

After ferroresonance initiation in the CVT, the behavior of the circuit can be explained by the following free oscillation circuit as shown in Fig. 3.6. The losses of the circuit are considered negligible. The magnetization characteristic curve φ(i) of the step down transformer iron core

coil is represented in Fig. 3.7. When ferroresonance is initiated in the circuit, the waveforms are typical of a periodic ferroresonance as shown in Fig. 3.8 and corresponding circuit equations can be expressed by Eqn. 3.1 to Eqn. 3.10. Fig. 3.8 (a) shows the periodic ferroresonance voltage waveform across the capacitor terminal and Fig 3.8 (b) shows the periodic ferroresonance current waveform through the circuit after initiation of ferroresonance.

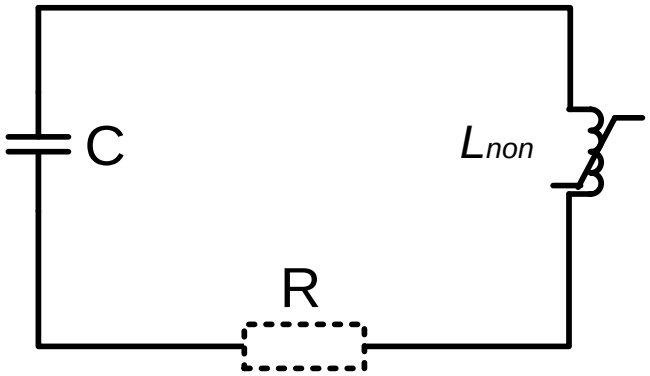


Figure 3.6: Schematic diagram of free oscillation circuit

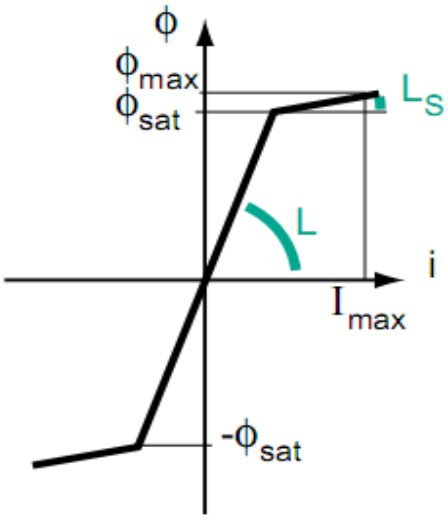


Figure 3.7: Simplified characteristic φ(i) (Ferracci, 1998)

Referring to Fig. 3.8 (b), at the instant $t = t_0$, a resonant current is created and oscillates at the pulsation of ω_1. Where

$$\omega_1 = \frac{1}{\sqrt{LC}} \tag{3.1}$$

The voltage across the capacitor terminal can be expressed as follows.

$$v = V_o \cos \omega_1 t \tag{3.2}$$

Therefore, the flux in the SDT coil can be expressed as:

$$\phi = \left(\frac{V_o}{\omega_1} \right) \sin \omega_1 t \tag{3.3}$$

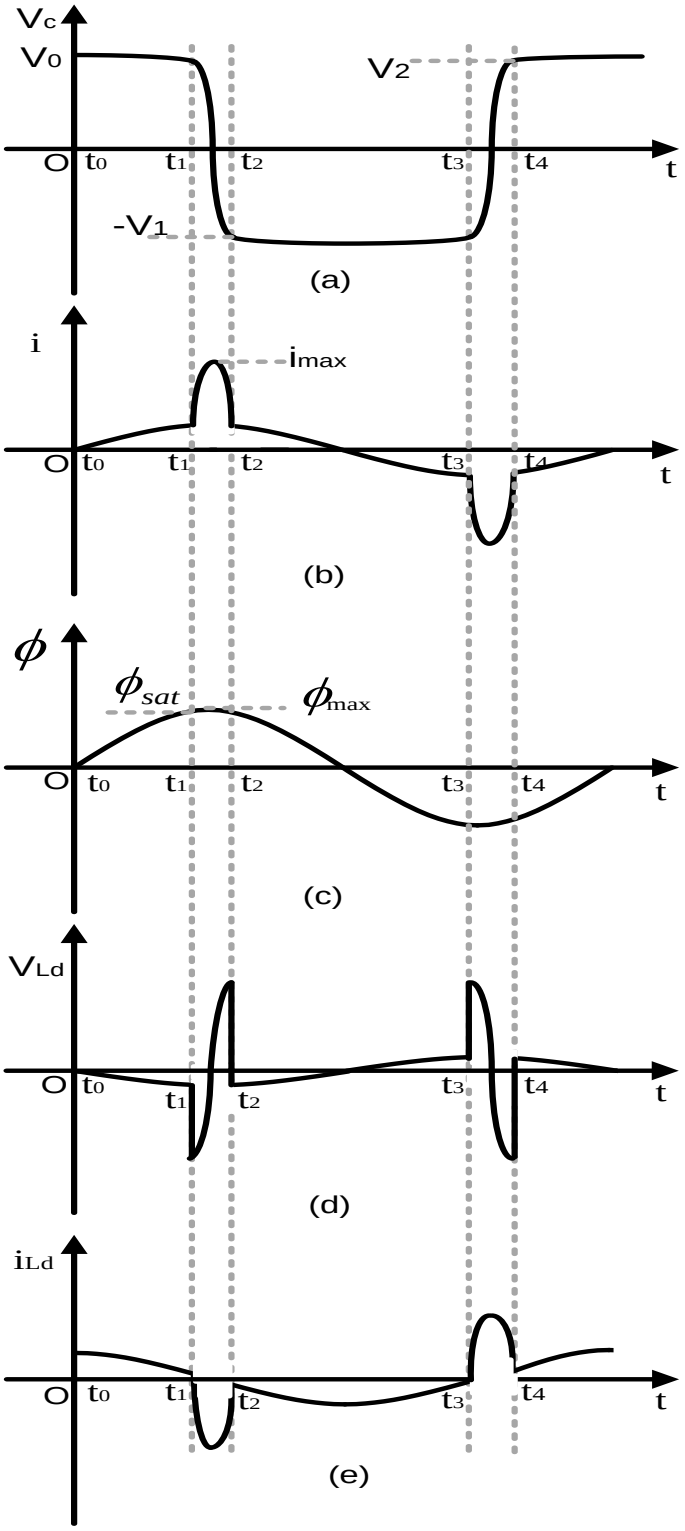


Figure 3.8: Free oscillations of a series ferroresonant circuit

Referring to Fig. 3.8 (c), at the instant $t = t_{1,}$ if $\frac{V_o}{\omega_1} > \phi_{sat}$, the flux ϕ in the coil reaches the saturation flux ϕ_{sat} and saturated inductance of the iron core coil becomes $L_{S.}$ At that instant, voltage across the capacitor is $V_{1.}$

Since saturated inductance L_S is very low compare to the non-saturated inductance L, therefore the capacitor abruptly discharges through the saturated coil in the form of an oscillation of pulsation $\omega_{2.}$ Where

$$\omega_2 = \frac{1}{\sqrt{L_S C}} \tag{3.4}$$

At the instant $t = t_{2,}$ the flux in the coil returns to ϕ_{sat}, and inductance of the coil returns to L. The voltage across the capacitor is then reversed to $-V_{1,}$ because, the losses of the circuit was considered negligible.

At the instant $t = t_3$, the flux in the coil becomes $-\phi_{sat}$ and voltage across the capacitor is $-V_{2.}$ Consequently, the period of the oscillation T is between non saturated case period T_{ns} and saturated case period $T_{s.}$

Where,

$$T_{ns} = 2\pi\sqrt{LC} \tag{3.5}$$

$$T_s = 2\pi\sqrt{L_S C} + 2(t_3 - t_2) \tag{3.6}$$

Therefore, the corresponding frequency f ($f = \frac{1}{T}$) is thus:

$$\frac{1}{2\pi\sqrt{LC}} < f < \frac{1}{2\pi\sqrt{L_S C}} \tag{3.7}$$

The flux change $\Delta\phi$ during non-saturated interval $(t_3 - t_2)$ can be expressed as:

$$\Delta\phi = 2\phi_{sat} = \int_{t_2}^{t_3} v\,dt \tag{3.8}$$

Since, ω_1 is very small in practice, therefore the voltages V_1, V_2 and V_o can be considered equal. Consequently, Eqn. 3.8 can be simplified as follows:

$$t_3 - t_2 \approx \frac{2\phi_{sat}}{V_0} \tag{3.9}$$

Replacing the value of $(t_3 - t_2)$ in Eqn. 3.6, the saturated case interval can be expressed as follows:

$$T_s \approx 2\pi\sqrt{L_S C} + \frac{4\phi_{sat}}{V_o} \tag{3.10}$$

In this research, ferroresonance phenomenon in CVT is detected by analyzing the power system frequency output terminal voltage signal and high frequency output terminal voltage signal during the saturated period $(t_2 - t_1)$ or $(t_4 - t_3)$.

In order to detect inrush current during the saturation period, a low pass filter is connected across the drain coil. The current waveform in the drain coil consists of inrush current during saturated period, as shown in Fig. 3.8(e). Therefore, the voltage across the coil can be represented by the following equation.

$$V_{L_d}(t) = -L_d \frac{di_{L_d}(t)}{dt} \tag{3.11}$$

Corresponding voltage waveform across the drain coil is shown in Fig. 3.8(d).

A short moving window of the CVT secondary voltage and drain coil voltage waveforms analysis is introduced to implement the ferroresonance detection scheme. Referring to Fig. 3.8 (e), during the saturation period, at the beginning of the inrush current in the drain coil, a high inrush voltage V_p appears across the drain coil which is much higher than the maximum nominal voltage and then the coil voltage follows the inrush current rising and falling rate according to Eqn.3.11. At the end of the saturation period the voltage in the drain coil is then reversed to $-V_p$ and then the current in the coil returns to the nominal current and voltage follows accordingly (according to the power system voltage). In other words, the inrush voltages at the beginning and end of the saturation period have to be close to equal and reversed. Referring to Fig. 3.8 (a) and Fig. 3.8 (d), CVT ferroresonance will be detected if negative voltage $(-V_1)$ appears in the CVT secondary during positive inrush voltage in the drain coil and positive voltage (V_1) appears in the CVT secondary during negative inrush voltage in the drain coil. In other words, during the saturation period there will be a zero crossing in the CVT output voltage and voltages at the beginning and end of the saturation period have to be equal and reversed. Since saturated inductance L_S is very low compare to the non-saturated inductance L, therefore saturation period is also very small. Consequently, a high sampling rate is necessary to detect ferroresonance accurately. In the proposed detection method, 0.2 cycles length of the data window will be taken under study. However, a new length of the window will be started if a high inrush positive or negative voltage appears across the drain coil.

3.2.2 Proposed decision making algorithm

The proposed detection method will be activated by the occurrence of high inrush positive or negative drain coil voltage which will be identified by comparing few voltage samples. The interval between the peak inrush positive and negative voltage of the drain coil is taken as the data under study. After ferroresonance initiation, when the peak of inrush voltage appears in the drain coil, the algorithm will search for opposite polarity inrush voltage for the next 0.2 cycles. The voltage in the CVT secondary at the instance when initial inrush voltage appears in the drain coil is then compared with the voltage at the moment when last opposite polarity inrush voltage appears in the drain coil. If the voltages at those two moments are equal or close to equal and reversed, CVT ferroresonance will be detected.

However, in order to achieve protection of the ferroresonance detection algorithm against noise and harmonics, the length of the moving window is considered to be longer than the inrush period of the drain coil voltage. Since the drain coil voltage $V_{Ld}(t)$ will be restored according to the system voltage $V_s(t)$ after the saturation period (t_2-t_1) or (t_4-t_3), therefore detection algorithm compares few voltage samples to obtain immunity against noise and harmonics. The moving window length is increased by 100 samples in addition with the duration of saturation period. Referring to Eqn. 3.10, since saturation period depends on the non-linearity (φ_{sat}) and initial capacitor voltage, therefore the window length is not fixed and the length of the moving window under study will be changed accordingly. The flowchart of the proposed CVT ferroresonance detection algorithm is illustrated in Fig. 3.9.

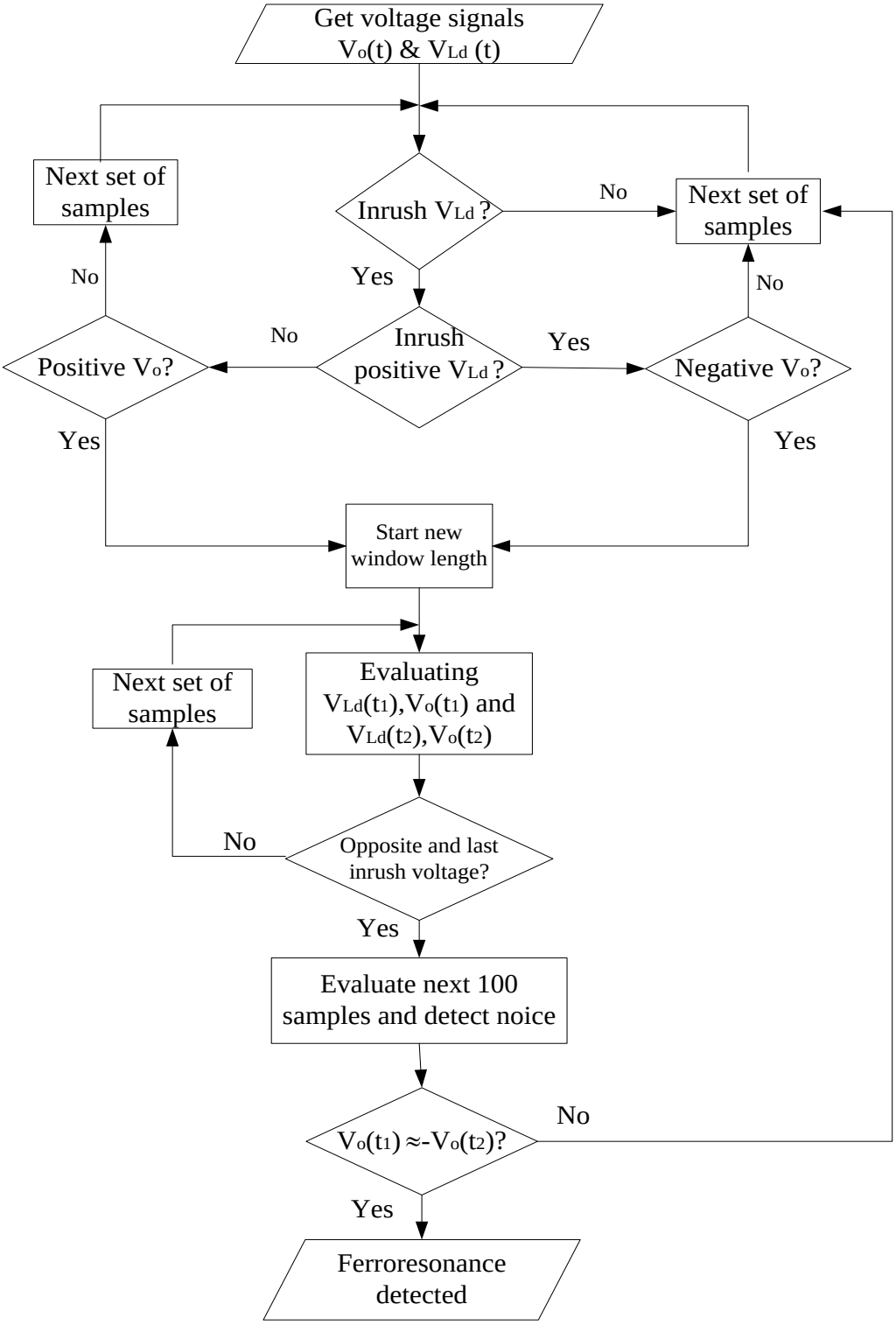


Figure 3.9: Flow chart of the proposed detection algorithm

3.3 Experimental Design

The protection system of capacitive voltage transformer is very important from the ferroresonance viewpoint, because capacitive voltage transformer is used for monitoring and controlling the high voltage system and as input sources to protective relays. However, until now only basic studies have been performed on Active and Passive ferroresonance suppression circuits using EMTP or PSCAD/EMTDC. On the other hand electronic type FSC has been proposed by an author. However ferroresonance detection technique was not explained in detail and did not perform any real time test to prove the effectiveness to mitigate ferroresonance in CVT. Therefore a more accurate study is needed on proposed ferroresonance mitigation circuits in CVT. In others words, it is needed to test the ferroresonance mitigation technique in real time using real time power system simulator, like RTDS.

The purpose of this experiment is to detect ferroresonance phenomenon in CVT and application of damping circuit in the CVT by the proposed electronic FSC to mitigate ferroresonance effectively in CVT. In this study, a test system is developed in RTDS and ferroresonance phenomenon is initiated in CVT using circuit breaker switching operation. In the proposed closed loop test system, ferroresonance in CVT will be detected by a detector circuit which includes a DSP microcontroller and the proposed electronic type FSC will be activated to mitigate ferroresonance in CVT.

3.3.1 Real time digital simulator (RTDS)

The RTDS is a real time digital electromagnetic transient power system analysis simulator manufactured by RTDS Technologies. It is a combination of advanced computer hardware and software (RSCAD). RSCAD is a graphical user interface (GUI) program used to model a test system and act as the user's main interface with the RTDS hardware. The RTDS hardware is made up of racks with custom parallel processing hardware architecture assembled in each unit. Each rack is made up of processing and communication modules. Processor modules are used for computing the solutions of the power system components and network equations fast enough modeled in the software. It can provide output conditions that represent actual conditions in the real network. Therefore, RTDS can be connected to the power system protective relay and control components. The RTDS hardware also contains various cards for sending analog or digital signals out of RTDS and for receiving signals from different equipment. Closed-loop testing is performed using RTDS to detect ferroresonance in CVT by taking two signals output from the RTDS hardware and then using these signals as inputs to the detecting circuit under test. The GTAO card is used to send analogue signals out of the RTDS for interfacing to the detecting circuit under test. The GTAO card has higher precision than the processor card front panel D/A channels and the GTAO card is optically isolated from the RTDS processor cards. A photograph of GTAO card of RTDS is shown in Fig. 3.10 from where analogue signals were taken as input signal sources for ferroresonance detecting circuit.

Figure 3.10: GTAO card of RTDS

Output from the ferroresonance detecting circuit being tested is then fed back into the simulation through the digital input port of the RTDS. In the test system, maximum “10 mA” signal is provided to the RTDS in order to operate the switch in the proposed electronic type FSC as shown in Fig. 3.5 to include the ferroresonance damping resistor in the circuit. Closed-loop testing of the ferroresonance detecting circuit using RTDS provides a realistic environment and closely resembles the actual performance of the proposed technique.

3.3.2 Hardware-in-loop (HIL) Testing

The proposed electronic type ferroresonance suppression technique is tested using real time digital simulator (RTDS). The procedure of the low level closed loop testing is as follows:

1. Modeling the test system with a 132kV CVT using RSCAD software.
2. Detecting ferroresonance phenomenon in the DSP microcontroller by analyzing the voltage signals in the CVT secondary and the voltage across the drain coil V_{Ld} generated from the real time simulation of the RTDS.
3. Feedback of the switch control signal ("on" or "off" signal) to the damping circuit in the RTDS to damp out ferroresonance in CVT.

Fig. 3.11 shows the block diagram of closed loop test hardware setup of the RTDS with the ferroresonance detecting circuit. To assess the proposed FSC method, a test system is modeled in RSCAD, where ferroresonance phenomenon is initiated in a CVT. A case study is carried out to validate the proposed FSC method to damp out ferroresonance in CVT. The test system is circuit breaker operation which initiates ferroresonance in CVT. Ferroresonance detecting algorithm is implemented in a DSP microcontroller (TMS320F28335). A photograph of the laboratory test setup is shown in Fig. 3.12.

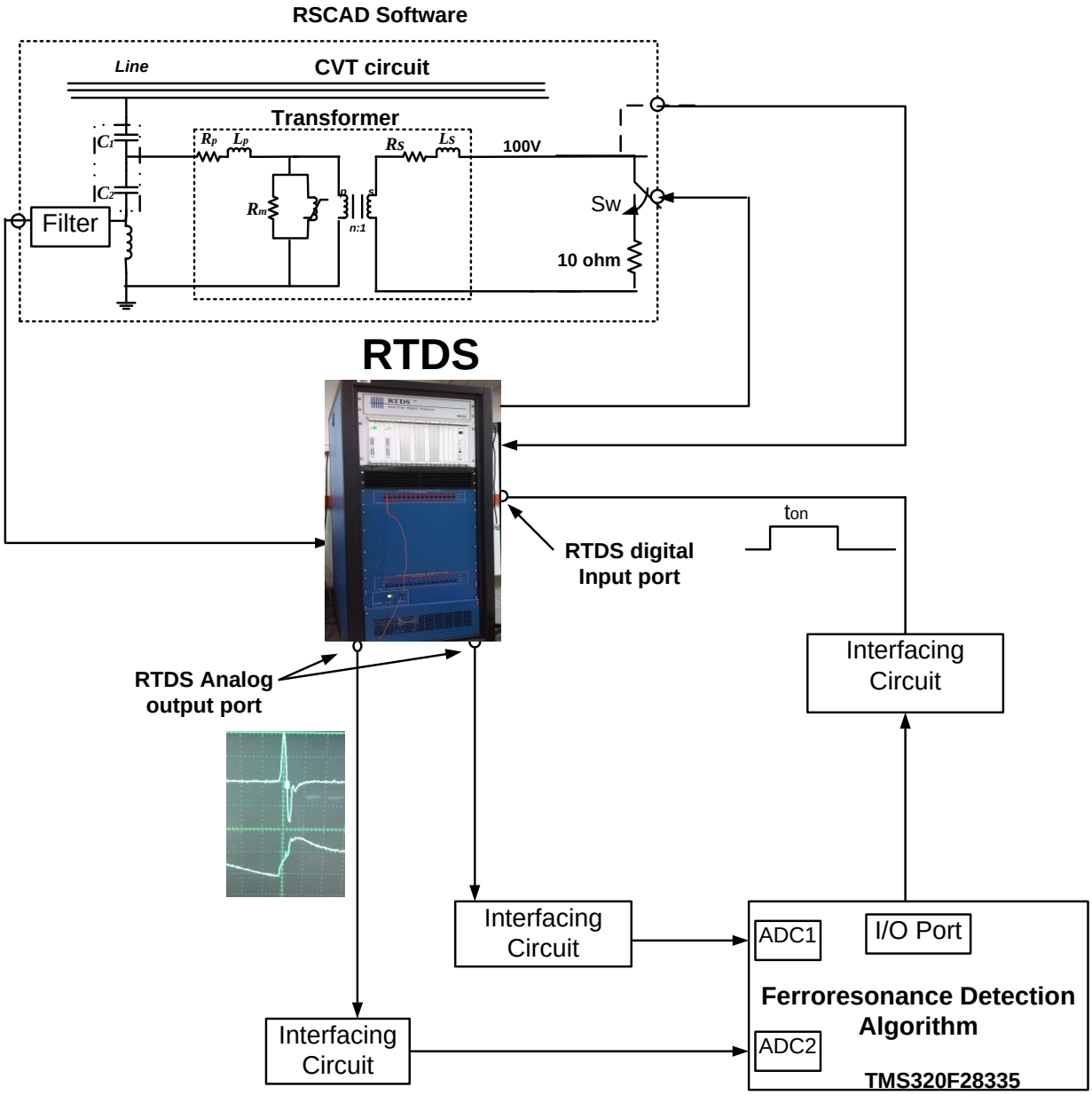


Fig. 3.11: Block diagram of hardware in loop testing using RTDS

Fig. 3.12: Laboratory test setup

CHAPTER 4: RESULTS AND DISCUSSION

4.1 Ferroresonance Suppression Performance

In this section, the ferroresonance suppression performance for the various FSC presented in the above section will be analyzed. In this simulation, ferroresonance oscillation is initiated by circuit breaker switching in a typical substation configuration adopted from (Bakar et al., 2011). The simulation model as shown in Fig.4.1 has been adopted for this study. Supply comes from 132kV transmission line and is connected to a circuit breaker. A bus CVT is connected to the busbar for voltage measurement.

The circuit breaker consists of a switch (SW1) in parallel with grading capacitor (Cb). The capacitor (Cg) represents the ground capacitance. In this simulation, the circuit breaker is opened at 200ms to initiate the ferroresonance phenomenon.

The ferroresonance damping performance for passive, active and proposed electronic FSCs are shown in Fig. 4.2. To investigate the ferroresonance mitigation performance of passive ferroresonance suppression circuit, a passive FSC is connected across the CVT secondary terminal. Referring to Fig. 4.2 (a), ferroresonance is initiated in CVT at 200ms by circuit breaker switching operation using the circuit configuration as shown in Fig. 4.1. It can be observed that the duration of over voltage appears across the CVT secondary due to ferroresonance is 100ms. In other words, the passive FSC requires about 5 cycles to damp out the ferroresonance oscillations in CVT. Furthermore, to investigate the ferroresonance mitigation performance of active ferroresonance suppression circuit, an active FSC is connected across the CVT secondary terminal and ferroresonance is initiated in CVT using the same procedure as described above. Referring to Fig. 4.2 (b), it can be observed that the duration of over voltage appears across the CVT secondary due to ferroresonance is 50ms. It

can be concluded that, the performance of active FSC is better compare to passive FSC where it managed to effectively damp out the ferroresonance oscillations within less than 2.5 cycles. Finally, the proposed electronic ferroresonance suppression circuit is connected in the CVT to investigate the ferroresonance detection and mitigation performance. The proposed electronic FSC achieved the best results where it successfully damped out the ferroresonance within 2 cycles, as shown in Fig. 4.2 (c). With proper ferroresonance detection circuit, higher burden can be connected across the secondary of CVT to suppress ferroresonance in shorter time.

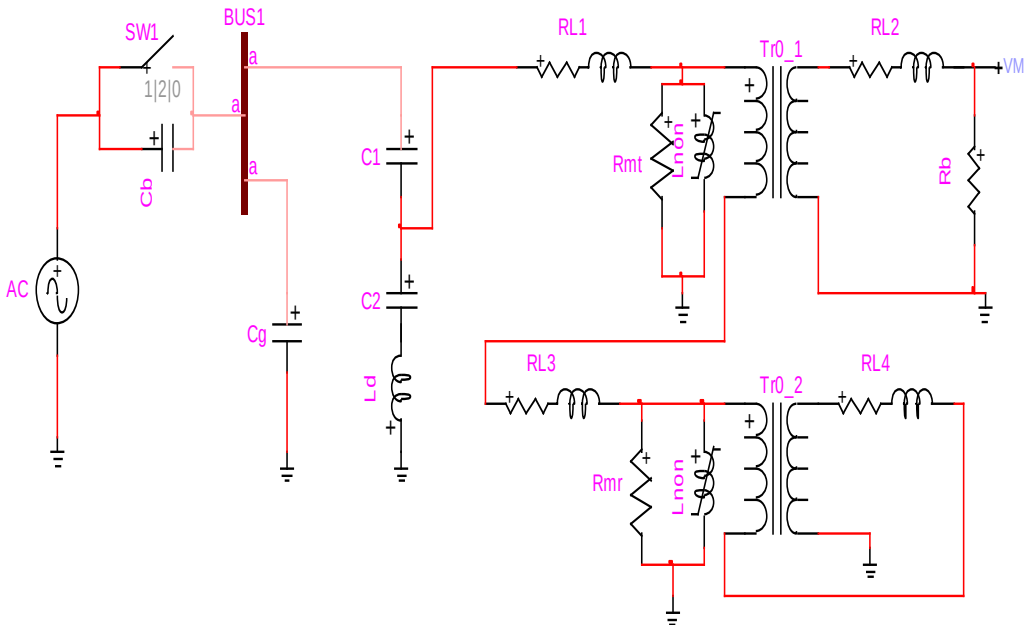
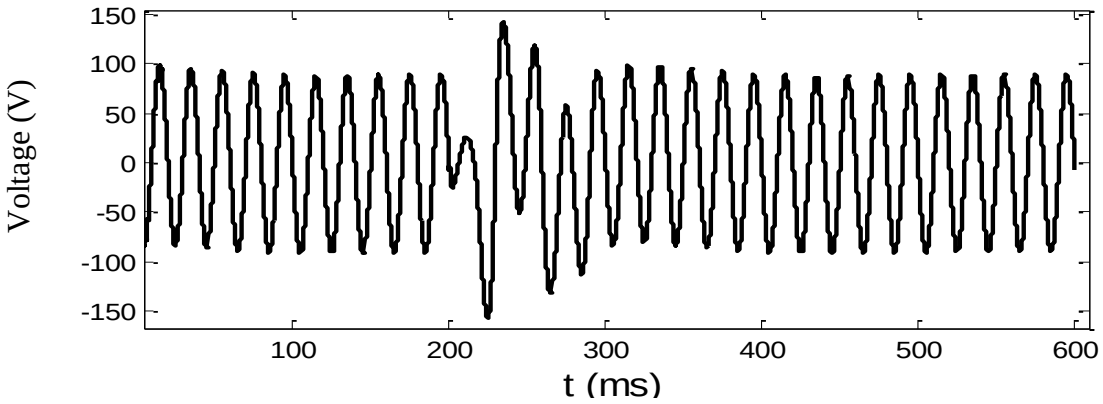
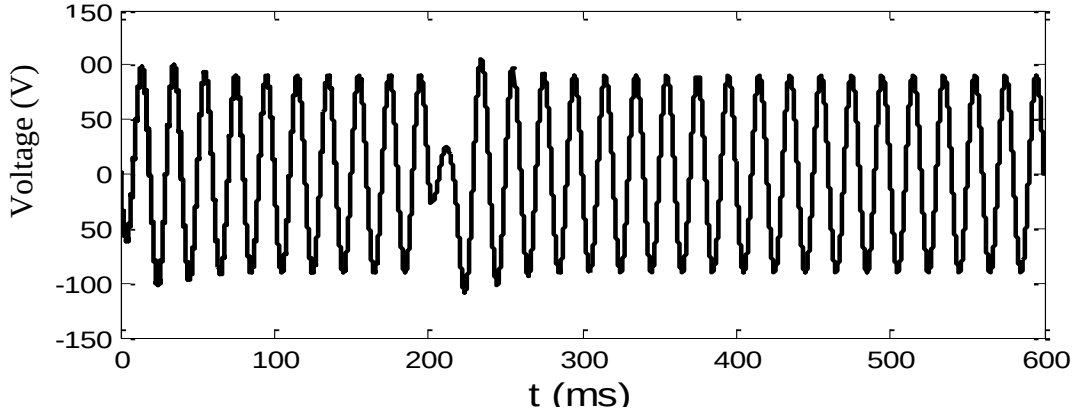


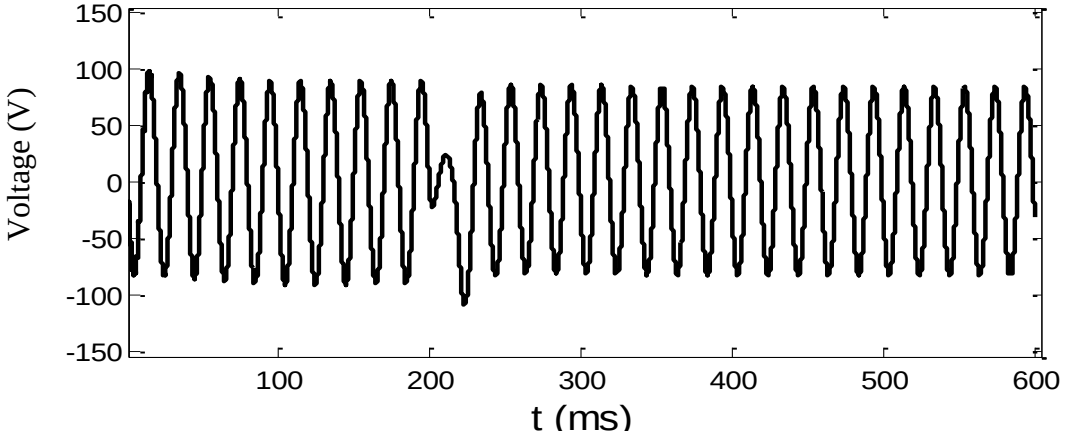
Figure 4.1: Switching simulation of CVT in EMTP software



(a) CVT output voltage with passive FSC in service



(b) CVT output voltage with Active FSC in service



(c) CVT output voltage with Electronic FSC in service

Figure 4.2: CVT secondary voltage with FSC in service

4.2 CVT Transient Response Performance

To analyse the transient response, a line to ground fault is simulated using EMTP-RV software by short circuiting the 132 kV line. The equivalent 132 kV CVT circuit used for this study is shown in Fig. 4.3. Stray capacitances are included in the CVT configuration for this transient response analysis.

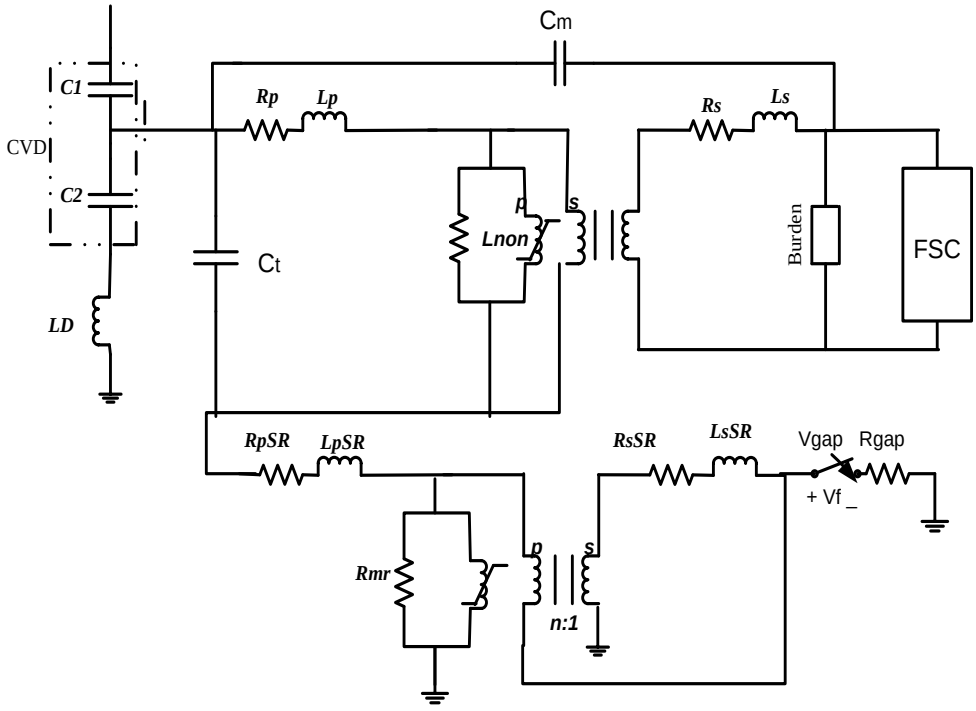


Figure 4.3: Schematic diagram of the CVT model used for transient simulation.

The transient response is analyzed for the 132 kV CVT with the three FSC configurations comprising of active, passive and electronic FSC. It must be noted that the burden for passive and active FSC are permanently connected regardless of the presence of ferroresonance. In contrast, the electronic FSC is designed to activate the loading resistor only when it is required to suppress ferroresonance. This electronic FSC will disconnect the resistor in the absence of

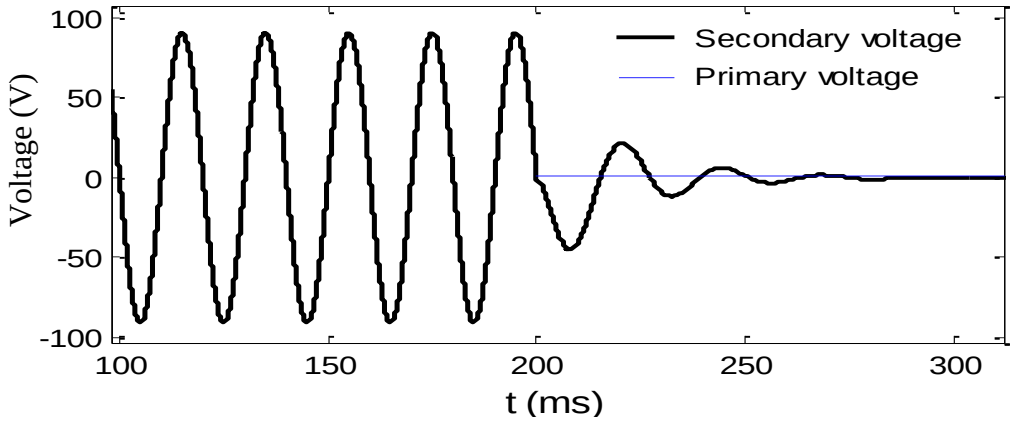
ferroresonance. Figs. 4.4, 4.5 and 4.6 show the transient responses of CVT during short circuit faults for active, passive and electronic FSC respectively. The transient responses are analyzed for faults that occurred at both voltage zero and peak voltage.

Referring to the results for fault with active FSC in service, it can be observed from Fig. 4.4 that at the instant the fault is applied, the primary 132 kV voltage reduces to zero immediately. However, the secondary CVT voltage could not track the primary 132 kV voltage accurately for both cases when fault is applied during voltage zero and peak voltage. The deviations can be observed to last for around 4 cycles. As for the results with passive FSC in service, it can be observed from Fig. 4.5 that the secondary CVT voltage could not track the primary 132 kV voltage accurately when fault occurs at voltage zero. However, the secondary CVT voltage is able to track the primary 132 kV voltage with reasonable accuracy when fault occurs at peak voltage.

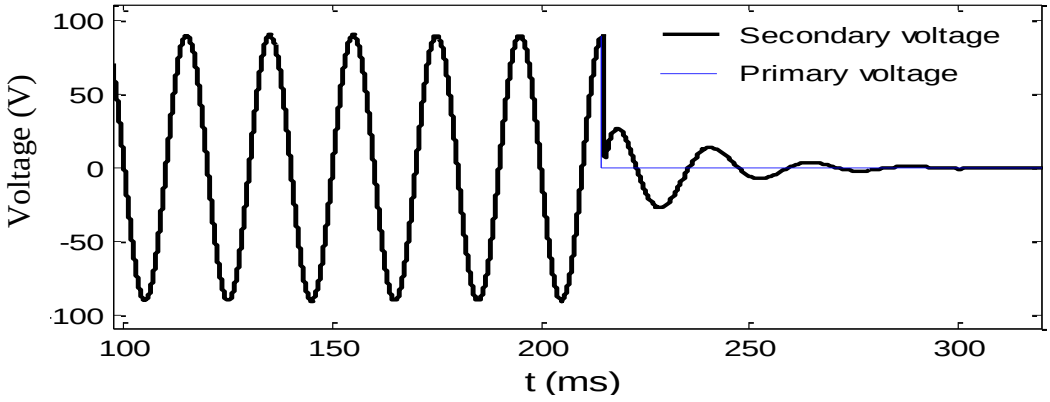
The CVT with electronic FSC achieved the best transient response where the secondary CVT voltage is able to track the primary 132 kV voltage accurately when fault occurs at both voltage zero and peak voltage, as shown in Fig. 4.6. The results indicated that active and passive ferroresonance suppression circuits can significantly impact the transient performance of CVT. The introduction of FSC may mitigate the impacts of ferroresonance. But, these FSCs have also presented new challenge to the CVT. The conventional passive or active FSC consists of non-linear components such as capacitors and inductors. These capacitors and inductors will store energy and distort the response of the CVT in accurately tracking the primary voltage waveforms. It has been reported that the inaccuracies in the transient response of CVT caused underreaching, overreaching or direction errors in impedance relays, which will jeopardize the protection scheme. The factors that influence the CVT transient response

include coupling capacitors, excitation current of the intermediate transformer, connected burden and ferroresonance suppression circuit.

In contrast, the proposed electronic FSC is designed to activate the loading resistor only when it is required to suppress ferroresonance. During this fault simulation, ferroresonance is not initiated. As such, the loading resistor is disconnected by the electronic FSC. Consequently, there is no non-linear components that will store energy and distorts the secondary voltage waveforms for the CVT with electronic FSC. As such, this electronic FSC achieved the best of both world where ferroresonance and transient response are of utmost concerned.

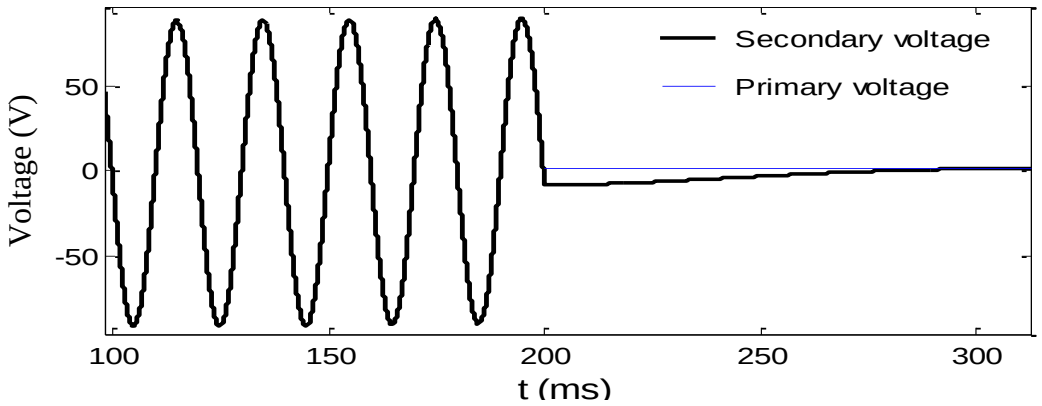


(a) CVT Transient for Fault at Voltage Zero

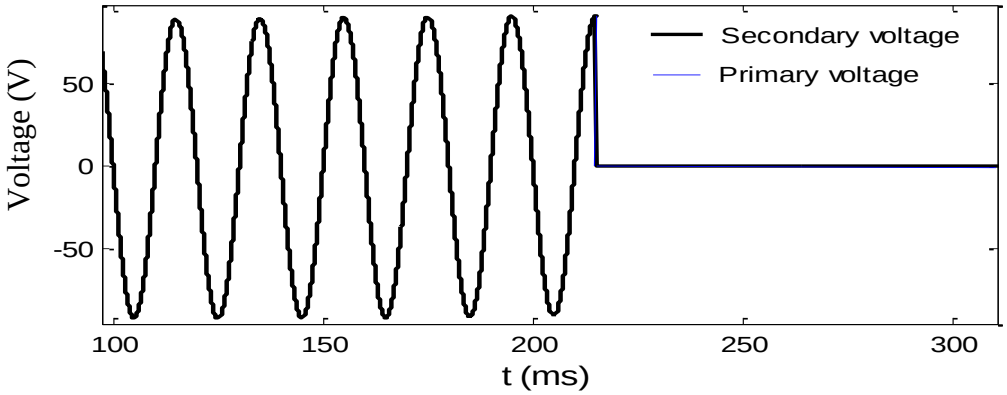


(b) CVT Transient for Fault at Voltage Peak

Figure 4.4: CVT transient response with active FSC in service

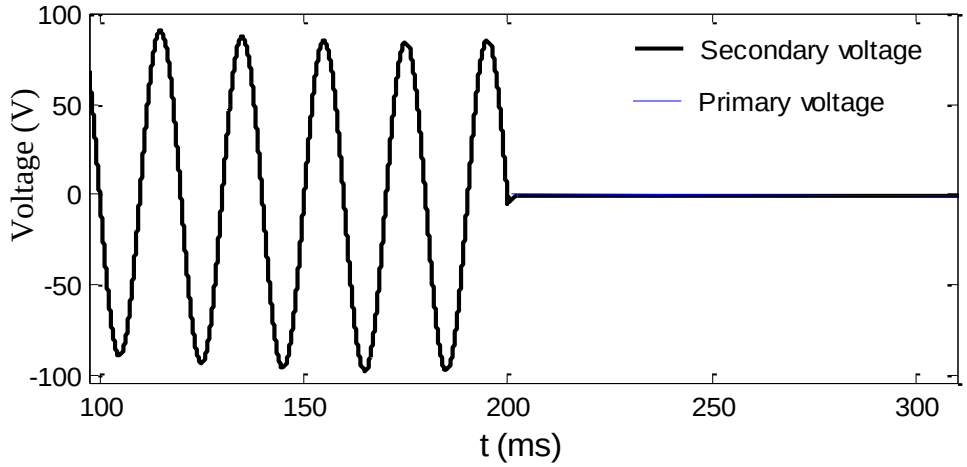


(a) CVT Transient for Fault at Voltage Zero

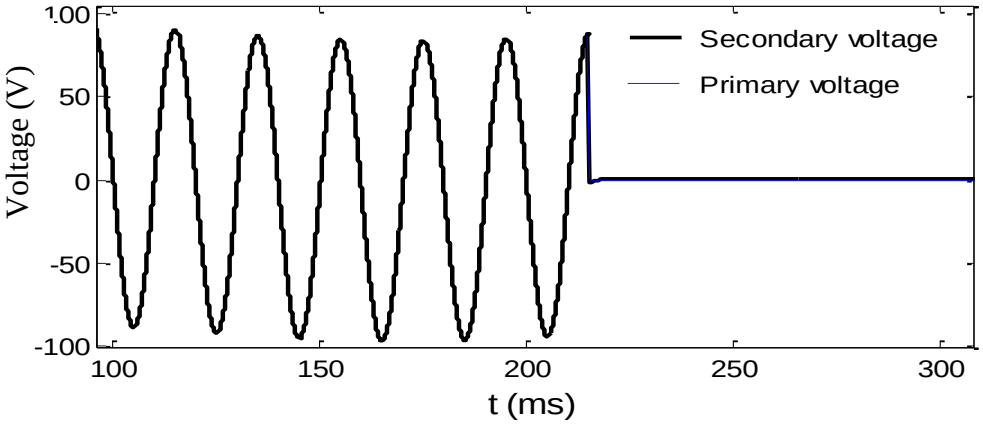


(b) CVT Transient for Fault at Voltage Peak

Figure 4.5: CVT transient response with passive FSC in service



(a) CVT Transient for Fault at Voltage Zero



(b) CVT Transient for Fault at Voltage Peak

Figure 4.6: CVT transient response with electronic FSC in service

It must also be noted that although both CVT with active and passive FSC could not accurately track the primary voltage, a comparison of transient error for both CVTs indicated that the CVT with passive FSC delivered a better results. It can also be observed that the fault at voltage zero generated a worse transient response as compared to fault at peak voltage.

4.3 Hardware Results

The proposed CVT ferroresonance mitigation technique is implemented by using a hardware setup with RTDS to test whether the proposed technique can properly detect the ferroresonance phenomenon in CVT. A test system is modeled in RTDS using the same circuit configuration as described in section 3.1 to initiate ferroresonance in CVT. In the test system, output terminals of the CVT circuit are configured so that exact replicas of the signals are available in the output control card of the RTDS hardware. The CVT outputs which are taken from the output control card (GTAO) of the RTDS hardware are fed to two ADC pins of a DSP microcontroller through first order low pass RC filter and interfacing circuit. The internal ADC of the DSP microcontroller discretizes the analog signals from RTDS. The discrete signals from these two voltage signals are then analyzed based on the proposed ferroresonance detection algorithm using a TMS320F28335 DSP microcontroller. Ferroresonance phenomenon was initiated in the CVT using circuit breaker switching operation to investigate the performance of the ferroresonance detection system. Fig. 4.7 shows the voltage signal from RTDS hardware which was taken from the secondary terminal of CVT and Fig. 4.8 shows the voltage signal which was taken from the drain coil of the CVT where ferroresonance suppression circuit was not connected. Fig. 4.9 shows the magnified view of voltage signals of Fig. 4.7 and Fig. 4.8 across the CVT secondary terminal (bottom) and across the drain coil (top) during ferroresonance. Fig. 4.10 shows the output of the CVT after implementation of the proposed electronic ferroresonance suppression circuit. The output demonstrates that the ferroresonance phenomenon in CVT was accurately detected by the proposed detection algorithm and the proposed electronic type FSC can mitigate ferroresonance within one cycle of the system voltage which is much better compare to the

existing active and passive FSC. A comparison of simulation and hardware results for different ferroresonance mitigation circuits are provided in Table 4.1.

Table 4.1: Comparison results for different FSC circuits

	AFSC	PFSC	EFSC
Simulation results	2.5 cycle	5 cycle	2 cycle
Hardware results	-	-	1 cycle

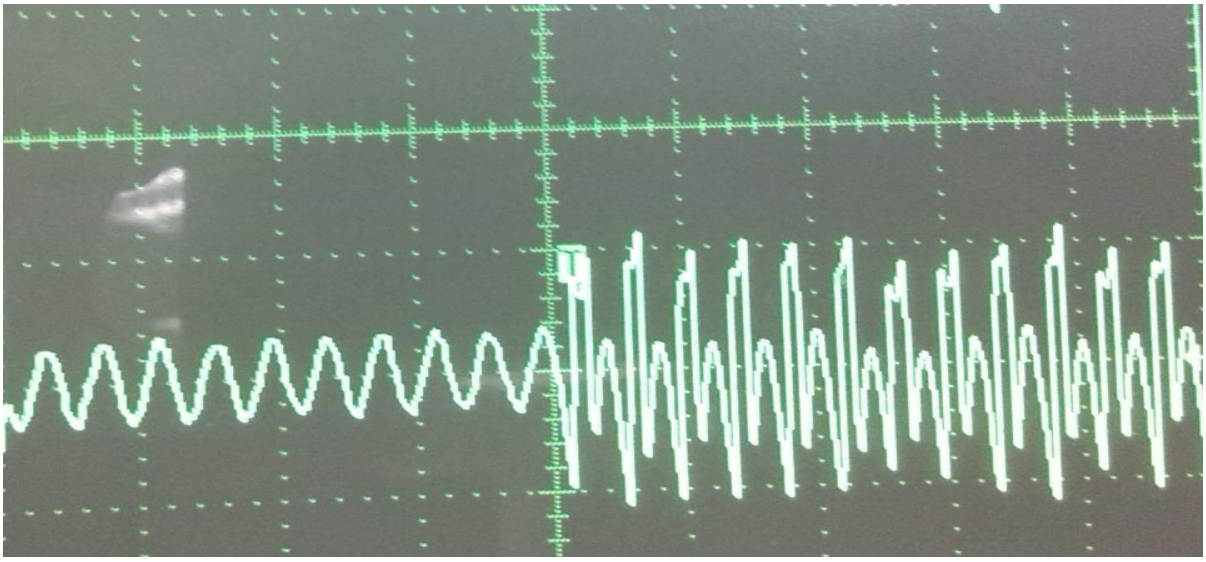


Figure 4.7: Signal from RTDS which was taken from the secondary terminal of CVT

Figure 4.8: Signal from RTDS which was taken from the drain coil of CVT

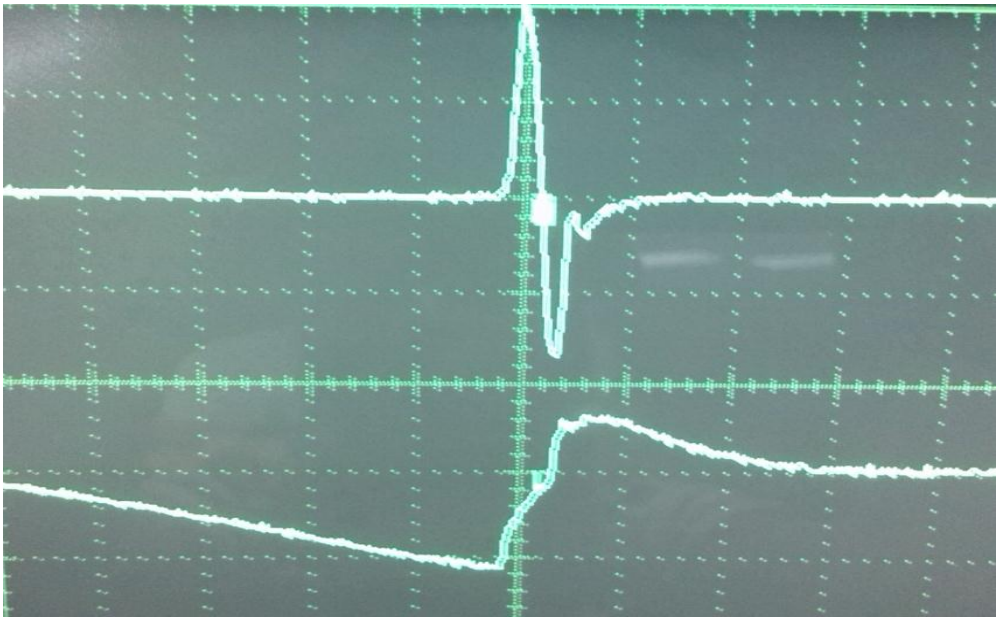


Figure 4.9: Magnified view of voltage signal across the CVT secondary terminal (bottom) and voltage signal across the drain coil (top)

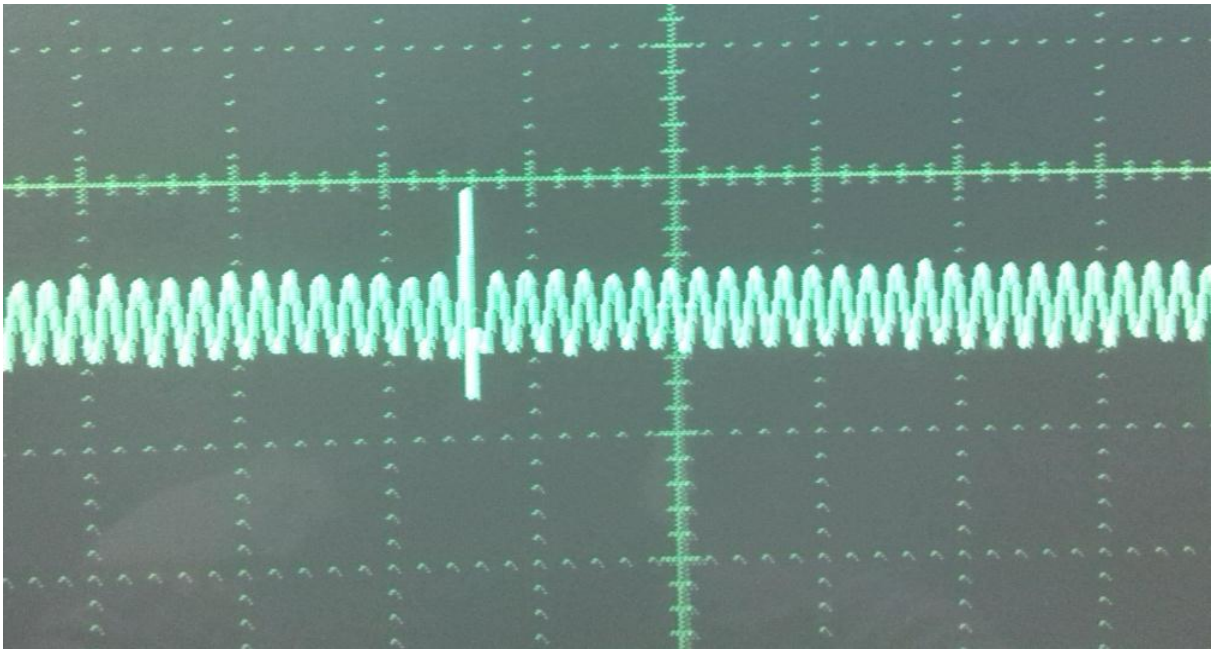


Figure 4.10: Output of the CVT after the implementation of ferroresonance suppression method.

CHAPTER 5: CONCLUSION AND FUTURE WORK

5.1 Conclusion

The awareness and knowledge on ferroresonance incidences in electrical power system is relatively low and they are commonly regarded as unexplained phenomenon which does not pose any real danger to the electrical system by the utility engineers. As a result, this research provided a detail description on the ferroresonance occurrences especially in CVT. An introduction to ferroresonance phenomenon, explanation on ferroresonance initiation phenomenon through its fundamental capacitor and inductor relation are presented in this research. The four modes of ferroresonance are also explained in detail. Ferroresonance occurrence in CVT is an area of special interest in this research due to the limited research conducted. The initiating mechanism for ferroresonance in CVT such as switching is presented. Contrary to the common misconception regarding the risk of ferroresonance, this research highlighted the importance of ferroresonance awareness among utility engineers.

In this research, different ferroresonance mitigation circuits have been reviewed. In addition, a simpler electronic FSC which consists of only one control switch has been proposed. Besides that, the core focus of this research lies in proposing a new technique to detect ferroresonance event in CVT and implementation of the detection algorithm in a DSP microcontroller. The accuracy of the proposed ferroresonance detection and mitigation technique is verified through the laboratory test results. Closed-loop testing is performed using real time digital simulator (RTDS). In addition, this research analyzed ferroresonance suppression performance and transient performance of the CVT with active, passive and electronic FSC respectively. It was demonstrated that the electronic FSC delivered the best results for both the ferroresonance suppression and transient performance. As such, the electronic FSC is recommended for implementation in the CVTs to suppress ferroresonance.

5.2 Future Work

Future works that can be recommended from this thesis are:

1. Distance relay performance analysis while CVT with different type of FSC is used for voltage monitoring.

2. Simulation of different switching event in power system that can cause ferroresonance in CVT.

3. Implementation of electronic ferroresonance suppression circuit in the CVT.

REFERENCES

A. H. A. Bakar, N. A. R., and M. K. M. Zambri. (2011). Analysis of lightning-caused ferroresonance in Capacitor Voltage Transformer (CVT). *International Journal of Electrical Power & Energy Systems, 33*, 1538.

Abbasi Fordoei, H. R., Gholami, A., Fathi, S. H., & Abbasi, A. (2013). Chaotic oscillations control in the voltage transformer including nonlinear core loss model by a nonlinear robust adaptive controller. *International Journal of Electrical Power & Energy Systems, 47*(0), 280-294. doi: http://dx.doi.org/10.1016/j.ijepes.2012.11.013

Ajaei, F. B., & Sanaye-Pasand, M. (2008, 12-15 Oct. 2008). *Minimizing the Impact of Transients of Capacitive Voltage Transformers on Distance Relay*. Paper presented at the Power System Technology and IEEE Power India Conference, 2008. POWERCON 2008. Joint International Conference on.

Akinci, T. C., Ekren, N., Seker, S., & Yildirim, S. (2013). Continuous wavelet transform for ferroresonance phenomena in electric power systems. *International Journal of Electrical Power & Energy Systems, 44*(1), 403-409. doi: http://dx.doi.org/10.1016/j.ijepes.2012.07.001

Ang, S. P. (2010). Ferroresonance Simulation Studies of Transformer Systems. *PhD Thesis Submitted to The University of Manchester*.

B. Tanggawelu, R. N. M., and A. E. Ariffin. (2003). *Ferroresonance Studies in Malaysian Utility's Distribution Network*. Paper presented at the IEEE Power Engineering Society General Meeting.

Badrkhani Ajaei, F., Sanaye-Pasand, M., Rezaei-Zare, A., & Iravani, R. (2009). Analysis and Suppression of the Coupling Capacitor Voltage Transformer Ferroresonance Phenomenon. *Power Delivery, IEEE Transactions on, 24*(4), 1968-1977. doi: 10.1109/TPWRD.2009.2028818

Bakar, A. H. A., Lim, C. H., & Mekhilef, S. (2006, 28-29 Nov. 2006). *Investigation of Transient Performance of Capacitor Voltage Transformer*. Paper presented at the Power and Energy Conference, 2006. PECon '06. IEEE International.

Bakar, A. H. A., Rahim, N. A., & Zambri, M. K. M. (2011). Analysis of lightning-caused ferroresonance in Capacitor Voltage Transformer (CVT). *International Journal of Electrical Power & Energy Systems, 33*(9), 1536-1541. doi: http://dx.doi.org/10.1016/j.ijepes.2011.06.004

Bakar, A. H. A., Talib, D. N. A., Mokhlis, H., & Illias, H. A. (2013). Lightning back flashover double circuit tripping pattern of 132kV lines in Malaysia. *International Journal of Electrical Power & Energy Systems, 45*(1), 235-241. doi: http://dx.doi.org/10.1016/j.ijepes.2012.08.048

Barbisio, E., Bottauscio, O., Chiampi, M., Crotti, G., & Giordano, D. (2008). Parameters Affecting Ferroresonance in "LCR" Electric Circuits. *Magnetics, IEEE Transactions on, 44*(6), 870-873. doi: 10.1109/tmag.2007.916314

Ben Amar, F., & Dhifaoui, R. (2011). Study of the periodic ferroresonance in the electrical power networks by bifurcation diagrams. *International Journal of Electrical Power & Energy Systems, 33*(1), 61-85. doi: http://dx.doi.org/10.1016/j.ijepes.2010.08.003

Bethenod, J. (Nov. 30, 1907). Sur le Transformateur à Résonance. *L'Éclairage Électrique, 53*, 289-296.

Bollen, M. H. J. (2000). Understanding Power Quality Problems: Voltage Sags and Interruptions. *Wiley Interscience, A John Wiley & Sons Inc Publication*.

Corporation, C. (May 29, 2002). Ferroresonance. *TECHNICAL BULLETIN-004a*.

Costello, D., & Zimmerman, K. (2012, 2-5 April 2012). *CVT transients revisited — Distance, directional overcurrent, and communications-assisted tripping concerns.* Paper presented at the Protective Relay Engineers, 2012 65th Annual Conference for.

Daqing, H., & Roberts, J. (1996, 26-29 May 1996). *Capacitive voltage transformer: transient overreach concerns and solutions for distance relaying.* Paper presented at the Electrical and Computer Engineering, 1996. Canadian Conference on.

Davarpanah, M., Sanaye-Pasand, M., & Badrkhani Ajaei, F. (2012). Compensation of CVT Increased Error and Its Impacts on Distance Relays. *Power Delivery, IEEE Transactions on, 27*(3), 1670-1677. doi: 10.1109/tpwrd.2012.2197422

Dugan, R. C. (2003). *Examples of Ferroresonance in Distribution System*. Paper presented at the IEEE Power Engineering Society General Meeting.

Fernandes Jr, D., Neves, W. L. A., & Vasconcelos, J. C. A. (2007). Coupling capacitor voltage transformer: A model for electromagnetic transient studies. *Electric Power Systems Research, 77*(2), 125-134. doi: http://dx.doi.org/10.1016/j.epsr.2006.02.007

Giordano, D., Banu, N., Barbisio, E., Bottauscio, O., Chiampi, M., & Crotti, G. (2009). Ferroresonance Analysis in LCR Circuits Including Rapidly Solidified Alloy Cores. *Magnetics, IEEE Transactions on, 45*(9), 3243-3250. doi: 10.1109/tmag.2009.2020978

Graovac, M., Iravani, R., Wang, X., & McTaggart, R. D. (2003). Fast ferroresonance suppression of coupling capacitor voltage transformers. *Power Delivery, IEEE Transactions on, 18*(1), 158-163. doi: 10.1109/TPWRD.2002.803837

Hassan, S., Vaziri, M., & Vadhva, S. (2011, 3-5 Aug. 2011). *Review of ferroresonance in power distribution grids.* Paper presented at the Information Reuse and Integration (IRI), 2011 IEEE International Conference on.

Hedding, R. A. (2012). *CCVT transient fundamentals*. Paper presented at the 2012 65th Annual Conference for Protective Relay Engineers.

Huang, S.-J., & Hsieh, C.-H. (2013). Relation analysis for ferroresonance of bus potential transformer and circuit breaker grading capacitance. *International Journal of Electrical Power & Energy Systems, 51*(0), 61-70. doi: http://dx.doi.org/10.1016/j.ijepes.2013.03.005

IEC International Standard on Instrument transformers Part 5: Capacitor Voltage Transformers. (2004). *IEC Standard*.

Jazebi, S., Farazmand, A., Murali, B. P., & de Leon, F. (2013). A Comparative Study on "pi"and "T" Equivalent Models for the Analysis of Transformer Ferroresonance. *Power Delivery, IEEE Transactions on, 28*(1), 526-528. doi: 10.1109/tpwrd.2012.2220637

K.Pattanapakdee, C. B. (2007). Failure of Riser Pole Arrester due to Station Service Transformer Ferroresonance. *Internatioinal Conference on Power Systems Transients (IPST'07) Lyon, France*.

Kieny, C. (1991). Application of the bifurcation theory in studying and understanding the global behavior of a ferroresonant electric power circuit. *Power Delivery, IEEE Transactions on, 6*(2), 866-872. doi: 10.1109/61.131146

Lacerda Ribas, J. C., Lourenco, E. M., Leite, J. V., & Batistela, N. J. (2013). Modeling Ferroresonance Phenomena With a Flux-Current Jiles-Atherton Hysteresis Approach. *Magnetics, IEEE Transactions on, 49*(5), 1797-1800. doi: 10.1109/tmag.2013.2243908

Lamba, H., Grinfeld, M., McKee, S., & Simpson, R. (1997). Subharmonic ferroresonance in an LCR circuit with hysteresis. *Magnetics, IEEE Transactions on, 33*(4), 2495-2500. doi: 10.1109/20.595906

Li, S., Han, H., & Zhang, W. (2012, 10-14 Sept. 2012). *Discrimination method of power frequency ferroresonance in GIS.* Paper presented at the Electricity Distribution (CICED), 2012 China International Conference on.

Liu, F., Li, M., Liu, P., Cao, Y., & Zeng, H. (2011). Theoretical Research on Ferroresonance in Neutral Grounded Power System. *Energy Procedia, 12*(0), 403-410. doi: http://dx.doi.org/10.1016/j.egypro.2011.10.054

Lucas, J. R., McLaren, P. G., Keerthipala, W. W. L., & Jayasinghe, R. P. (1992). Improved simulation models for current and voltage transformers in relay studies. *Power Delivery, IEEE Transactions on, 7*(1), 152-159. doi: 10.1109/61.108902

M. V. Escudero, I. D., and M. Redfem. (2004). *Understanding Ferroresonance*. Paper presented at the 39th International Universities Power Engineering Conference.

Mahdi Davarpanah, M. S.-P., and Firouz Badrkhani Ajaei. (2012). CCVT Failure due to Improper Design of Auxiliary Voltage Transformers *IEEE Transactions on Power Delivery, 27*(1), 391-400.

McDermit, D., Shipp, D., Dionise, T., & Lorch, V. (2013). Medium voltage switching transient induced potential transformer failures; Prediction, measurement and practical solutions. *Industry Applications, IEEE Transactions on, PP*(99), 1-1. doi: 10.1109/tia.2013.2258453

McDermit, D., Shipp, D. D., Dionise, T. J., & Lorch, V. (2012, 20-24 May 2012). *Medium voltage switching transient induced potential transformer failures; prediction, measurement and practical solutions.* Paper presented at the Industrial & Commercial Power Systems Technical Conference (I&CPS), 2012 IEEE/IAS 48th.

Milicevic, K., & Emin, Z. (2013). Initiation of Characteristic Ferroresonance States Based on Flux Reflection Model. *Circuits and Systems II: Express Briefs, IEEE Transactions on, 60*(1), 51-55. doi: 10.1109/tcsii.2012.2234897

Mork, B. A., & Stuehm, D. L. (1994). Application of nonlinear dynamics and chaos to ferroresonance in distribution systems. *Power Delivery, IEEE Transactions on, 9*(2), 1009-1017. doi: 10.1109/61.296285

Moses, P. S., Masoum, M. A. S., & Toliyat, H. A. (2011). Impacts of Hysteresis and Magnetic Couplings on the Stability Domain of Ferroresonance in Asymmetric Three-Phase Three-Leg Transformers. *Energy Conversion, IEEE Transactions on, 26*(2), 581-592. doi: 10.1109/tec.2010.2088400

Naidu, S. R., & De Souza, B. A. (1997). Newton-Raphson approach for the analysis of ferroresonant circuits. *Generation, Transmission and Distribution, IEE Proceedings-, 144*(5), 489-494. doi: 10.1049/ip-gtd:19971432

P.Ferracci. (1998). Ferroresonance. *Cahier technique no. 190*.

Piasecki, W., Florkowski, M., Fulczyk, M., Mahonen, P., & Nowak, W. (2007). Mitigating Ferroresonance in Voltage Transformers in Ungrounded MV Networks. *Power Delivery, IEEE Transactions on, 22*(4), 2362-2369. doi: 10.1109/tpwrd.2007.905383

R. C. Dugan, M. F. M., S. Santoso, H. W. Beaty. (2003). Electrical Power Systems Quality. *McGraw-Hill, Second Edition*.

Radmanesh, H., & Gharehpetian, G. B. (2013). Ferroresonance suppression in power transformers using chaos theory. *International Journal of Electrical Power & Energy Systems, 45*(1), 1-9. doi: http://dx.doi.org/10.1016/j.ijepes.2012.08.028

Radmanesh, H., Hosseinian, S. H., & Fathi, S. H. (2012, 28-31 May 2012). *Harmonic study in electromagnetic voltage transformers.* Paper presented at the Industrial Electronics (ISIE), 2012 IEEE International Symposium on.

Rezaei-Zare, A., Iravani, R., & Sanaye-Pasand, M. (2009). Impacts of Transformer Core Hysteresis Formation on Stability Domain of Ferroresonance Modes. *Power Delivery, IEEE Transactions on, 24*(1), 177-186. doi: 10.1109/tpwrd.2008.2002668

S. Shahabi, D. B., R. E. Shirvani, and M. Purrezagholi. (2009). *Mitigating ferroresonance in coupling capacitor voltage transformers with ferroresonance suppressing circuits*. Paper presented at the IEEE EUROCON.

S. Shahabi, M. M., A. Gholami, and S. Taheri. (2009). *Investigation of performance of ferroresonance suppressing circuits in coupling capacitor voltage transformers*. Paper presented at the 4th IEEE Conference on Industrial Electronics and Applications.

Sakamuri, J., & Yesuraj, D. J. (2011, 19-23 June 2011). *Modeling and simulation of capacitor voltage transformer transients using PSCAD/EMTDC.* Paper presented at the PowerTech, 2011 IEEE Trondheim.

Sanaye-Pasand, M., Rezaei-Zare, A., Mohseni, H., Farhangi, S., & Iravani, R. (2006, 0-0 0). *Comparison of performance of various ferroresonance suppressing methods in inductive and capacitive voltage transformers.* Paper presented at the Power India Conference, 2006 IEEE.

Saravanaselvan, R., & Ramanujam, R. (2012). Isolated ferroresonant solutions in transmission lines in the same right-of-way. *International Journal of Electrical Power & Energy Systems, 41*(1), 11-15. doi: http://dx.doi.org/10.1016/j.ijepes.2012.02.003

Shein, D., Zissu, S., & Schapiro, W. (1989, 7-9 Mar 1989). *Voltage Transformer Ferroresonance in One 400 Kv Gis Substation.* Paper presented at the Electrical and Electronics Engineers in Israel, 1989. The Sixteenth Conference of.

Shipp, D. D., Dionise, T. J., Lorch, V., & MacFarlane, B. G. (2011). Transformer Failure Due to Circuit-Breaker-Induced Switching Transients. *Industry Applications, IEEE Transactions on, 47*(2), 707-718. doi: 10.1109/tia.2010.2101996

Shipp, D. D., Dionise, T. J., Lorch, V., & MacFarlane, W. G. (2012). Vacuum Circuit Breaker Transients During Switching of an LMF Transformer. *Industry Applications, IEEE Transactions on, 48*(1), 37-44. doi: 10.1109/tia.2011.2175430

Simha, V., & Wei-jen, L. (2008). The jump phenomena. *Industry Applications Magazine, IEEE, 14*(5), 53-59. doi: 10.1109/mias.2008.927533

Siregar, R., & Setiawan, A. (2012, 23-27 Sept. 2012). *Investigation experience due to high number of CVTs failure (Part:1).* Paper presented at the Condition Monitoring and Diagnosis (CMD), 2012 International Conference on.

Ta-Peng, T., & Chia-Ching, N. (2006). Analysis of ferroresonant overvoltages at Maanshan Nuclear Power Station in Taiwan. *Power Delivery, IEEE Transactions on, 21*(2), 1006-1012. doi: 10.1109/tpwrd.2005.860268

Tseng, K. H., & Cheng, P. Y. (2011). Mitigating 161 kV electromagnetic potential transformers' ferroresonance with damping reactors in a gas-insulated switchgear. *Generation, Transmission & Distribution, IET, 5*(4), 479-488. doi: 10.1049/iet-gtd.2010.0453

V. Simha, a. W. J. L. (2008). The Jump Phenomena – Investigation of a Sudden Overvoltage Incident due to Ferroresonance. *IEEE Industrial Applications Magazine, 14*(5), 53-59.

V. Valverde, A. J. M., I. Zamora, and G. Buigues. (2011). *Ferroresonance in Voltage Transformer: Analysis and Simulations*. Paper presented at the International conference on renewable energies and power quality.

Val Escudero, M., Dudurych, I., & Redfern, M. A. (2007). Characterization of ferroresonant modes in HV substation with CB grading capacitors. *Electric Power Systems Research, 77*(11), 1506-1513. doi: http://dx.doi.org/10.1016/j.epsr.2006.08.033

Valverde, V., Buigues, G., Fernandez, E., Mazon, A. J., & Zamora, I. (2012, 25-28 March 2012). *Behavioral patterns in voltage transformer for ferroresonance detection.* Paper presented at the Electrotechnical Conference (MELECON), 2012 16th IEEE Mediterranean.

Zare, M. H., Mirzaei, A., & Abyaneh, H. A. (2012, 15-17 May 2012). *Improving capacitive voltage transformer response and its impact on distance relay performance.* Paper presented at the Electrical Engineering (ICEE), 2012 20th Iranian Conference on.

LIST OF PUBLICATIONS

Journal Articles

1) Ab Halim Abu Bakar, **Shakil Ahamed Khan**, Tan Chia Kwang, Nasrudin Abd. Rahim, “A Review of Ferroresonance in Capacitive Voltage Transformer,” *IEEJ Transactions on Electrical and Electronic Engineering*, Vol. 10, p. 28–35, 2015.

Conference Publications

1) **Shakil Ahamed Khan,** Ab Halim Abu Bakar, Nasrudin Abd. Rahim, Tan Chia Kwang," Analysis of Ferroresonance Suppression and Transient Response Performances for Various Ferroresonance Suppression Circuits in Capacitive Voltage Transformers," in *The 3rd International Conference on Clean Energy and Technology (CEAT)*, 2014.

APPENDICES

Appendix A

TMS320F28335 program

MAIN.C

```
//###########################################################################
//
//###########################################################################

#include "DSP2833x_Device.h"

// external function prototypes
extern void InitAdc(void);
extern void InitSysCtrl(void);
extern void InitPieCtrl(void);
extern void InitPieVectTable(void);
extern void InitCpuTimers(void);
extern void ConfigCpuTimer(struct CPUTIMER_VARS *, float, float);


// Prototype statements for functions found within this file.
void Gpio_select(void);
interrupt void cpu_timer0_isr(void);
interrupt void adc_isr(void);           // ADC  End of Sequence ISR

// Global Variables
int Voltage_VR1,Voltage_VR2, present=0, previous1=0,previous2=0,max=0,min, v_sd =
0, v_Ld = 0,f_data,v_data;
int counter=0;
int count=0, loop=0,zc=0,zcmax=0,f_count=0, disp_f_count;
int FR_P=0,FR_N=0,t_count,counting,nfcount,NS,PS,adc_count;

//###########################################################################
//                               main code
//###########################################################################

void main(void)
{
       InitSysCtrl();      // Basic Core Init from DSP2833x_SysCtrl.c

       EALLOW;
       SysCtrlRegs.WDCR= 0x00AF; // Re-enable the watchdog
       EDIS;               // 0x00AF  to NOT disable the Watchdog, Prescaler = 64

       DINT;               // Disable all interrupts

       Gpio_select();            // GPIO9, GPIO11, GPIO34 and GPIO49 as output

       InitPieCtrl();            // basic setup of PIE table; from
DSP2833x_PieCtrl.c
```

```
       InitPieVectTable(); // default ISR's in PIE

       InitAdc();          // Basic ADC setup, incl. calibration

       AdcRegs.ADCTRL1.all = 0;
       AdcRegs.ADCTRL1.bit.ACQ_PS = 7;  // 7 = 8 x ADCCLK
       AdcRegs.ADCTRL1.bit.SEQ_CASC =1;       // 1=cascaded sequencer
       AdcRegs.ADCTRL1.bit.CPS = 0;           // divide by 1
       AdcRegs.ADCTRL1.bit.CONT_RUN = 0;      // single run mode

       AdcRegs.ADCTRL2.all = 0;
       AdcRegs.ADCTRL2.bit.INT_ENA_SEQ1 = 1;  // 1=enable SEQ1 interrupt
       AdcRegs.ADCTRL2.bit.EPWM_SOCA_SEQ1 =1;// 1=SEQ1 start from ePWM_SOCA trigger
       AdcRegs.ADCTRL2.bit.INT_MOD_SEQ1 = 0;  // 0= interrupt after every end of
sequence

       AdcRegs.ADCTRL3.bit.ADCCLKPS = 3;      // ADC clock: FCLK = HSPCLK / 2 *
ADCCLKPS
                                       // HSPCLK = 75MHz (see DSP2833x_SysCtrl.c)
                                       // FCLK = 12.5 MHz

       AdcRegs.ADCMAXCONV.all = 1;    // 1 conversion from Sequencer 1    //     0

       AdcRegs.ADCCHSELSEQ1.bit.CONV00 = 0; // Setup ADCINA0 as 1st SEQ1 conv.
       AdcRegs.ADCCHSELSEQ1.bit.CONV01 = 1; // Setup ADCINA0 as 1st SEQ1 conv.
// del

       EPwm2Regs.TBCTL.all = 0xC030;    // Configure timer control register
       /*
        bit 15-14     11:     FREE/SOFT, 11 = ignore emulation suspend
        bit 13        0:      PHSDIR, 0 = count down after sync event
        bit 12-10     000:    CLKDIV, 000 => TBCLK = HSPCLK/1
        bit 9-7       000:    HSPCLKDIV, 000 => HSPCLK = SYSCLKOUT/1
        bit 6         0:      SWFSYNC, 0 = no software sync produced
        bit 5-4       11:     SYNCOSEL, 11 = sync-out disabled
        bit 3         0:      PRDLD, 0 = reload PRD on counter=0
        bit 2         0:      PHSEN, 0 = phase control disabled
        bit 1-0       00:     CTRMODE, 00 = count up mode
       */

       EPwm2Regs.TBPRD = 2999;// TPPRD +1  =  TPWM / (HSPCLKDIV * CLKDIV * TSYSCLK)
                              //                =  20 µs / 6.667 ns

       EPwm2Regs.ETPS.all = 0x0100;     // Configure ADC start by ePWM2
       /*
        bit 15-14     00:     EPWMxSOCB, read-only
        bit 13-12     00:     SOCBPRD, don't care
        bit 11-10     00:     EPWMxSOCA, read-only
        bit 9-8       01:     SOCAPRD, 01 = generate SOCA on first event
        bit 7-4       0000:   reserved
        bit 3-2       00:     INTCNT, don't care
        bit 1-0       00:     INTPRD, don't care
       */

       EPwm2Regs.ETSEL.all = 0x0A00;    // Enable SOCA to ADC
       /*
        bit 15        0:      SOCBEN, 0 = disable SOCB
```

```
         bit 14-12     000:    SOCBSEL, don't care
         bit 11        1:      SOCAEN, 1 = enable SOCA
         bit 10-8      010:    SOCASEL, 010 = SOCA on PRD event
         bit 7-4       0000:   reserved
         bit 3         0:      INTEN, 0 = disable interrupt
         bit 2-0       000:    INTSEL, don't care
        */

        EALLOW;
        PieVectTable.TINT0 = &cpu_timer0_isr;
        PieVectTable.ADCINT = &adc_isr;
        EDIS;

        InitCpuTimers();    // basic setup CPU Timer0, 1 and 2

        ConfigCpuTimer(&CpuTimer0,150,100000);

        PieCtrlRegs.PIEIER1.bit.INTx7 = 1;            // CPU Timer 0
        PieCtrlRegs.PIEIER1.bit.INTx6 = 1;            // ADC

        IER |=1;

        EINT;
        ERTM;

        CpuTimer0Regs.TCR.bit.TSS = 0;   // start timer0

        while(1)
        {
               while(CpuTimer0.InterruptCount == 0)
               {
                      EALLOW;
                      SysCtrlRegs.WDKEY = 0x55; // service WD #1
                      SysCtrlRegs.WDKEY = 0xAA; // service WD #2
                      EDIS;
               }

               CpuTimer0.InterruptCount = 0;
               ConfigCpuTimer(&CpuTimer0,100,20000 + Voltage_VR1 * 239.32);
               CpuTimer0Regs.TCR.bit.TSS = 0;   //     restart timer0

               counter++;
               if(counter&1) GpioDataRegs.GPASET.bit.GPIO9 = 1;
                      else GpioDataRegs.GPACLEAR.bit.GPIO9 = 1;
//             if(counter&2) GpioDataRegs.GPASET.bit.GPIO11 = 1;
//     else GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;
               if(counter&4) GpioDataRegs.GPBSET.bit.GPIO34 = 1;
                      else GpioDataRegs.GPBCLEAR.bit.GPIO34 = 1;
               if(counter&8) GpioDataRegs.GPBSET.bit.GPIO49 = 1;
                      else GpioDataRegs.GPBCLEAR.bit.GPIO49 = 1;
        }
}

void Gpio_select(void)
{
        EALLOW;
        // Setting as GPIO function
```

```
	// GPAMUX1    >> GPIO15 ... GPIO0
	// GPAMUX2    >> GPIO31 ... GPIO16
	// GPBMUX1    >> GPIO47 ... GPIO32
	// GPBMUX2    >> GPIO63 ... GPIO48
	// GPCMUX1    >> GPIO79 ... GPIO64
	// GPCMUX2    >> GPIO87 ... GPIO80
	GpioCtrlRegs.GPAMUX1.bit.GPIO9 = 0;
	GpioCtrlRegs.GPAMUX1.bit.GPIO11 = 0;
	GpioCtrlRegs.GPBMUX1.bit.GPIO34 = 0;
	GpioCtrlRegs.GPBMUX2.bit.GPIO49 = 0;

	// Setting GPIO as output
	GpioCtrlRegs.GPADIR.bit.GPIO9 = 1;      // LED D1 at GPIO9
	GpioCtrlRegs.GPADIR.bit.GPIO11 = 1;     // LED D2 at GPIO11
	GpioCtrlRegs.GPBDIR.bit.GPIO34 = 1;     // LED D3 at GPIO34
	GpioCtrlRegs.GPBDIR.bit.GPIO49 = 1;     // LED D4 at GPIO49
	EDIS;
}

interrupt void cpu_timer0_isr(void)
{
	CpuTimer0.InterruptCount++;
	PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;
}



//void delay_loop()
//{
//    volatile long i;
 //   for (i = 0; i < 1000000; i++) {}
//}

void delay_loop(long end)
{
	long i;
	for (i = 0; i < end; i++)
	{
		asm(" NOP");
		EALLOW;
		SysCtrlRegs.WDKEY = 0x55;
		SysCtrlRegs.WDKEY = 0xAA;
		EDIS;
	}
}







interrupt void  adc_isr(void)
{
	Voltage_VR1 = AdcMirror.ADCRESULT0;     // store result global
	Voltage_VR2 = AdcMirror.ADCRESULT1;     // store result global
```

```
    f_data= Voltage_VR1 - 2264 ;
    v_data= Voltage_VR2 - 2264 ;

       adc_count=adc_count+1;
       if(adc_count>10)       /// if adc_count=1000, then full cycle
(50Hz)sample_count=82,,,,,,,    //// 1V == 1300
                            //// ckt  DC value = 1.66 == 2264 digital need to be
substructed
             {

             adc_count = 0;
////////////////////////////////////////////////////////////////////////////////
////////////////////////////////

        counting = counting + 1;


          if  (f_data>=500)
          {

             t_count = t_count +1;

         //    if  (v_data<=-500)
          //       {
           //      NS=1;
            //     }


             count = count +1 ;
              if (count>=1)
              {
               FR_P=5;
               count=0;
              }
          }

          else if (f_data<= -500)            // -155 normal
          {
             t_count = t_count +1;
        //     if  (v_data>=500)
         //        {
         //        PS=1;
         //        }

             count = count +1 ;
              if (count>=1)
              {
               FR_N=5;
               count=0;
              }

          }

          if (FR_P==5)
          {
             if(FR_N==5)
```

```
        {

//        if(NS==1)
  //       {
  //        if(PS==1)
  //        {
           if(t_count<100)
            {
            nfcount = nfcount + 1;

           FR_P=0;
           FR_N=0;
           NS=0;
           PS=0;

           t_count = 0;
           count = 0;
           counting = 0;

             GpioDataRegs.GPASET.bit.GPIO11 = 1;
             GpioDataRegs.GPASET.bit.GPIO9 = 1;
             delay_loop(1000000);
             delay_loop(1000000);
             delay_loop(1000000);
            GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;
            GpioDataRegs.GPACLEAR.bit.GPIO9 = 1;
  //         GpioDataRegs.GPASET.bit.GPIO11 = 1;
   //        GpioDataRegs.GPASET.bit.GPIO9 = 1;
  //          delay_loop();
 //           delay_loop();



     //       }
   //        }
         }
        }
      }


   else
   {
         GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;
         GpioDataRegs.GPACLEAR.bit.GPIO9 = 1;
   }


  if (counting >= 30)
      {
        if (FR_P == 0)   ///   ||
        {
          if (FR_N == 0)
          {
            t_count = 0;
            count = 0;
```

```
                counting = 0;

                GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;
                GpioDataRegs.GPACLEAR.bit.GPIO9 = 1;
 //               delay_loop();
       //         delay_loop();

              }

            }
          }




////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////

             }



//                  if(Voltage_VR1>2000) GpioDataRegs.GPASET.bit.GPIO11 = 1;
//           else GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;

/*

             if(Voltage_VR1>2000)
             {
                   count=count+1;
                   if(count>30000)

                         {
                         GpioDataRegs.GPASET.bit.GPIO11 = 1;
                         count=0;
                         }

              }
             else
             {

                   GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;
                   count=0;
             }

   */



  //  previous = Voltage_VR1;

       // Reinitialize for next ADC sequence
       AdcRegs.ADCTRL2.bit.RST_SEQ1 = 1;       // Reset SEQ1
       AdcRegs.ADCST.bit.INT_SEQ1_CLR = 1;          // Clear INT SEQ1 bit
```

```
	PieCtrlRegs.PIEACK.all = PIEACK_GROUP1; // Acknowledge interrupt to PIE
}

//===========================================================================
// End of SourceCode.
//===========================================================================
```

DISPLAY_ADC.C

```
#include "DSP2833x_Device.h"

void display_ADC(unsigned int result)
/* show the 12-bit "result" of the AD-conversion at 4 LEDs (GPIO49,34,11 and 9)
	*/
/* the result will be show as light-beam
		*/
{
	switch((result>>8)/3)
	{
		case 4:
		{
			GpioDataRegs.GPBSET.bit.GPIO49 = 1;
			GpioDataRegs.GPBSET.bit.GPIO34 = 1;
			GpioDataRegs.GPASET.bit.GPIO11 = 1;
			GpioDataRegs.GPASET.bit.GPIO9 = 1;
			break;
		}
		case 3:
		{
			GpioDataRegs.GPBCLEAR.bit.GPIO49 = 1;
			GpioDataRegs.GPBSET.bit.GPIO34 = 1;
			GpioDataRegs.GPASET.bit.GPIO11 = 1;
			GpioDataRegs.GPASET.bit.GPIO9 = 1;
			break;
		}
		case 2:
		{
			GpioDataRegs.GPBCLEAR.bit.GPIO49 = 1;
			GpioDataRegs.GPBCLEAR.bit.GPIO34 = 1;
			GpioDataRegs.GPASET.bit.GPIO11 = 1;
			GpioDataRegs.GPASET.bit.GPIO9 = 1;
			break;
		}
		case 1:
		{
			GpioDataRegs.GPBCLEAR.bit.GPIO49 = 1;
			GpioDataRegs.GPBCLEAR.bit.GPIO34 = 1;
			GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;
			GpioDataRegs.GPASET.bit.GPIO9 = 1;
			break;
		}
```

```
            case 0:
            {
                  GpioDataRegs.GPBCLEAR.bit.GPIO49 = 1;
                  GpioDataRegs.GPBCLEAR.bit.GPIO34 = 1;
                  GpioDataRegs.GPACLEAR.bit.GPIO11 = 1;
                  GpioDataRegs.GPACLEAR.bit.GPIO9 = 1;
                  break;
            }
            default:
            {
                  GpioDataRegs.GPBSET.bit.GPIO49 = 1;
                  GpioDataRegs.GPBSET.bit.GPIO34 = 1;
                  GpioDataRegs.GPASET.bit.GPIO11 = 1;
                  GpioDataRegs.GPASET.bit.GPIO9 = 1;
            }
      }
}
```

DSP2833x_ADC.C

```
//###########################################################################
//
// FILE:     DSP2833x_Adc.c
//
// TITLE:    DSP2833x ADC Initialization & Support Functions.
//
//###########################################################################
// $TI Release: DSP2833x/DSP2823x C/C++ Header Files V1.31 $
//###########################################################################

#include "DSP2833x_Device.h"     // DSP2833x Headerfile Include File
#include "DSP2833x_Examples.h"   // DSP2833x Examples Include File

#define ADC_usDELAY  5000L

//---------------------------------------------------------------------------
// InitAdc:
//---------------------------------------------------------------------------
// This function initializes ADC to a known state.
//
void InitAdc(void)
{
    extern void DSP28x_usDelay(Uint32 Count);


    // *IMPORTANT*
      // The ADC_cal function, which  copies the ADC calibration values from TI
reserved
      // OTP into the ADCREFSEL and ADCOFFTRIM registers, occurs automatically in
the
      // Boot ROM. If the boot ROM code is bypassed during the debug process, the
      // following function MUST be called for the ADC to function according
```

```
      // to specification. The clocks to the ADC MUST be enabled before calling
this
      // function.
      // See the device data manual and/or the ADC Reference
      // Manual for more information.

          EALLOW;
             SysCtrlRegs.PCLKCR0.bit.ADCENCLK = 1;
             ADC_cal();
             EDIS;



    // To powerup the ADC the ADCENCLK bit should be set first to enable
    // clocks, followed by powering up the bandgap, reference circuitry, and ADC
core.
    // Before the first conversion is performed a 5ms delay must be observed
      // after power up to give all analog circuits time to power up and settle

    // Please note that for the delay function below to operate correctly the
      // CPU_RATE define statement in the DSP2833x_Examples.h file must
      // contain the correct CPU clock period in nanoseconds.

    AdcRegs.ADCTRL3.all = 0x00E0;  // Power up bandgap/reference/ADC circuits
    DELAY_US(ADC_usDELAY);         // Delay before converting ADC channels
}

//===========================================================================
// End of file.
//===========================================================================
```

DSP2833X_CPUTIMERS.C

```
//###########################################################################
//
// FILE:    DSP2833x_CpuTimers.c
//
// TITLE:   CPU 32-bit Timers Initialization & Support Functions.
//
// NOTES:   CpuTimer2 is reserved for use with DSP BIOS and
//          other realtime operating systems.
//
//          Do not use these this timer in your application if you ever plan
//          on integrating DSP-BIOS or another realtime OS.
//
//###########################################################################
// $TI Release: DSP2833x/DSP2823x C/C++ Header Files V1.31 $
// //###########################################################################
```

```
#include "DSP2833x_Device.h"     // Headerfile Include File
#include "DSP2833x_Examples.h"   // Examples Include File

struct CPUTIMER_VARS CpuTimer0;

// When using DSP BIOS & other RTOS, comment out CPU Timer 2 code.
struct CPUTIMER_VARS CpuTimer1;
struct CPUTIMER_VARS CpuTimer2;

//---------------------------------------------------------------------------
// InitCpuTimers:
//---------------------------------------------------------------------------
// This function initializes all three CPU timers to a known state.
//
void InitCpuTimers(void)
{
    // CPU Timer 0
    // Initialize address pointers to respective timer registers:
    CpuTimer0.RegsAddr = &CpuTimer0Regs;
    // Initialize timer period to maximum:
    CpuTimer0Regs.PRD.all  = 0xFFFFFFFF;
    // Initialize pre-scale counter to divide by 1 (SYSCLKOUT):
    CpuTimer0Regs.TPR.all  = 0;
    CpuTimer0Regs.TPRH.all = 0;
    // Make sure timer is stopped:
    CpuTimer0Regs.TCR.bit.TSS = 1;
    // Reload all counter register with period value:
    CpuTimer0Regs.TCR.bit.TRB = 1;
    // Reset interrupt counters:
    CpuTimer0.InterruptCount = 0;


// CpuTimer2 is reserved for DSP BIOS & other RTOS
// Do not use this timer if you ever plan on integrating
// DSP-BIOS or another realtime OS.

    // Initialize address pointers to respective timer registers:
    CpuTimer1.RegsAddr = &CpuTimer1Regs;
    CpuTimer2.RegsAddr = &CpuTimer2Regs;
    // Initialize timer period to maximum:
    CpuTimer1Regs.PRD.all  = 0xFFFFFFFF;
    CpuTimer2Regs.PRD.all  = 0xFFFFFFFF;
    // Make sure timers are stopped:
    CpuTimer1Regs.TCR.bit.TSS = 1;
    CpuTimer2Regs.TCR.bit.TSS = 1;
    // Reload all counter register with period value:
    CpuTimer1Regs.TCR.bit.TRB = 1;
    CpuTimer2Regs.TCR.bit.TRB = 1;
    // Reset interrupt counters:
    CpuTimer1.InterruptCount = 0;
    CpuTimer2.InterruptCount = 0;

}

//---------------------------------------------------------------------------
// ConfigCpuTimer:
//---------------------------------------------------------------------------
```

```
// This function initializes the selected timer to the period specified
// by the "Freq" and "Period" parameters. The "Freq" is entered as "MHz"
// and the period in "uSeconds". The timer is held in the stopped state
// after configuration.
//
void ConfigCpuTimer(struct CPUTIMER_VARS *Timer, float Freq, float Period)
{
    Uint32  temp;

    // Initialize timer period:
    Timer->CPUFreqInMHz = Freq;
    Timer->PeriodInUSec = Period;
    temp = (long) (Freq * Period);
    Timer->RegsAddr->PRD.all = temp;

    // Set pre-scale counter to divide by 1 (SYSCLKOUT):
    Timer->RegsAddr->TPR.all  = 0;
    Timer->RegsAddr->TPRH.all  = 0;

    // Initialize timer control register:
    Timer->RegsAddr->TCR.bit.TSS = 1;      // 1 = Stop timer, 0 = Start/Restart
Timer
    Timer->RegsAddr->TCR.bit.TRB = 1;      // 1 = reload timer
    Timer->RegsAddr->TCR.bit.SOFT = 1;
    Timer->RegsAddr->TCR.bit.FREE = 1;     // Timer Free Run
    Timer->RegsAddr->TCR.bit.TIE = 1;      // 0 = Disable/ 1 = Enable Timer
Interrupt

    // Reset interrupt counter:
    Timer->InterruptCount = 0;
}

//===========================================================================
// End of file.
//===========================================================================
```

DSP2833x_GlobalVariableDefs.c

```
//###########################################################################
//
// FILE:     DSP2833x_GlobalVariableDefs.c
//
// TITLE:    DSP2833x Global Variables and Data Section Pragmas.
//
//###########################################################################
// $TI Release: DSP2833x/DSP2823x C/C++ Header Files V1.31 $
// $
//###########################################################################

#include "DSP2833x_Device.h"     // DSP2833x Headerfile Include File

//---------------------------------------------------------------------------
// Define Global Peripheral Variables:
//
//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("AdcRegsFile")
```

```
#else
#pragma DATA_SECTION(AdcRegs,"AdcRegsFile");
#endif
volatile struct ADC_REGS AdcRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("AdcMirrorFile")
#else
#pragma DATA_SECTION(AdcMirror,"AdcMirrorFile");
#endif
volatile struct ADC_RESULT_MIRROR_REGS AdcMirror;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("CpuTimer0RegsFile")
#else
#pragma DATA_SECTION(CpuTimer0Regs,"CpuTimer0RegsFile");
#endif
volatile struct CPUTIMER_REGS CpuTimer0Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("CpuTimer1RegsFile")
#else
#pragma DATA_SECTION(CpuTimer1Regs,"CpuTimer1RegsFile");
#endif
volatile struct CPUTIMER_REGS CpuTimer1Regs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("CpuTimer2RegsFile")
#else
#pragma DATA_SECTION(CpuTimer2Regs,"CpuTimer2RegsFile");
#endif
volatile struct CPUTIMER_REGS CpuTimer2Regs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("CsmPwlFile")
#else
#pragma DATA_SECTION(CsmPwl,"CsmPwlFile");
#endif
volatile struct CSM_PWL CsmPwl;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("CsmRegsFile")
#else
#pragma DATA_SECTION(CsmRegs,"CsmRegsFile");
#endif
volatile struct CSM_REGS CsmRegs;
```

```
//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("DevEmuRegsFile")
#else
#pragma DATA_SECTION(DevEmuRegs,"DevEmuRegsFile");
#endif
volatile struct DEV_EMU_REGS DevEmuRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("DmaRegsFile")
#else
#pragma DATA_SECTION(DmaRegs,"DmaRegsFile");
#endif
volatile struct DMA_REGS DmaRegs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanaRegsFile")
#else
#pragma DATA_SECTION(ECanaRegs,"ECanaRegsFile");
#endif
volatile struct ECAN_REGS ECanaRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanaMboxesFile")
#else
#pragma DATA_SECTION(ECanaMboxes,"ECanaMboxesFile");
#endif
volatile struct ECAN_MBOXES ECanaMboxes;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanaLAMRegsFile")
#else
#pragma DATA_SECTION(ECanaLAMRegs,"ECanaLAMRegsFile");
#endif
volatile struct LAM_REGS ECanaLAMRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanaMOTSRegsFile")
#else
#pragma DATA_SECTION(ECanaMOTSRegs,"ECanaMOTSRegsFile");
#endif
volatile struct MOTS_REGS ECanaMOTSRegs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanaMOTORegsFile")
#else
#pragma DATA_SECTION(ECanaMOTORegs,"ECanaMOTORegsFile");
#endif
volatile struct MOTO_REGS ECanaMOTORegs;
```

```
//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanbRegsFile")
#else
#pragma DATA_SECTION(ECanbRegs,"ECanbRegsFile");
#endif
volatile struct ECAN_REGS ECanbRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanbMboxesFile")
#else
#pragma DATA_SECTION(ECanbMboxes,"ECanbMboxesFile");
#endif
volatile struct ECAN_MBOXES ECanbMboxes;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanbLAMRegsFile")
#else
#pragma DATA_SECTION(ECanbLAMRegs,"ECanbLAMRegsFile");
#endif
volatile struct LAM_REGS ECanbLAMRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanbMOTSRegsFile")
#else
#pragma DATA_SECTION(ECanbMOTSRegs,"ECanbMOTSRegsFile");
#endif
volatile struct MOTS_REGS ECanbMOTSRegs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECanbMOTORegsFile")
#else
#pragma DATA_SECTION(ECanbMOTORegs,"ECanbMOTORegsFile");
#endif
volatile struct MOTO_REGS ECanbMOTORegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EPwm1RegsFile")
#else
#pragma DATA_SECTION(EPwm1Regs,"EPwm1RegsFile");
#endif
volatile struct EPWM_REGS EPwm1Regs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EPwm2RegsFile")
#else
#pragma DATA_SECTION(EPwm2Regs,"EPwm2RegsFile");
```

```
#endif
volatile struct EPWM_REGS EPwm2Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EPwm3RegsFile")
#else
#pragma DATA_SECTION(EPwm3Regs,"EPwm3RegsFile");
#endif
volatile struct EPWM_REGS EPwm3Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EPwm4RegsFile")
#else
#pragma DATA_SECTION(EPwm4Regs,"EPwm4RegsFile");
#endif
volatile struct EPWM_REGS EPwm4Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EPwm5RegsFile")
#else
#pragma DATA_SECTION(EPwm5Regs,"EPwm5RegsFile");
#endif
volatile struct EPWM_REGS EPwm5Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EPwm6RegsFile")
#else
#pragma DATA_SECTION(EPwm6Regs,"EPwm6RegsFile");
#endif
volatile struct EPWM_REGS EPwm6Regs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECap1RegsFile")
#else
#pragma DATA_SECTION(ECap1Regs,"ECap1RegsFile");
#endif
volatile struct ECAP_REGS ECap1Regs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECap2RegsFile")
#else
#pragma DATA_SECTION(ECap2Regs,"ECap2RegsFile");
#endif
volatile struct ECAP_REGS ECap2Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECap3RegsFile")
#else
```

```
#pragma DATA_SECTION(ECap3Regs,"ECap3RegsFile");
#endif
volatile struct ECAP_REGS ECap3Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECap4RegsFile")
#else
#pragma DATA_SECTION(ECap4Regs,"ECap4RegsFile");
#endif
volatile struct ECAP_REGS ECap4Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECap5RegsFile")
#else
#pragma DATA_SECTION(ECap5Regs,"ECap5RegsFile");
#endif
volatile struct ECAP_REGS ECap5Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ECap6RegsFile")
#else
#pragma DATA_SECTION(ECap6Regs,"ECap6RegsFile");
#endif
volatile struct ECAP_REGS ECap6Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EQep1RegsFile")
#else
#pragma DATA_SECTION(EQep1Regs,"EQep1RegsFile");
#endif
volatile struct EQEP_REGS EQep1Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("EQep2RegsFile")
#else
#pragma DATA_SECTION(EQep2Regs,"EQep2RegsFile");
#endif
volatile struct EQEP_REGS EQep2Regs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("GpioCtrlRegsFile")
#else
#pragma DATA_SECTION(GpioCtrlRegs,"GpioCtrlRegsFile");
#endif
volatile struct GPIO_CTRL_REGS GpioCtrlRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("GpioDataRegsFile")
#else
#pragma DATA_SECTION(GpioDataRegs,"GpioDataRegsFile");
```

```
#endif
volatile struct GPIO_DATA_REGS GpioDataRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("GpioIntRegsFile")
#else
#pragma DATA_SECTION(GpioIntRegs,"GpioIntRegsFile");
#endif
volatile struct GPIO_INT_REGS GpioIntRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("I2caRegsFile")
#else
#pragma DATA_SECTION(I2caRegs,"I2caRegsFile");
#endif
volatile struct I2C_REGS I2caRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("McbspaRegsFile")
#else
#pragma DATA_SECTION(McbspaRegs,"McbspaRegsFile");
#endif
volatile struct MCBSP_REGS McbspaRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("McbspbRegsFile")
#else
#pragma DATA_SECTION(McbspbRegs,"McbspbRegsFile");
#endif
volatile struct MCBSP_REGS McbspbRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("PartIdRegsFile")
#else
#pragma DATA_SECTION(PartIdRegs,"PartIdRegsFile");
#endif
volatile struct PARTID_REGS PartIdRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("PieCtrlRegsFile")
#else
#pragma DATA_SECTION(PieCtrlRegs,"PieCtrlRegsFile");
#endif
volatile struct PIE_CTRL_REGS PieCtrlRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("PieVectTableFile")
#else
#pragma DATA_SECTION(PieVectTable,"PieVectTableFile");
#endif
```

```
struct PIE_VECT_TABLE PieVectTable;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("SciaRegsFile")
#else
#pragma DATA_SECTION(SciaRegs,"SciaRegsFile");
#endif
volatile struct SCI_REGS SciaRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ScibRegsFile")
#else
#pragma DATA_SECTION(ScibRegs,"ScibRegsFile");
#endif
volatile struct SCI_REGS ScibRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("ScicRegsFile")
#else
#pragma DATA_SECTION(ScicRegs,"ScicRegsFile");
#endif
volatile struct SCI_REGS ScicRegs;


//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("SpiaRegsFile")
#else
#pragma DATA_SECTION(SpiaRegs,"SpiaRegsFile");
#endif
volatile struct SPI_REGS SpiaRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("SysCtrlRegsFile")
#else
#pragma DATA_SECTION(SysCtrlRegs,"SysCtrlRegsFile");
#endif
volatile struct SYS_CTRL_REGS SysCtrlRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("FlashRegsFile")
#else
#pragma DATA_SECTION(FlashRegs,"FlashRegsFile");
#endif
volatile struct FLASH_REGS FlashRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("XIntruptRegsFile")
#else
#pragma DATA_SECTION(XIntruptRegs,"XIntruptRegsFile");
#endif
```

```
volatile struct XINTRUPT_REGS XIntruptRegs;

//---------------------------------------
#ifdef __cplusplus
#pragma DATA_SECTION("XintfRegsFile")
#else
#pragma DATA_SECTION(XintfRegs,"XintfRegsFile");
#endif
volatile struct XINTF_REGS XintfRegs;



//===========================================================================
// End of file.
//===========================================================================
```

DSP2833x_PieCtrl.c

```
//###########################################################################
//
// FILE:     DSP2833x_PieCtrl.c
//
// TITLE:    DSP2833x Device PIE Control Register Initialization Functions.
//
//###########################################################################
// $TI Release: DSP2833x/DSP2823x C/C++ Header Files V1.31 $
//###########################################################################

#include "DSP2833x_Device.h"     // DSP2833x Headerfile Include File
#include "DSP2833x_Examples.h"   // DSP2833x Examples Include File

//---------------------------------------------------------------------------
// InitPieCtrl:
//---------------------------------------------------------------------------
// This function initializes the PIE control registers to a known state.
//
void InitPieCtrl(void)
{
    // Disable Interrupts at the CPU level:
    DINT;

    // Disable the PIE
    PieCtrlRegs.PIECTRL.bit.ENPIE = 0;

      // Clear all PIEIER registers:
      PieCtrlRegs.PIEIER1.all = 0;
      PieCtrlRegs.PIEIER2.all = 0;
      PieCtrlRegs.PIEIER3.all = 0;
      PieCtrlRegs.PIEIER4.all = 0;
      PieCtrlRegs.PIEIER5.all = 0;
      PieCtrlRegs.PIEIER6.all = 0;
      PieCtrlRegs.PIEIER7.all = 0;
      PieCtrlRegs.PIEIER8.all = 0;
      PieCtrlRegs.PIEIER9.all = 0;
      PieCtrlRegs.PIEIER10.all = 0;
```

```
       PieCtrlRegs.PIEIER11.all = 0;
       PieCtrlRegs.PIEIER12.all = 0;

       // Clear all PIEIFR registers:
       PieCtrlRegs.PIEIFR1.all = 0;
       PieCtrlRegs.PIEIFR2.all = 0;
       PieCtrlRegs.PIEIFR3.all = 0;
       PieCtrlRegs.PIEIFR4.all = 0;
       PieCtrlRegs.PIEIFR5.all = 0;
       PieCtrlRegs.PIEIFR6.all = 0;
       PieCtrlRegs.PIEIFR7.all = 0;
       PieCtrlRegs.PIEIFR8.all = 0;
       PieCtrlRegs.PIEIFR9.all = 0;
       PieCtrlRegs.PIEIFR10.all = 0;
       PieCtrlRegs.PIEIFR11.all = 0;
       PieCtrlRegs.PIEIFR12.all = 0;


}

//---------------------------------------------------------------------------
// EnableInterrupts:
//---------------------------------------------------------------------------
// This function enables the PIE module and CPU interrupts
//
void EnableInterrupts()
{

    // Enable the PIE
    PieCtrlRegs.PIECTRL.bit.ENPIE = 1;

       // Enables PIE to drive a pulse into the CPU
       PieCtrlRegs.PIEACK.all = 0xFFFF;

       // Enable Interrupts at the CPU level
    EINT;

}


//===========================================================================
// End of file.
//===========================================================================
```

DSP2833X_PIEVECT.C

```
//###########################################################################
//
// FILE:     DSP2833x_PieVect.c
//
// TITLE:    DSP2833x Devices PIE Vector Table Initialization Functions.
//
//###########################################################################
// $TI Release: DSP2833x/DSP2823x C/C++ Header Files V1.31 $
////#########################################################################
```

```
#include "DSP2833x_Device.h"     // DSP2833x Headerfile Include File
#include "DSP2833x_Examples.h"   // DSP2833x Examples Include File


const struct PIE_VECT_TABLE PieVectTableInit = {

      PIE_RESERVED,  // 0  Reserved space
      PIE_RESERVED,  // 1  Reserved space
      PIE_RESERVED,  // 2  Reserved space
      PIE_RESERVED,  // 3  Reserved space
      PIE_RESERVED,  // 4  Reserved space
      PIE_RESERVED,  // 5  Reserved space
      PIE_RESERVED,  // 6  Reserved space
      PIE_RESERVED,  // 7  Reserved space
      PIE_RESERVED,  // 8  Reserved space
      PIE_RESERVED,  // 9  Reserved space
      PIE_RESERVED,  // 10 Reserved space
      PIE_RESERVED,  // 11 Reserved space
      PIE_RESERVED,  // 12 Reserved space


// Non-Peripheral Interrupts
      INT13_ISR,     // XINT13 or CPU-Timer 1
      INT14_ISR,     // CPU-Timer2
      DATALOG_ISR,   // Datalogging interrupt
      RTOSINT_ISR,   // RTOS interrupt
      EMUINT_ISR,    // Emulation interrupt
      NMI_ISR,       // Non-maskable interrupt
      ILLEGAL_ISR,   // Illegal operation TRAP
      USER1_ISR,     // User Defined trap 1
      USER2_ISR,     // User Defined trap 2
      USER3_ISR,     // User Defined trap 3
      USER4_ISR,     // User Defined trap 4
      USER5_ISR,     // User Defined trap 5
      USER6_ISR,     // User Defined trap 6
      USER7_ISR,     // User Defined trap 7
      USER8_ISR,     // User Defined trap 8
      USER9_ISR,     // User Defined trap 9
      USER10_ISR,    // User Defined trap 10
      USER11_ISR,    // User Defined trap 11
      USER12_ISR,    // User Defined trap 12

// Group 1 PIE Vectors
      SEQ1INT_ISR,     // 1.1 ADC
      SEQ2INT_ISR,     // 1.2 ADC
      rsvd_ISR,        // 1.3
      XINT1_ISR,       // 1.4
      XINT2_ISR,       // 1.5
      ADCINT_ISR,      // 1.6 ADC
      TINT0_ISR,       // 1.7 Timer 0
      WAKEINT_ISR,     // 1.8 WD, Low Power

// Group 2 PIE Vectors
      EPWM1_TZINT_ISR, // 2.1 EPWM-1 Trip Zone
      EPWM2_TZINT_ISR, // 2.2 EPWM-2 Trip Zone
      EPWM3_TZINT_ISR, // 2.3 EPWM-3 Trip Zone
      EPWM4_TZINT_ISR, // 2.4 EPWM-4 Trip Zone
```

```
        EPWM5_TZINT_ISR, // 2.5 EPWM-5 Trip Zone
        EPWM6_TZINT_ISR, // 2.6 EPWM-6 Trip Zone
        rsvd_ISR,        // 2.7
        rsvd_ISR,        // 2.8

    // Group 3 PIE Vectors
        EPWM1_INT_ISR,   // 3.1 EPWM-1 Interrupt
        EPWM2_INT_ISR,   // 3.2 EPWM-2 Interrupt
        EPWM3_INT_ISR,   // 3.3 EPWM-3 Interrupt
        EPWM4_INT_ISR,   // 3.4 EPWM-4 Interrupt
        EPWM5_INT_ISR,   // 3.5 EPWM-5 Interrupt
        EPWM6_INT_ISR,   // 3.6 EPWM-6 Interrupt
        rsvd_ISR,        // 3.7
        rsvd_ISR,        // 3.8

    // Group 4 PIE Vectors
        ECAP1_INT_ISR,   // 4.1 ECAP-1
        ECAP2_INT_ISR,   // 4.2 ECAP-2
        ECAP3_INT_ISR,   // 4.3 ECAP-3
        ECAP4_INT_ISR,   // 4.4 ECAP-4
        ECAP5_INT_ISR,   // 4.5 ECAP-5
        ECAP6_INT_ISR,   // 4.6 ECAP-6
        rsvd_ISR,        // 4.7
        rsvd_ISR,        // 4.8

    // Group 5 PIE Vectors
        EQEP1_INT_ISR,   // 5.1 EQEP-1
        EQEP2_INT_ISR,   // 5.2 EQEP-2
        rsvd_ISR,        // 5.3
        rsvd_ISR,        // 5.4
        rsvd_ISR,        // 5.5
        rsvd_ISR,        // 5.6
        rsvd_ISR,        // 5.7
        rsvd_ISR,        // 5.8


    // Group 6 PIE Vectors
        SPIRXINTA_ISR,   // 6.1 SPI-A
        SPITXINTA_ISR,   // 6.2 SPI-A
        MRINTA_ISR,      // 6.3 McBSP-A
        MXINTA_ISR,      // 6.4 McBSP-A
        MRINTB_ISR,      // 6.5 McBSP-B
        MXINTB_ISR,      // 6.6 McBSP-B
        rsvd_ISR,        // 6.7
        rsvd_ISR,        // 6.8


    // Group 7 PIE Vectors
        DINTCH1_ISR,     // 7.1  DMA channel 1
        DINTCH2_ISR,     // 7.2  DMA channel 2
        DINTCH3_ISR,     // 7.3  DMA channel 3
        DINTCH4_ISR,     // 7.4  DMA channel 4
        DINTCH5_ISR,     // 7.5  DMA channel 5
        DINTCH6_ISR,     // 7.6  DMA channel 6
        rsvd_ISR,        // 7.7
        rsvd_ISR,        // 7.8
```

```
// Group 8 PIE Vectors
      I2CINT1A_ISR,     // 8.1  I2C
      I2CINT2A_ISR,     // 8.2  I2C
      rsvd_ISR,         // 8.3
      rsvd_ISR,         // 8.4
      SCIRXINTC_ISR,    // 8.5  SCI-C
      SCITXINTC_ISR,    // 8.6  SCI-C
      rsvd_ISR,         // 8.7
      rsvd_ISR,         // 8.8

// Group 9 PIE Vectors
      SCIRXINTA_ISR,    // 9.1 SCI-A
      SCITXINTA_ISR,    // 9.2 SCI-A
      SCIRXINTB_ISR,    // 9.3 SCI-B
      SCITXINTB_ISR,    // 9.4 SCI-B
      ECAN0INTA_ISR,    // 9.5 eCAN-A
      ECAN1INTA_ISR,    // 9.6 eCAN-A
      ECAN0INTB_ISR,    // 9.7 eCAN-B
      ECAN1INTB_ISR,    // 9.8 eCAN-B

// Group 10 PIE Vectors
      rsvd_ISR,         // 10.1
      rsvd_ISR,         // 10.2
      rsvd_ISR,         // 10.3
      rsvd_ISR,         // 10.4
      rsvd_ISR,         // 10.5
      rsvd_ISR,         // 10.6
      rsvd_ISR,         // 10.7
      rsvd_ISR,         // 10.8

// Group 11 PIE Vectors
      rsvd_ISR,         // 11.1
      rsvd_ISR,         // 11.2
      rsvd_ISR,         // 11.3
      rsvd_ISR,         // 11.4
      rsvd_ISR,         // 11.5
      rsvd_ISR,         // 11.6
      rsvd_ISR,         // 11.7
      rsvd_ISR,         // 11.8

// Group 12 PIE Vectors
      XINT3_ISR,        // 12.1
      XINT4_ISR,        // 12.2
      XINT5_ISR,        // 12.3
      XINT6_ISR,        // 12.4
      XINT7_ISR,        // 12.5
      rsvd_ISR,         // 12.6
      LVF_ISR,          // 12.7
      LUF_ISR,          // 12.8
};


//---------------------------------------------------------------------------
// InitPieVectTable:
//---------------------------------------------------------------------------
// This function initializes the PIE vector table to a known state.
// This function must be executed after boot time.
```

```
//

void InitPieVectTable(void)
{
	int16  i;
	Uint32 *Source = (void *) &PieVectTableInit;
	Uint32 *Dest = (void *) &PieVectTable;

	EALLOW;
	for(i=0; i < 128; i++)
		*Dest++ = *Source++;
	EDIS;

	// Enable the PIE Vector Table
	PieCtrlRegs.PIECTRL.bit.ENPIE = 1;

}

//===========================================================================
// End of file.
//===========================================================================
```

DSP2833X_SYSCTRL.C

```
//###########################################################################
//
// FILE:   DSP2833x_SysCtrl.c
//
// TITLE:  DSP2833x Device System Control Initialization & Support Functions.
//
// DESCRIPTION:
//
//         Example initialization of system resources.
//
//###########################################################################
// $TI Release: DSP2833x/DSP2823x C/C++ Header Files V1.31 $
//
//###########################################################################


#include "DSP2833x_Device.h"     // Headerfile Include File
#include "DSP2833x_Examples.h"   // Examples Include File

// Functions that will be run from RAM need to be assigned to
// a different section.  This section will then be mapped to a load and
// run address using the linker cmd file.

#pragma CODE_SECTION(InitFlash, "ramfuncs");

//---------------------------------------------------------------------------
// InitSysCtrl:
//---------------------------------------------------------------------------
// This function initializes the System Control registers to a known state.
// - Disables the watchdog
// - Set the PLLCR for proper SYSCLKOUT frequency
// - Set the pre-scaler for the high and low frequency peripheral clocks
// - Enable the clocks to the peripherals
```

```
void InitSysCtrl(void)
{

   // Disable the watchdog
   DisableDog();

   // Initialize the PLL control: PLLCR and DIVSEL
   // DSP28_PLLCR and DSP28_DIVSEL are defined in DSP2833x_Examples.h
   InitPll(DSP28_PLLCR,DSP28_DIVSEL);

   // Initialize the peripheral clocks
   InitPeripheralClocks();
}


//---------------------------------------------------------------------------
// Example: InitFlash:
//---------------------------------------------------------------------------
// This function initializes the Flash Control registers

//                   CAUTION
// This function MUST be executed out of RAM. Executing it
// out of OTP/Flash will yield unpredictable results

void InitFlash(void)
{
   EALLOW;
   //Enable Flash Pipeline mode to improve performance
   //of code executed from Flash.
   FlashRegs.FOPT.bit.ENPIPE = 1;

   //                CAUTION
   //Minimum waitstates required for the flash operating
   //at a given CPU rate must be characterized by TI.
   //Refer to the datasheet for the latest information.
#if CPU_FRQ_150MHZ
   //Set the Paged Waitstate for the Flash
   FlashRegs.FBANKWAIT.bit.PAGEWAIT = 5;

   //Set the Random Waitstate for the Flash
   FlashRegs.FBANKWAIT.bit.RANDWAIT = 5;

   //Set the Waitstate for the OTP
   FlashRegs.FOTPWAIT.bit.OTPWAIT = 8;
#endif

#if CPU_FRQ_100MHZ
   //Set the Paged Waitstate for the Flash
   FlashRegs.FBANKWAIT.bit.PAGEWAIT = 3;

   //Set the Random Waitstate for the Flash
   FlashRegs.FBANKWAIT.bit.RANDWAIT = 3;

   //Set the Waitstate for the OTP
   FlashRegs.FOTPWAIT.bit.OTPWAIT = 5;
#endif
```

```
   //               CAUTION
   //ONLY THE DEFAULT VALUE FOR THESE 2 REGISTERS SHOULD BE USED
   FlashRegs.FSTDBYWAIT.bit.STDBYWAIT = 0x01FF;
   FlashRegs.FACTIVEWAIT.bit.ACTIVEWAIT = 0x01FF;
   EDIS;

   //Force a pipeline flush to ensure that the write to
   //the last register configured occurs before returning.

   asm(" RPT #7 || NOP");
}


//---------------------------------------------------------------------------
// Example: ServiceDog:
//---------------------------------------------------------------------------
// This function resets the watchdog timer.
// Enable this function for using ServiceDog in the application

void ServiceDog(void)
{
    EALLOW;
    SysCtrlRegs.WDKEY = 0x0055;
    SysCtrlRegs.WDKEY = 0x00AA;
    EDIS;
}

//---------------------------------------------------------------------------
// Example: DisableDog:
//---------------------------------------------------------------------------
// This function disables the watchdog timer.

void DisableDog(void)
{
    EALLOW;
    SysCtrlRegs.WDCR= 0x0068;
    EDIS;
}

//---------------------------------------------------------------------------
// Example: InitPll:
//---------------------------------------------------------------------------
// This function initializes the PLLCR register.

void InitPll(Uint16 val, Uint16 divsel)
{

   // Make sure the PLL is not running in limp mode
   if (SysCtrlRegs.PLLSTS.bit.MCLKSTS != 0)
   {
      // Missing external clock has been detected
      // Replace this line with a call to an appropriate
      // SystemShutdown(); function.
      asm("        ESTOP0");
   }

   // DIVSEL MUST be 0 before PLLCR can be changed from
```

```
// 0x0000. It is set to 0 by an external reset XRSn
// This puts us in 1/4
if (SysCtrlRegs.PLLSTS.bit.DIVSEL != 0)
{
    EALLOW;
    SysCtrlRegs.PLLSTS.bit.DIVSEL = 0;
    EDIS;
}

// Change the PLLCR
if (SysCtrlRegs.PLLCR.bit.DIV != val)
{

   EALLOW;
   // Before setting PLLCR turn off missing clock detect logic
   SysCtrlRegs.PLLSTS.bit.MCLKOFF = 1;
   SysCtrlRegs.PLLCR.bit.DIV = val;
   EDIS;

   // Optional: Wait for PLL to lock.
   // During this time the CPU will switch to OSCCLK/2 until
   // the PLL is stable.  Once the PLL is stable the CPU will
   // switch to the new PLL value.
   //
   // This time-to-lock is monitored by a PLL lock counter.
   //
   // Code is not required to sit and wait for the PLL to lock.
   // However, if the code does anything that is timing critical,
   // and requires the correct clock be locked, then it is best to
   // wait until this switching has completed.

   // Wait for the PLL lock bit to be set.

   // The watchdog should be disabled before this loop, or fed within
   // the loop via ServiceDog().

     // Uncomment to disable the watchdog
   DisableDog();

   while(SysCtrlRegs.PLLSTS.bit.PLLLOCKS != 1)
   {
         // Uncomment to service the watchdog
       // ServiceDog();
   }

   EALLOW;
   SysCtrlRegs.PLLSTS.bit.MCLKOFF = 0;
   EDIS;
 }

 // If switching to 1/2
   if((divsel == 1)||(divsel == 2))
   {
         EALLOW;
       SysCtrlRegs.PLLSTS.bit.DIVSEL = divsel;
       EDIS;
   }
```

```
       // NOTE: ONLY USE THIS SETTING IF PLL IS BYPASSED (I.E. PLLCR = 0) OR OFF
       // If switching to 1/1
       // * First go to 1/2 and let the power settle
       //   The time required will depend on the system, this is only an example
       // * Then switch to 1/1
       if(divsel == 3)
       {
              EALLOW;
           SysCtrlRegs.PLLSTS.bit.DIVSEL = 2;
           DELAY_US(50L);
           SysCtrlRegs.PLLSTS.bit.DIVSEL = 3;
           EDIS;
    }
}

//---------------------------------------------------------------------------
// Example: InitPeripheralClocks:
//---------------------------------------------------------------------------
// This function initializes the clocks to the peripheral modules.
// First the high and low clock prescalers are set
// Second the clocks are enabled to each peripheral.
// To reduce power, leave clocks to unused peripherals disabled
//
// Note: If a peripherals clock is not enabled then you cannot
// read or write to the registers for that peripheral

void InitPeripheralClocks(void)
{
   EALLOW;

// HISPCP/LOSPCP prescale register settings, normally it will be set to default
values
   SysCtrlRegs.HISPCP.all = 0x0001;
   SysCtrlRegs.LOSPCP.all = 0x0002;

// XCLKOUT to SYSCLKOUT ratio.  By default XCLKOUT = 1/4 SYSCLKOUT
   // XTIMCLK = SYSCLKOUT/2
   XintfRegs.XINTCNF2.bit.XTIMCLK = 1;
   // XCLKOUT = XTIMCLK/2
   XintfRegs.XINTCNF2.bit.CLKMODE = 1;
   // Enable XCLKOUT
   XintfRegs.XINTCNF2.bit.CLKOFF = 0;

// Peripheral clock enables set for the selected peripherals.
// If you are not using a peripheral leave the clock off
// to save on power.
//
// Note: not all peripherals are available on all 2833x derivates.
// Refer to the datasheet for your particular device.
//
// This function is not written to be an example of efficient code.

   SysCtrlRegs.PCLKCR0.bit.ADCENCLK = 1;    // ADC

   // *IMPORTANT*
```

```
   // The ADC_cal function, which  copies the ADC calibration values from TI
reserved
   // OTP into the ADCREFSEL and ADCOFFTRIM registers, occurs automatically in the
   // Boot ROM. If the boot ROM code is bypassed during the debug process, the
   // following function MUST be called for the ADC to function according
   // to specification. The clocks to the ADC MUST be enabled before calling this
   // function.
   // See the device data manual and/or the ADC Reference
   // Manual for more information.

   ADC_cal();


   SysCtrlRegs.PCLKCR0.bit.I2CAENCLK = 1;   // I2C
   SysCtrlRegs.PCLKCR0.bit.SCIAENCLK = 1;   // SCI-A
   SysCtrlRegs.PCLKCR0.bit.SCIBENCLK = 1;   // SCI-B
   SysCtrlRegs.PCLKCR0.bit.SCICENCLK = 1;   // SCI-C
   SysCtrlRegs.PCLKCR0.bit.SPIAENCLK = 1;   // SPI-A
   SysCtrlRegs.PCLKCR0.bit.MCBSPAENCLK = 1; // McBSP-A
   SysCtrlRegs.PCLKCR0.bit.MCBSPBENCLK = 1; // McBSP-B
   SysCtrlRegs.PCLKCR0.bit.ECANAENCLK=1;    // eCAN-A
   SysCtrlRegs.PCLKCR0.bit.ECANBENCLK=1;    // eCAN-B

   SysCtrlRegs.PCLKCR0.bit.TBCLKSYNC = 0;   // Disable TBCLK within the ePWM
   SysCtrlRegs.PCLKCR1.bit.EPWM1ENCLK = 1;  // ePWM1
   SysCtrlRegs.PCLKCR1.bit.EPWM2ENCLK = 1;  // ePWM2
   SysCtrlRegs.PCLKCR1.bit.EPWM3ENCLK = 1;  // ePWM3
   SysCtrlRegs.PCLKCR1.bit.EPWM4ENCLK = 1;  // ePWM4
   SysCtrlRegs.PCLKCR1.bit.EPWM5ENCLK = 1;  // ePWM5
   SysCtrlRegs.PCLKCR1.bit.EPWM6ENCLK = 1;  // ePWM6
   SysCtrlRegs.PCLKCR0.bit.TBCLKSYNC = 1;   // Enable TBCLK within the ePWM

   SysCtrlRegs.PCLKCR1.bit.ECAP3ENCLK = 1;  // eCAP3
   SysCtrlRegs.PCLKCR1.bit.ECAP4ENCLK = 1;  // eCAP4
   SysCtrlRegs.PCLKCR1.bit.ECAP5ENCLK = 1;  // eCAP5
   SysCtrlRegs.PCLKCR1.bit.ECAP6ENCLK = 1;  // eCAP6
   SysCtrlRegs.PCLKCR1.bit.ECAP1ENCLK = 1;  // eCAP1
   SysCtrlRegs.PCLKCR1.bit.ECAP2ENCLK = 1;  // eCAP2
   SysCtrlRegs.PCLKCR1.bit.EQEP1ENCLK = 1;  // eQEP1
   SysCtrlRegs.PCLKCR1.bit.EQEP2ENCLK = 1;  // eQEP2

   SysCtrlRegs.PCLKCR3.bit.CPUTIMER0ENCLK = 1; // CPU Timer 0
   SysCtrlRegs.PCLKCR3.bit.CPUTIMER1ENCLK = 1; // CPU Timer 1
   SysCtrlRegs.PCLKCR3.bit.CPUTIMER2ENCLK = 1; // CPU Timer 2

   SysCtrlRegs.PCLKCR3.bit.DMAENCLK = 1;       // DMA Clock
   SysCtrlRegs.PCLKCR3.bit.XINTFENCLK = 1;     // XTIMCLK
   SysCtrlRegs.PCLKCR3.bit.GPIOINENCLK = 1;    // GPIO input clock

   EDIS;
}

//---------------------------------------------------------------------------
// Example: CsmUnlock:
//---------------------------------------------------------------------------
// This function unlocks the CSM. User must replace 0xFFFF's with current
// password for the DSP. Returns 1 if unlock is successful.
```

```
#define STATUS_FAIL         0
#define STATUS_SUCCESS      1

Uint16 CsmUnlock()
{
    volatile Uint16 temp;

    // Load the key registers with the current password. The 0xFFFF's are dummy
      // passwords.  User should replace them with the correct password for the
DSP.

    EALLOW;
    CsmRegs.KEY0 = 0xFFFF;
    CsmRegs.KEY1 = 0xFFFF;
    CsmRegs.KEY2 = 0xFFFF;
    CsmRegs.KEY3 = 0xFFFF;
    CsmRegs.KEY4 = 0xFFFF;
    CsmRegs.KEY5 = 0xFFFF;
    CsmRegs.KEY6 = 0xFFFF;
    CsmRegs.KEY7 = 0xFFFF;
    EDIS;

    // Perform a dummy read of the password locations
    // if they match the key values, the CSM will unlock

    temp = CsmPwl.PSWD0;
    temp = CsmPwl.PSWD1;
    temp = CsmPwl.PSWD2;
    temp = CsmPwl.PSWD3;
    temp = CsmPwl.PSWD4;
    temp = CsmPwl.PSWD5;
    temp = CsmPwl.PSWD6;
    temp = CsmPwl.PSWD7;

    // If the CSM unlocked, return succes, otherwise return
    // failure.
    if (CsmRegs.CSMSCR.bit.SECURE == 0) return STATUS_SUCCESS;
    else return STATUS_FAIL;

}


//===========================================================================
// //===========================================================================
```