DEVELOPMENT AND PROTOTYPING OF A GROUND FAULT CIRCUIT
INTERRUPTER FOR 3-PHASE 480 VOLT SYSTEMS

A Thesis
presented to
the Faculty of California Polytechnic State University,
San Luis Obispo

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Electrical Engineering

by
Matthew Ryan Keane Norlander

October 2011

© 2011

Matthew Ryan Keane Norlander

ALL RIGHTS RESERVED

COMMITTEE MEMBERSHIP

TITLE: Development and Prototyping of a Ground Fault Circuit
Interrupter for 3-Phase 480 Volt Systems

AUTHOR: Matthew Ryan Keane Norlander

DATE SUBMITTED: October 2011

COMMITTEE CHAIR: Dr. Dale Dolan, Professor

COMMITTEE MEMBER: Dr. Ahmad Nafisi, Professor

COMMITTEE MEMBER: Dr. Taufik, Professor

ABSTRACT

Development and Prototyping of a Ground Fault Circuit Interrupter for 3-Phase 480 Volt Systems

Matthew Ryan Keane Norlander

Ground Fault Circuit Interrupter (GFCI) technology was first introduced in the NEC in 1971, yet four decades later the technology has not been introduced to a great extent outside of the home environment. This thesis introduces the difficulties encountered in low-voltage three phase ground fault current detection, and adopts a previously patented tripping scheme to develop and prototype a digital relay for 3-phase 480 volt systems capable of the sensitivity and speed required for personnel safety. Results demonstrate the feasibility of the concept and suggest commercial development should be pursued. The prototype is capable of mA sensitivity and reliably detects and removes the faulted feeder from the system, without causing false tripping on non-faulted feeders. The prototype system has been tested and responds appropriately for faults over the 1 mA to 1000 A range.

Keywords: Ground Fault Circuit Interrupter, 3-Phase, 480 Volt, 480V

ACKNOWLEDGEMENTS

Many thanks to Paul Hamer, Mike Fousha and the Chevron Energy Technology Company, for providing the funding without which the project would not have been possible. Thanks to Dr. Dale Dolan, for all the advice, support, and help throughout the last year on this project, and for giving me the confidence that I could finish. Thanks to Dr. Prodanov, Dr. Oliver, and Dr. DePiero, for all of the support in and out of office hours. Thanks to Adrian Smet, Jaime Carmo, and Steven Roy, for help on various portions of the project.

TABLE OF CONTENTS

LIST OF TABLES	ix
LIST OF FIGURES	x
CHAPTER 1: INTRODUCTION	1
Electrocution and Electrical Safety	1
Current GFCI Technology	3
UL 943 Class A Tripping Curve	4
CHAPTER 2: 3-PHASE GFCI DESIGN ISSUES	6
Line Charging Currents	6
Effect of Line Charging Currents on CBCT Measurements	9
Hamer's Solution: The 3-Phase GFCI Digital Relay	10
CBCT Measurements	11
Effects of System Grounding	13
CHAPTER 3: EMTP COMPUTER SIMULATIONS	16
CHAPTER 4: PROPOSED SENSING SOLUTION	20
Simplification of Sensing Requirements	20
CT Solution	21
Digital Filter	28
Full-Cycle Cosine Filter	28
Half-Cycle Cosine Filter	33
CHAPTER 5: DETAILS OF TEST SYSTEM	36
Design and Construction	36
Wiring	37
Feeder Models	39
Source Transformer	41
Contactors	42
Relay Driver PCB	43
DC Power Supply	45
Load Banks	46
Neutral Grounding Resistor	48
Fault Current Generation	49

Resolution of EMI Issues via Phase Restricted Switching	52
Verification of System Properties	54
Feeder Energization Error Currents.....	54
Transient Characteristics of System	56
Verification of Steady-State Nuisance Tripping Currents.....	62
CHAPTER 6: DETAILS OF 3-PHASE GFCI DIGITAL RELAY	66
Hardware	66
ADSP-BF538F Processor	67
Sensor PCB design	69
Software	71
Variable Length Filter Delay	71
C Code Summary.....	73
Signal Chain Characterization.....	78
CHAPTER 7: HIGH CURRENT AND PREVENTATIVE NUISANCE TRIPPING	
TEST RESULTS.....	88
Buck-Boost Transformer High Current Testing.....	88
System Characterization.....	92
Variability in Tripping Time due to Feeder, Phase, and Magnitude of Fault.....	92
Small Fault Current Resolution	95
Test Cases	99
Remnant Flux	104
CHAPTER 8: CONCLUSIONS	112
BIBLIOGRAPHY.....	114
APPENDICES	116
Appendix A - EMTP vs. V-HARM Verification Simulations	116
Appendix B - Ungrounded System Tripping Logic Solution	124
Appendix C - CT Saturation Characteristics.....	127
CT Behavior in Extreme Saturation	127
Appendix D - W-20 Characterization Data.....	131
Saturation Current and Burden	137
Excitation Curve	137
Modeling of a Core.....	141

Appendix E - EMTP Simulations vs. Test System Measurements	144
Appendix F - Prototype C Code	148

LIST OF TABLES

Table 1: Burden Selection Parameters	82
Table 2: Steady-State Filter Characteristics.....	84
Table 3: Filter 1 A Harmonic Rejection Data	86
Table 4: Relay Response Time for Variation in Fault Magnitude, Feeder, and Phase.....	93
Table 5: Small Fault Current System Response	95
Table 6: Selectivity and Interrupting Time, Case A-1 (Solidly Ground System, Balanced Load)	100
Table 7: Selectivity and Interrupting Time, Case A-4 (Solidly Grounded System, No Load)	101
Table 8: Selectivity and Interrupting Time, Case A-7 (Solidly Grounded System, Varied Load)	102
Table 9: Selectivity and Interrupting Time, Case A-2 (HRG System, Balanced Load).	103
Table 10: Selectivity and Interrupting Time, Case A-5 (HRG System, No Load).....	103
Table 11: Turns Ratio Correction	133
Table 12: W-20 Saturation Current for Various Burdens	137

LIST OF FIGURES

Figure 1: UL943 Class A Tripping Curve	4
Figure 2: Cable Line Charging Currents [6]	7
Figure 3: Charging Currents for Motors, Generators, and Surge Capacitors [6]	8
Figure 4: Electrical Schematic for EMTP vs. V-HARM Comparison	17
Figure 5: CT Preservation of Volt-seconds [12]	23
Figure 6: Operating Time for CT with Passive Burdens During Large Fault	24
Figure 7: TVS Diodes Clamping Saturated CT Output	25
Figure 8: Volt-second Area Preservation with TVS Diode Clamping	26
Figure 9: Analog Frequency Response (Magnitude) of Filter	30
Figure 10: PSOC Hardware Data- 66 Hz Filter Input and Output	31
Figure 11: Analog Frequency Response (dB) of Filter	32
Figure 12: Half-Cycle Cosine Filter Steady-State Response	33
Figure 13: Half-Cycle Cosine Filter Transient Response; No DC Offset	34
Figure 14: Half-Cycle Cosine Filter Transient Response; DC offset = 1/2 Peak 60Hz Input	34
Figure 15: Signal Chain Hardware Block Diagram	35
Figure 16: 480V Test System One-Line Diagram [10]	36
Figure 17: Overhead view of 480 V Test System	37
Figure 18: Wiring Schematic for 480 V Test System	38
Figure 19: Nominal- π 3-Phase Mutually Coupled Cable Network	39
Figure 20: Feeder Model Close-up (Feeder 4 shown)	41

Figure 21: 15 kVA 240-480/277 V Δ -Y Transformer with Internal Wiring and Side Connections.....	42
Figure 22: Fuji SC-E05 Contactor	43
Figure 23: Relay Driver PCB Layout	44
Figure 24: Assembled Relay Driver PCB.....	45
Figure 25: 3-Phase 120 W 24 VDC SMPS.....	45
Figure 26: System Power PCB Layout	46
Figure 27: Load Banks with Connection Panels.....	47
Figure 28: Back Panel Binding Post Connections	48
Figure 29: Neutral Grounding Resistor.....	48
Figure 30: Small Fault Current Board	49
Figure 31: Available Fault Currents from Small Fault Current Board	50
Figure 32: High Current Test Set.....	51
Figure 33: Steady-State Feeder Energization Errors	55
Figure 34: Feeder 4 Phase-A Line Charging Current	56
Figure 35: Feeder 4 Phase-A Energizing Harmonic Content	56
Figure 36: Filter Transient Response to 180 Hz Feeder 3 Energization.....	57
Figure 37: CT Output Post-Energization	58
Figure 38: ATP-DRAW EMTP Schematic of Test System	62
Figure 39: Solidly Grounded System, 700 Ω Fault on Feeder 1.....	63
Figure 40: HRG System, 0 Ω Fault on Feeder 3.....	64
Figure 41: HRG System, 47 k Fault on Feeder 4.....	64
Figure 42: 3-Phase GFCI Prototype Hardware.....	66

Figure 43: Digital System Block Diagram.....	67
Figure 44: ADSP-B538F EZ-KIT Lite	68
Figure 45: Sensor PCB CAD Layout with Comments	70
Figure 46: Ideal 3-Ph GFCI Maximum Response Time.....	73
Figure 47: High Level Software Flow	74
Figure 48: PSOC5 Steady-State Filter Output, 60 Hz Input.....	79
Figure 49: Steady-State Filter Output, 120 Hz Input.....	79
Figure 50: Filter Transient Response, 60 Hz Input.....	80
Figure 51: 500x Sample Noise Floor of AD7606.....	81
Figure 52: Signal Chain Input and Output Data, 0.98mA Primary Current	82
Figure 53: Signal Chain Input and Output Data, 5.9mA Primary Current	83
Figure 54: Calibrated Signal Chain Percent Error	85
Figure 55: 5.9 mA Transient Response.....	86
Figure 56: 4A Transient Response.....	87
Figure 57: Secondary CT Voltage, 800 A Primary Current (No tripping)	89
Figure 58: Filter Output (top) and Input (bottom), 800 A Bus Fault.....	90
Figure 59: Filter Input, 800 A Bus Fault, Relay Tripping Enabled	91
Figure 60: Feeder 1, Phase A, 4 A Fault Interruption.....	94
Figure 61: Filter Output, Feeder 4 Energization Error Current Creating Incorrect Trip in Feeder 1 3.93 mA Fault	96
Figure 62: System Response to 3.93 mA Fault without Feeder 4 Energized	97
Figure 63: System Response to 5.9 mA Fault, Feeder 1, Phase A	97
Figure 64: Measured 5.9 mA Fault Current, Feeder 1, Phase A.....	98

Figure 65: Measured and Ideal Relay Response Times versus UL 943 Class A Requirements	99
Figure 66: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 4 A Fault (yellow), 0° Fault Inception Angle	105
Figure 67: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 4 A Fault (yellow), 90° Fault Inception Angle	106
Figure 68: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 293 mA Fault (yellow), 0° Fault Inception Angle	107
Figure 69: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 1 A Fault (yellow), 0° Fault Inception Angle	107
Figure 70: Filter Output (top) and Input (bottom), 4 A Feeder 3 Fault with Main Output Diminished due to Remnant Flux	109
Figure 71: Filter Output (top) and Input (bottom), 4 A Feeder 3 Fault with Feeder 3 Output Diminished due to Remnant Flux	110
Figure 72: EMTP Schematic Corresponding an Ungrounded System	124
Figure 73: Instantaneous Zero Sequence Current Sensed by CBCTs, 700Ω SLG fault on Feeder 1	125
Figure 74: Instantaneous Zero Sequence Current Sensed by CBCTs, 700Ω SLG fault on Main Bus.....	125
Figure 75: Simplified CT Schematic for Volt-Second Preservation Analysis [12].....	127
Figure 76: Operating Time for CT during Large Fault (f = 60 Hz).....	130
Figure 77: W-20 Core and Winding	131

Figure 78: Linearity of W-20 for 179, 427, and 978 Ω Burdens with FG Source, 4-30 mA Input.....	132
Figure 79: LTSpice CT Frequency Response (magnitude is solid) for 180 Ω (pink), 427 Ω (yellow), and 1000 Ω (blue) Burdens	134
Figure 80: CT Gain, FG vs. Precision Oscillator Source, 1-30 mA Input.....	135
Figure 81: CT Gain, Precision Oscillator, .001-1 A Input.....	136
Figure 82: W-20 Excitation Curve.....	138
Figure 83: Interpolated Excitation Curve for U3402A Data in Figure 30	139
Figure 84: RCF for a) 179 Ω Burden, 582 N-eff, b) 427 Ω Burden, 622 N-eff, c) 980 Ω Burden, 678 N-eff.....	140
Figure 85: W-20 B-H Curve for 3.4 A-rms Input.....	142
Figure 86: Chan Core Based LTSpice W-20 Model.....	143

CHAPTER 1

INTRODUCTION

The basics of electrical safety and modern household ground-fault circuit interrupter (GFCI) technology are introduced as a prelude to the 3-phase GFCI issues and design.

Electrocution and Electrical Safety

Electrical safety is paramount in today's society, where use of electrical power is commonplace. Insulation, for the most part, solves this problem. Nevertheless, people will inevitably come into contact with energized conductors, thereby allowing current to flow through their bodies to ground. This flow of current, known as "ground fault current", can pose a serious health threat.

The severity of a personnel ground fault depends on the path the current takes, the magnitude of the current, and the duration of current flow (larger fault currents can only be safely tolerated for a small amount of time, whereas smaller currents can be tolerated for larger amounts of time). Since it is impossible to predict the current path, the worst must be assumed: a fault current (of sufficiently high magnitude and duration) can flow through a person's heart, and ventricular fibrillation can occur; thus the heart stops rhythmic contraction. Without immediate medical assistance, death will follow within minutes.

With this type of scenario in mind, acceptable limits on magnitude and duration of fault current, once defined, can be imposed via a GFCI type device such that ventricular fibrillation, and all other milder effects are prevented. As outlined in the next few

sections, these limits have been previously defined by Underwriters Laboratory (UL). From an engineering standpoint, no other information is needed to design a GFCI, however, for reference, the mechanics of electrocution and a few salient points are included.

The effects of milliamp 60 Hz current flow through a person were first established by Charles Dalziel, who most importantly determined that 99.5% of the population is able to let go of a conductor sourcing 6 mA or less. Tests revealed that on average, perception of current sets in at around 1.1 mA for men, and 0.6 mA for women, pain is first experienced at 9 mA and 6 mA (men and women), and inability to let go at 16 mA and 10.5 mA [1]. Based on this data, current UL standards require "shock prevention", a guarantee that the smallest current which results in the GFCI taking action is "not below 4 mA, and not above 6 mA". In other words, the tripping threshold is set at 5 mA and the system must have milliamp resolution. Shock prevention implies very little possibility of a person becoming 'stuck' on a line, unable to let go. Elsewhere, in places like Europe, the tripping threshold is concerned only with electrocution, and so the threshold is relaxed to 30 mA and higher [2]. This is acceptable, as prolonged current flow at lower levels decreases the body impedance, which in turn increases the fault past the 30 mA threshold. Kouwenhoven and Milnor state that currents 50 mA and less "usually have no serious results, though they may be very painful and may result in temporary loss of consciousness" [3].

As current levels increase, the possibility of a serious health risk, ventricular fibrillation, increases. "[Ventricular fibrillation] is probably the most common cause of death in electric shock cases, and can be produced by moderately small currents that

cause chaotic, uncontrolled, rapid contraction of the heart muscle. The muscle writhes, something like a bag of worms, though faster" [4]. If a minimum internal body resistance of $200\ \Omega$ is considered [3], the maximum current a 480 V system can source is held under 1.4 A. According to Kouwenhoven and Milnor, the range of currents between 100 mA and 2 or 3 amperes result in ventricular fibrillation from which the heart rarely spontaneously recovers. Indeed, this range covers the expected fault currents on a 480 V system; at a touch voltage of 400 V, 5% of the population has a $700\ \Omega$ body resistance, or an anticipated fault current of about 400 mA [5]. Since only about $1/7^{\text{th}}$ of the heart cycle is vulnerable to current induced cardiac arrest [2], there is somewhat of an allowable time during which a larger current can be tolerated, and hence the trade-off between allowable fault magnitude and duration. [5] has quantitative information regarding this.

Further research into the medical and biological effects of electric shock should begin with a review of Dalziel's original work. The 3-phase GFCI is designed simply with the goal of meeting the UL mandated requirements.

Current GFCI Technology

The prevalent GFCI typically found in the kitchen or bathroom is termed a Class A GFCI by UL, where the class namely indicates operation at 4-6 mA for systems 150 V and less, within a specific timeframe. A Class A GFCI consists of a toroidal core-balanced current transformer (CBCT), analog amplifier and discriminator and circuit breaker (CB), which are all commonly integrated into a single package and located in an 120-240 V outlet housing. All wires (except for the ground return, if present), pass through the window of the CBCT. If the net, or zero-sequence current (they are proportional), in all of the wires exceeds the 4-6 mA threshold, a ground fault is assumed,

and the resulting signal on the secondary of the CT is used to trip a CB after it reaches the 5 mA threshold. Tripping is thereafter immediate, and a GFCI is not required to coordinate with any other protection.

The means by which a tripping decision is made vary. One approach is to input the voltage developed across the CT secondary to a window comparator with a time delay (see RV4141A), while another is to use secondary current to charge a capacitor which is otherwise continually discharged; sufficient voltage on the capacitor indicates charge accumulation due to a ground fault (see LM1851). Both of these methods are simple and marketed as integrated circuits for GFCI developers; they are completely analog solutions.

UL 943 Class A Tripping Curve

In order to satisfy the personnel safety requirements described above, UL standard 943 defines the maximum allowable response time of a Class A GFCI via equation (1.1), where t is in seconds and i is in milliamps.

$$t = (20/i)^{1.43} \quad (1.1)$$

Shown below in Figure 1 is (1.1) plotted on a log-log scale.

Figure 1: UL943 Class A Tripping Curve

As expected, faster tripping times are required for larger fault currents. As long as fault current duration is limited by this curve, a GFCI ensures shock and electrocution prevention. Points of interest in Figure 1 are:

- At the 5 mA threshold, over 9 seconds is allowed to trip, whereas at 264 mA, only 25 ms is allowed to trip.
- Above 264 mA, only 25 ms is allowed to trip, but just up to fault currents of 609 mA.
- Beyond 609 mA, there is no concern for the allowable time to trip; due to the known system voltage and minimum human body impedance, fault currents greater than 609 mA (this is for 120 V; compare to the 1.4 A estimated on 480 V systems) simply cannot be personnel faults. As such, tripping time is irrelevant.

In keeping with the curve described by (1.1) and shown in Figure 1, the 3-phase GFCI is designed with the ability to meet the Class A tripping curve. Due to an increased phase-ground voltage of 277 V and a constant range in body impedances, larger personnel fault currents occur at 480 V, and so the 3-phase GFCI, unlike a Class A 150 V GFCI, is responsible for detecting and tripping fault currents beyond the current 609 mA upper limit. In fact, the prototype is designed to trip within 25 ms for *all* fault currents greater than 264 mA.

As a last note, for 480V systems, UL currently has a tentative outline titled UL 943C, which requires tripping at 6-20 mA (electrocution prevention only; no shock prevention), with a maximum time-to-trip of 20 ms. This prototype is designed with a quickest response of 25 ms.

CHAPTER 2

3-PHASE GFCI DESIGN ISSUES

Concepts needed to apply GFCI technology to 3-phase 480 V systems, and the tripping scheme to be implemented in the prototype and previously patented in US Patent #7301739 are summarized.

Line Charging Currents

The most significant difference for GFCI design between a 120 V Class A GFCI environment and that of a 3-phase 480 V system is the amount of line charging current present. Line charging current is the amount of current flowing to ground, most commonly due to the distributed phase-ground capacitance of the cables, or the lumped capacitance to ground of any surge capacitors. In general, it is the current flowing due to the total capacitive coupling to ground on the electrical network. System, or total line charging current can be determined in an ungrounded system by solidly faulting a phase to ground, and measuring the resultant fault current. In a high resistance grounded (HRG) or solidly grounded system, the measured fault current would be the vector addition of the line charging current and the fault current flowing through the transformer neutral connection.

The amount of line charging current a system has will vary depending on the voltage of the system, the kVA of the system, and the extent of the feeder lengths. At the cable level, the distributed capacitance scales linearly with line length (surface area) and system voltage, and logarithmically with proximity to nearby ground conductors. Line

charging capacitance and methods for determining these parameters theoretically and experimentally can be found in [6]. The cable used in many residences has minimal capacitance, around 20 nF per 1000 feet; at 120V, this translates to about 1 mA per thousand feet. Unshielded cable capacitance ranges from 20 to 60 nF per 1000 feet, depending on the cable size and construction type [7]. As such, in a household environment, line charging currents will typically remain under a milliamp, and are therefore negligible for the Class A GFCI tripping requirements described in Chapter 1.

On the other hand, in 3-phase 480 V systems, line charging currents can range up to 5 A. Systems are much larger both in physical size and load capacity. Summarized below in Figure 2 and Figure 3 are general guidelines for estimating line charging current contributions at the 480 V level [6].

Figure 2: Cable Line Charging Currents [6]

Figure 2 shows expected line charging current magnitudes for various ranges of system voltages as a function of conductor size, length, and dielectric constant of the insulating material. The currents are proportional to the applied voltage and the total surface area of the conductor (the mils in the figure indicate insulation thickness). The values given in Figure 2 estimate the ground fault current contributed by a segment of cable due to a bolted phase fault to ground. The shown value is actually the sum of the charging current in all three phases.

<u>Voltage</u>	<u>Motors, ma/1000hp</u>		<u>Generators, ma/MVA</u>	
	<u>Min.</u>	<u>Max.</u>	<u>Min.</u>	<u>Max.</u>
0.48 kV	5	10	--	--
2.4 kV	20	40	12	16
4.16 kV	35	70	18	25
6.9 kV	60	115	25	35
13.8 kV	115	230	36	50

<u>Voltage</u>	<u>C, ufd.</u>	<u>$3I_{co}$, ma.</u>
0.48 kV	1.0	313
2.4 kV	0.5	784
4.16 kV	0.5	1357
6.9 kV	0.5	2253
13.8 kV	0.25	2253

Figure 3: Charging Currents for Motors, Generators, and Surge Capacitors [6]

As shown in Figure 3, surge capacitors used for equipment protection contribute a considerable amount of line charging current, whereas motors and generators themselves are relatively small. Since motors are often protected with surge capacitors to prevent transient winding insulation failures, systems with primarily mechanical loading may have higher line charging currents than expected.

Though the previous two figures give a means to estimate individual contributions to the total line charging current, for design purposes, industry experience and field

measurements have shown that at 480 V there is usually less than 1 A of total line charging current, with a maximum of about 5 A [8]. In the test system used for this thesis, the system line charging current is only about 364 mA. For purposes of the prototype, 5 A is considered a maximum.

Effect of Line Charging Currents on CBCT Measurements

In 120 V residential applications, line charging currents are not a concern, however at 480 V it becomes an issue for GFCI sensitivity and resolution. The approach taken for Class A GFCI design is not valid in general, primarily because zero-sequence currents arise whenever an imbalance in system ground current exists. Ground fault current is merely one instance of imbalanced system ground current, and a CBCT itself cannot discriminate between a fault current and imbalance in the current flowing through the phase-to-ground capacitances of the line.

When a CBCT monitors only a single phase, the line charging current of that phase always appears as a source of zero-sequence current. In a three-phase 480 V system, the line charging current in each phase can be well over the 5 mA tripping threshold. However, assuming the capacitance to ground on each phase is equal (using bundled 3-phase cables), and assuming the phase voltages are balanced, line charging currents will sum to a very small net current, necessarily below the 5 mA threshold, if not a milliamp or so. In an interview with the author of [9], field tests at the Chevron Richmond refinery indicated this to be true.

Regardless, phase voltages cannot be guaranteed to remain perfectly balanced at all times. During a single line to ground (SLG) fault (somewhere downstream of the main bus), a depressed phase voltage is observed at the bus due to voltage division (on

the faulted phase) amongst the source, line, and fault impedances. If the faulted phase's voltage at the bus is depressed severely enough, imbalance in line charging currents on all feeders can exceed 5 mA. This is verified with computer simulations in the following chapter.

Since a SLG bolted fault at the end of a feeder can create currents in the non-faulted feeders greater than 5 mA, the conventional GFCI design approach is inadequate; independent tripping at 5 mA results in non-faulted feeders' breakers opening, and relay selectivity is lost. When the imbalance in line charging currents on non-faulted feeders exceeds 5 mA, the net current is termed a "nuisance" current. In many electrical systems, "nuisance tripping" is unacceptable, and has so far prevented the implementation of GFCI technology.

Hamer's Solution: The 3-Phase GFCI Digital Relay

This thesis is based on the work of Hamer in [9], in which the tripping logic described in US Patent #7301739 [10] is detailed as a solution to implementing a 3-phase 480 V GFCI. In summary, rather than each feeder acting independently on its own CBCT measurements, a digital relay is used to monitor and control tripping on all feeders and the main bus. Thus, tripping decisions are made with information gathered from each feeder and the main bus. With this broader picture, selectivity can be maintained.

As validated in the following chapter, the tripping logic is surprisingly simple in concept. Since a severely large fault must occur for a noticeable voltage depression at the bus, the faulted current is always much larger than any nuisance currents it is responsible for generating. Therefore, the faulted feeder is always the largest, so it should be tripped, and all other feeders - even if over the 5 mA threshold - should remain energized. Since

simultaneous ground faults on different portions of the system are highly unlikely, it is acceptable to withhold tripping of feeders with smaller fault currents. If they persist after the largest feeder is cleared, they are then cleared.

With this in mind, a prototype was constructed so as to implement and validate the tripping logic. The prototype included all necessary hardware and software, as detailed in Chapters 4 and 6. A 480 V test system was constructed as a platform on which to test the relay, as detailed in Chapter 5. Following are some final points to be made regarding power system analysis and the 3-phase GFCI.

CBCT Measurements

Load currents are not measured by a CBCT, so long as the loads are not locally grounded. Line currents into a Δ or 3-wire Y connected load always sum to zero, regardless of whether or not the load is balanced. If the load is 4-wire Y connected, the neutral must be returned through the CBCT before connecting to the system neutral,. Where the connection to ground is made, the residual current traveling in the opposite direction of the line current acts to balance the three line currents. CBCT load current transparency holds for unbalanced system voltages.

Zero-sequence current is the vector sum of two quadrature components: the ground current flowing due to the system coupling to ground at the transformer neutral (assumed resistive), and the capacitive line charging current. This is described by equation (2.1).

$$\overline{I_0} = \overline{I_{R0}} + \overline{I_{C0}} \quad (2.1)$$

Depending on the location of the CBCT, measured current can be defined by (2.2), where a lack of over-bar indicates magnitude.

- Main

$$I_M = I_{R0} = \text{Resistive Fault Contribution} \quad (2.2.a)$$

The main CBCT will sense only fault current contributions that flow through the system grounding at the transformer neutral, since it is assumed there are negligible line charging current contributions upstream of the main bus.

- Non-Faulted Feeder

$$I_i < 3I_{C0} = \text{System Line Charging Current} \quad (2.2.b)$$

A non-faulted feeder will sense only the imbalance in its line charging currents. This will always be less than the total line charging current.

- Faulted Feeder

$$I_j = \sqrt{I_M^2 + \left(\sum_{i, i \neq j} I_i \right)^2} = \left| \overline{I_F} - \overline{I_{C, Feeder j}} \right| \quad (2.2.c)$$

The faulted feeder will sense the vector sum of the resistive component of fault current (seen at the main), and the sum of the non-faulted feeders' line charging current contributions. It cannot sense the fault current due to its own feeder alone.

Since a faulted feeder can only measure the fault current contributed from the neutral grounding and non-faulted feeders, there is a special case in which the CBCT cannot detect a fault properly. On an ungrounded system, if the fault occurs on a feeder which is responsible for a majority of the line charging current, small faults cannot be detected, since the fault current essentially circulates downstream of the CBCT. This instance is highlighted in green in Appendix A, although it is not of much concern since it only occurs on an ungrounded system, for which this prototype is not concerned (explained in Chapter 3).

Effects of System Grounding

A system is most at risk to experience nuisance tripping when it is grounded through some impedance, or not at all. When a system's neutral is grounded through an impedance and the source transformer is connected in delta on the high side of the transformer, the only source of ground fault current (other than the line charging current) is through the neutral impedance. In simulations, the voltage in the secondary windings are assumed constant, and do not change during a fault (e.g., the system is connected to an infinite bus). As such, for a SLG fault on the main bus, the only impedance limiting the fault current is that of the transformer:

$$I_{Fault} = \frac{V_{l-g}}{\sqrt{R_{txfr}^2 + X_{txfr}^2}} = \frac{S}{\sqrt{3}V_{l-l}\sqrt{(\%R)^2 + (\%X)^2}} \quad (2.3)$$

Naturally, fault current is much further reduced by adding a neutral grounding impedance to the transformer; however this serves to separate the system neutral from ground potential during a ground fault. Consequently, the sum of the phase voltages shifts with respect to ground, and line charging currents become imbalanced; thus the exaggerated nuisance currents on HRG or ungrounded systems.

Consider a HRG system. Phase voltages with respect to ground are defined via (2.4) as

$$\overline{V_{AG}} = \overline{V_{AN}} - \overline{V_{GN}} \quad (2.4.a)$$

$$\overline{V_{BG}} = \overline{V_{BN}} - \overline{V_{GN}} \quad (2.4.b)$$

$$\overline{V_{CG}} = \overline{V_{CN}} - \overline{V_{GN}} \quad (2.4.c)$$

and the source is constant, such that

$$\overline{V_{AN}} = V_p \angle 0^\circ = constant \quad (2.5.a)$$

$$\overline{V_{BN}} = Vp\angle(-120)^\circ = \text{constant} \quad (2.5.b)$$

$$\overline{V_{CN}} = Vp\angle(+120)^\circ = \text{constant} \quad (2.5.c)$$

In the simplest case, a bolted fault on main bus forces phase A to ground potential.

$$\overline{V_{AG}} = 0 \quad (2.6.a)$$

Combining (2.4) through (2.6) for phase B and C voltages to ground yields

$$\begin{aligned} \overline{V_{BG}} &= \overline{V_{BN}} - (\overline{V_{AN}} - \overline{V_{AG}}) = \overline{V_{BN}} - \overline{V_{AN}} \\ &= Vp\angle(-120)^\circ - Vp\angle 0^\circ = \sqrt{3}Vp\angle(-150)^\circ \end{aligned} \quad (2.6.b)$$

$$\begin{aligned} \overline{V_{CG}} &= \overline{V_{CN}} - (\overline{V_{AN}} - \overline{V_{AG}}) = \overline{V_{CN}} - \overline{V_{AN}} \\ &= Vp\angle(+120)^\circ - Vp\angle 0^\circ = \sqrt{3}Vp\angle(+150)^\circ \end{aligned} \quad (2.6.c)$$

Using the results of (2.6), the current sensed in a non-faulted CBCT is the sum of the line charging currents, which is proportional to the sum of the voltages

$$\begin{aligned} \sum \overline{I_{ch}} &= j\omega C \sum \overline{V_G} \propto (0 + \sqrt{3}Vp\angle(-150)^\circ + \sqrt{3}Vp\angle(+150)^\circ) \\ &= -3Vp \end{aligned} \quad (2.7)$$

On the other hand, for a solidly grounded system, the neutral is fixed at ground potential, and the largest current imbalance that can occur on a non-faulted feeder due to a solid bus fault is proportional to

$$\begin{aligned} \sum \overline{I_{ch}} &= j\omega C \sum \overline{V_G} \propto (0 + Vp\angle(-120)^\circ + Vp\angle(+120)^\circ) \\ &= -Vp \end{aligned} \quad (2.8)$$

A difference of a factor of three between (2.7) and (2.8) can be observed in the results of Appendix A. This difference partially indicates why nuisance tripping currents can occur more easily on a HRG or ungrounded system, but it must be noted that a bolted bus fault was assumed. The more complete reason as to why there is a large susceptibility in HRG or ungrounded systems to nuisance tripping currents is that smaller

fault currents can create more imbalance by separating the neutral from ground potential, whereas on a solidly grounded system this does not happen and imbalance is only created by forcing the faulted phase to decrease at the bus (which requires a very large fault current). This is why comparing nuisance tripping between different groundings on the basis of fault current magnitude results in such large disparity. Refer to Appendix A for confirmation that nuisance tripping is more likely on a HRG or ungrounded system.

In terms of practicality, a solidly grounded system is nevertheless just as susceptible to nuisance tripping, as bolted SLG faults are most common.

CHAPTER 3

EMTP COMPUTER SIMULATIONS

ATP-EMTP (Alternative Transients Program-ElectroMagnetic Transients Program), hereon referred to as EMTP, was used to first verify the V-HARM results presented in [9] which highlight the nuisance tripping currents, and later on to validate test system measurements with simulations. This chapter details the successful verification of the solidly grounded and high-resistance grounded (HRG) systems with V-HARM results. An error in the ungrounded system simulations and tripping logic is exposed, and an alternative tripping scheme is proposed.

As a way to verify the problem of nuisance tripping, V-HARM results were validated using an identical system within EMTP. The electrical schematic as produced using ATP-DRAW for the V-HARM verification studies is shown below in Figure 4.

Figure 4: Electrical Schematic for EMTP vs. V-HARM Comparison

Figure 4 shows a simple 480V radial network with four feeders, each of which have sufficient line charging currents to generate nuisance tripping currents. TACS (Transient Analysis of Control Systems), a sub-program of EMTP, was used to measure the steady-state root-mean-square (RMS) of the sum of phase currents in the feeder and main branches, emulating zero-sequence CBCT current measurements. For more details, refer to Chapter 5 or [9], which both detail the parameters used. In short, solidly grounded, HRG, and ungrounded systems were all simulated for solid, 700 Ω , and 47 k Ω faults on the bus and each feeder. Loading was varied, and lines were simulated using 3-

phase nominal- π sequence networks, based on 3-phase plus neutral 2/0 copper conductors in steel conduit with lengths of 500 feet, 1000 feet, 2000 feet, and 500 feet with 1 μ F surge capacitors.

Tabulated results comparing EMTP results with V-HARM appear in Appendix A; they are in the same format as that presented in [9]. Other than the exceptions noted below, results between EMTP and V-HARM agree within 5%, or 1mA, whichever is larger. The nuisance tripping currents are confirmed, and all would-be nuisance tripping currents are highlighted in yellow in Appendix A. HRG and ungrounded systems are more likely to develop a nuisance tripping current. All instances where fault current is not detected properly such that it affects relay operation are highlighted in green, however as this only occurs on an ungrounded system, it is of no concern; the prototype is only designed for solidly grounded and HRG systems.

The main discrepancy between EMTP and V-HARM simulations appears in the ungrounded system. The only source of zero sequence or fault current during a single-line-to-ground (SLG) fault on an ungrounded system is from the distributed phase capacitance to ground. Since the transformer is ungrounded and the primary winding is delta, there will be no zero-sequence current contributed from the transformer. Therefore, during a fault the main CBCT will read the imbalanced line charging current due to the section of line running from the bus to the transformer. However, no such line is modeled (in both simulations) and so there should be no current measured at the main. EMTP results confirm this, however V-HARM shows a non-zero value - the fault current magnitude itself. Either bus faults were simulated in V-HARM as upstream of the main CBCT, or the main CBCT was placed in the fault path to ground. EMTP results were

proved in lab to be correct. With this in mind, for bus faults on the ungrounded system, the tripping scheme described in [9] is invalid on ungrounded systems, and so is not further pursued in the prototype. An alternative tripping logic approach utilizing the phase of the extracted 60 Hz component of the measured current's phasor, rather than the magnitude, is detailed in Appendix C, but is not further pursued in this study.

Some V-HARM results indicate 1 mA sensed on a non-faulted feeder during a 46k Ω fault, however 0 mA during a 700 Ω fault; this does not make sense and remains unexplainable. It has no impact on any tripping scheme either way.

CHAPTER 4

PROPOSED SENSING SOLUTION

Sensing for the 3-phase GFCI requires both mA resolution in order to meet the 4-6 mA threshold window, and the ability to measure currents up to the SCCR (short circuit current rating) of a solidly grounded system - roughly 10 kA. This approximates 7 orders of magnitude, or 140 dB of dynamic range. Since fault current values are given in RMS and the sampled values are actually points on a sine wave, the required dynamic range is even further increased. After closer inspection of the necessary information for the tripping logic, these requirements are simplified, and a sensing solution is developed.

Simplification of Sensing Requirements

Since the highest possible measured current on any non-faulted feeder is some fraction of the system line charging current, only currents below the system line charging current need to be measured with milliamp resolution for the system to have proper discrimination between the faulted and non-faulted feeders. In other words, if the measured current on a feeder ever exceeds the system line charging current, it will always be the faulted feeder, and no other information is needed to maintain selectivity. In this case, both the main and the faulted feeder would measure a current larger than the system line charging current; the main would be slightly larger, but there is no need to differentiate between the two, as either way the feeder is tripped. If the main is ever greater than the system line charging current, and none of the feeder's measurements are, then the bus is faulted.

This relaxes the required sensor current measurement range. The sensing solution needs only to measure with milliamp accuracy up to the system line charging current, beyond which the current level can effectively be condensed to a single-bit of information - whether it is above or below the system line charging current. Any feeder in a multi-feeder system has only a fraction of the total system line charging current, and so should never approach the threshold; however it is best to consider the worst case scenario for systems with few feeders of lop-sided lengths. As discussed previously, the maximum line charging current expected on a 480 V system is 5 A. Thus, mA resolution is only required up to 5 A.

CT Solution

Traditionally, iron-core current transformers have been used in Class A GFCI applications. Since a Class A GFCI trips once the output passes 6 mA, there is no need to differentiate between fault current levels which are over 6 mA. The core of Class A GFCI has a linear range that is large enough such that it can enter saturation with no ill effect. The output available while the major B-H loop is traversed back and forth between saturation is sufficient enough to cause a trip. More so, remnant flux is not an issue, as the minor hysteresis loop available in a maximum remnant flux state with the same sense fault inception angle (worst-case) is also enough to cause a trip to occur [11]. Although the Class A acceptable transient response with remnant flux will not be entirely acceptable in the 3-phase GFCI (which needs to differentiate fault current levels), they were chosen as a suitable sensor for the 3-phase GFCI since CTs are commercially available from Bender with a linear range capable of 5 A with mA resolution. The transient response issue due to remnant flux is detailed in Chapter 7.

If the maximum line charging current in a majority of systems is 5 A, then the desired sensing capability of the system is 0 to 5 A with mA resolution, and simple detection of the presence of fault currents greater than 5 A (with 1 bit resolution). The W-20 CT from Bender can fulfill both of these requirements, and has thermal ratings of 6 kA for 40 ms, 2.4 kA for 1 s, and 40 A continuously.

For the 0-5 A range, a burden resistance is chosen such that the CT saturates near 5 A, in order to increase the secondary output and improve the signal-to-noise ratio (SNR). The voltage developed across the burden is input to an analog-to-digital converter (ADC), and processed through a digital filter before being used in the tripping logic; all standard signal processing. The W-20 core is shown in the next section to be linear enough over this range for this approach to be valid.

For fault currents greater than the rated primary current (which is forced to be near 5 A), essentially the same approach can be taken, but it is not immediately evident why this is acceptable. When a CT exceeds its rated primary current, it begins to saturate. As described in [12], for primary currents beyond the rated level, the CT preserves only the product of secondary volts and seconds for which it is present, or the volt-second area. Once this area surpasses the area corresponding to that due to the primary rated current, secondary output goes to zero. If we define the minimum volt-time area that is produced by (4.1.b) as that produced by the rated primary current,

$$V_{S,peak-em.sat.} = \frac{I_{P,peak-rated}}{N} \cdot Z_B = I_{S,peak-rated} \cdot Z_B \quad (4.1.a)$$

$$A_{min} = \int_0^{T/2} V_{S,peak-em.sat.} \sin(\omega t) dt \quad (4.1.b)$$

then for any fault current which is M times larger than the rated primary current, it can be shown (see Appendix C) that it is possible to calculate via (4.2) the length of time that the

output is present on the secondary of the CT and larger than a given threshold voltage. The use of a threshold voltage is to exclude time elapsed over the range in which the CT remains linear, but can be set to 0 to nullify its effect.

$$t_{op} = \frac{1}{\omega} \left\{ \cos^{-1} \left(\frac{M-2}{M} \right) - \sin^{-1} \left(\frac{V_{th}}{M \cdot V_{S,peak-em.sat.}} \right) \right\} \quad (4.2)$$

The volt-time area preservation is shown qualitatively in Figure 5, where for larger fault currents, secondary output duration becomes smaller and peak voltage increases.

Figure 5: CT Preservation of Volt-seconds [12]

The process of CT saturation is best explained in terms of the necessary voltage required by the core. A CT ideally translates current from the primary to the secondary with a constant turns ratio. This is accomplished via normal transformer action, in that the time derivative of flux through the core creates a secondary voltage, which when connected across a passive burden resistance, gives the proportionally correct secondary current flow. When the core saturates, the time derivative of the flux density diminishes and there is no induced secondary voltage produced to generate the secondary current flow. In other words, the secondary voltage is an indicator of whether or not the core is saturated. Any given secondary voltage can only be sustained for so long, and so if

assuming a 60 Hz sine-wave, the duration can be calculated (Appendix C). This is the general explanation for CTs with passive burdens.

When a 5 A primary rated CT measures a 5 kA fault, the output faithfully traces out the 5 kA sine-wave, but only for a *very* short duration of time, and then goes to zero. The amount of time as calculated in (4.2) can be graphed for any arbitrary fault current to CT rated current ratio greater than two.

Figure 6: Operating Time for CT with Passive Burdens During Large Fault

Figure 6 shows the expected time a secondary voltage is expected to be present once it passes a given threshold voltage; the exact threshold becomes irrelevant at larger currents. As an example, for a 5 A rated CT with a 5 kA fault, $M = 1000$, and the expected output is greater than the threshold for just under 200 μs . The derivation of this approach is included in Appendix C.

CTs are self-protecting to some extent, since once saturated the secondary voltage transforms into an impulse, and the energy of the fault is no longer proportionally

transferred to the burden. Disregarding non-magnetic coupling, the maximum transferrable energy is defined at the point of saturation, where $dB/dH = \mu_0$; i.e., once the core saturates, very little further energy is transferred. Despite this soft limit on energy transfer, the large magnitude of the saturation induced voltage spikes can damage semiconductor devices. For protection, the W-20 includes bi-directional $\pm 6.8\text{V}$ 600-1500W transient voltage suppressing (TVS) diodes (both a P6KE 600W and 1.5KE6.8 1500W diode were found when dissecting two W-20s).

Figure 7: TVS Diodes Clamping Saturated CT Output

Bipolar diode clamping of secondary CT voltage is shown in Figure 7, where the secondary waveform is clipped at $\pm 6.8\text{V}$.

The TVS diodes inadvertently provide more than just over-voltage protection. By clamping the voltage, they are acting as an active impedance; open when the voltage is below 6.8V , and nearly a short when the voltage is above 6.8V . What is important is that preventing the secondary voltage to increase much past 6.8V actually precludes the core from entering saturation. It is simplest to understand this in terms of volt-second area preservation. Since the secondary voltage is limited, it must be held at this value for a longer time than it would otherwise be before the volt-second area is maximized, and secondary output goes to zero. What this means is that secondary voltages actually exist for much longer than expected if referencing the Appendix C or Figure 6.

Figure 8: Volt-second Area Preservation with TVS Diode Clamping

Shown in Figure 8 are secondary CT outputs with and without over-voltage protection when the primary current is 3 times the rated current. Channel 1 has TVS diodes and Channel 2 does not. By volt-second preservation, the areas in which the two curves do not overlap should be equal. Approximating the excess area of Channel 2 shown by half the area constrained by the on-screen cursors yields 4.8mV-s, whereas the excess rectangular area of Channel 1 was calculated at 5.1mVs. To a rough approximation, the concept is validated, and in fact holds true when much larger currents are present- see the high current test results of Chapter 7.

During a very large ground fault, clamped CT output will appear as successive positive and negative rectangular pulses, due to the high slew rate of the secondary voltages traces prior to clamping. The rectangular pulses have large enough duration to be sampled by the ADC, and the extracted 60 Hz component of the sampled bipolar pulse train is large enough to always be greater than any line charging currents used in the test

setup. In practice, it may be best to implement a bipolar peak detector as suggested in [13], rather than simply allow it to pass through the filter.

Effectively, the presence of very large currents can be detected by the same circuit used for the 0-5 A range, especially if line charging currents do not approach the 5 A that the filter is capable of measuring. Otherwise, any number of tricks in the processor can be used to differentiate the faulted feeder from the unfaulted feeder, as the extracted 60 Hz signal from a rectangular pulse may be on the same order of magnitude as a non-faulted feeder in some obscure cases. At mildly saturating current levels, CT output will begin to appear as rectangular pulses, and remain as such for all greater currents; output never significantly diminishes. The limiting factor for systems with high fault current availability is the short-time thermal ratings of a CT sensitive to a milliamp level, rather than ability to sense CT output at very large currents.

Digital Filter

Once values are sampled in the ADC, a digital filter is used to notch out power system harmonics, remove or lower DC offset, and extract only the fundamental component of the secondary CT voltage. Coupled with the proper analog signal conditioning in the analog front end (AFE), a digital filter allows less restraint on the corner frequency of the anti-aliasing filter, so the bandwidth can be opened up, improving the transient response of the signal chain.

Full-Cycle Cosine Filter

The following full-cycle cosine filter used in the prototype is motivated from [14], where the filter implementation is explicitly described. The choice of the filter, the filter length, and the sampling rate is justified in, and expounded upon, in [15]. A similar approach is taken in [16], where the filter is termed a DFT algorithm, and the specific application to nuisance tripping in 3-phase systems is discussed.

The filter coefficients

$$CFC_n = \cos\left(\frac{2\pi}{16}n\right) \quad (4.4)$$

The cosine filter

$$IX_{smpl+spc} = \frac{2}{N+1} \sum_{n=0}^N I_{smpl+spc-n} CFC_n \quad (4.5)$$

The phasor magnitude

$$|I_o| = \sqrt{(IX_{smpl+spc})^2 + \left(IX_{smpl+spc-\frac{spc}{4}}\right)^2} \quad (4.6)$$

Where

$$N = 15$$

$$n = 0,1,2 \dots N$$

$$smpl = \text{sequence of samples } 0,1,2,3 \dots$$

$$spc = \text{number of samples per cycle} = 16$$

$$I_{smpl+spc-n} = \text{current sample}$$

$$IX_{smpl+spc} = \text{filter output}$$

The output of the filter (the accumulator variable) is the extracted 60 Hz component of the input. With all harmonics removed, it is a phasor. Computing the phasor magnitude of the accumulator variable via the in-phase and quadrature component truncates the measured current into a single variable with the consequence of adding an additional quarter cycle of sampling to fill the filter window with post-fault data. This lengthens filter settling time to ~21 ms, and imposes a soft limit on the minimum possible trip time (this is adjusted with a variable length filter delay).

Simplifying the above and fitting to a finite impulse response (FIR) difference equation gives the following:

$$a_1 y(n) = b_0 x(n) + b_1 x(n-1) + \dots + b_M x(n-M) \quad (4.3)$$

With coefficients

$$b_n = CFC_n$$

$$a_n = \frac{N+1}{2}$$

Where

$$y(n) = \text{filter output}$$

$$x(n) = \text{newest input}$$

This difference equation, (4.3), is implemented in MATLAB, and analyzed at a 960 Hz sampling rate.

Figure 9: Analog Frequency Response (Magnitude) of Filter

Figure 9 shows desired unity gain at the 60 Hz fundamental, and notching of all integer harmonics. Notice the filter output is unity also at ~73 Hz, and greater than unity for frequencies between 60 Hz and 73 Hz. This is part of the filter design. Captured phasor magnitude filter output for a 66 Hz input is shown below in Figure 10.

Figure 10: PSOC Hardware Data- 66 Hz Filter Input and Output

Figure 10 shows oscillating, greater than peak-input, filter output for a 66 Hz input. While this is not desired, spectral noise from 60-73 Hz, as well as for the entire spectrum (minus the harmonics), will be quite low, so this susceptibility is an acceptable filter design compromise. See the hardware implementation of the filter in Chapter 6 for broadband signal-to-noise ratio (SNR) and noise characterization.

Figure 11: Analog Frequency Response (dB) of Filter

Figure 11 duplicates Figure 9 with dB scaling, emphasizing harmonic notching of all integer harmonics.

Since all digital filters are dependent on the correct sampling rate, the processor needs to minimize errors in sampling periodicity. The interrupts which generate the sampling are expected to be stable to 960 ± 5 Hz. To examine errors due to finite tolerances in the sampling rate, the sampling rate of the filter in MATLAB was varied ± 5 Hz (sampling at 965, 955 Hz). These sampling rate errors affected the 60 Hz component less than 1%, and harmonics were held to less than 2% of the 60 Hz component. A tolerance of 5 Hz at a 960 Hz sampling rate is equivalently a change of the filter window by $87 \mu\text{s}$, or .5%. Results indicate small variances in sampling rate can be neglected.

Half-Cycle Cosine Filter

The full-cycle cosine filter can be shortened to a half-cycle cosine filter, reducing settling time from 1.25 to .75 cycles, however this results in notching of only odd harmonics, over-shoot, and inability to completely remove DC offset.

Figure 12: Half-Cycle Cosine Filter Steady-State Response

Figure 12 confirms half-cycle filter rejection of only odd harmonics, and the inability to completely reject DC. Inability to remove DC may hurt the resolution of the signal chain, depending on the size of the input amplifier voltage offset error.

The transient overreach of the half-cycle filter is investigated in a C environment.

Figure 13: Half-Cycle Cosine Filter Transient Response; No DC Offset

Figure 13 shows overshoot in the transient response to a unit-less 200 count 60 Hz input.

Figure 14: Half-Cycle Cosine Filter Transient Response; DC offset = 1/2 Peak 60Hz Input

With a DC offset present equal to half the magnitude of the 60 Hz input component, the output does not remain constant. The advantage of the half-cycle filter is the lessened time to filter settling, however even with half a cycle less of delay it would

be impossible for the relay to meet the UL 943 Class A 25 ms tripping time requirement for larger faults (although this is mostly due to contactor opening times) without a variable length filter delay. Thus, a variable length filter delay is used, and the full cycle filter is elected, as it provides a better response when allowed to fully settle.

In summary, the sensor solution includes a single CT per channel, followed by a single pole 360 Hz anti-aliasing filter buffering the ADC. The ADC communicates sampled values to the processor, where the digital filtering occurs.

Figure 15: Signal Chain Hardware Block Diagram

Only a single channel is shown in Figure 15, although all channels are functionally the same. In reality there is only a single processor and two AD7606s.

CHAPTER 5

DETAILS OF TEST SYSTEM

Existing 480 V systems are expensive and installed for a particular purpose; namely something other than digital relay development. As such, a 480 V test system was constructed to provide a means for 3-phase GFCI development and verification.

Design and Construction

The 480 V system shown below in Figure 16, based on the original V-HARM studies presented in [9], was used as a template for the test system.

FIG. 1a

Figure 16: 480V Test System One-Line Diagram [10]

With the electrical system as shown in Figure 16, the test system was constructed.

Figure 17: Overhead view of 480 V Test System

The basic components of the test system are highlighted in Figure 17. The feeders are modeled, and there is contactor control to turn the feeders on and off. 10 AWG is used everywhere except for the 16 AWG from the feeders to the load banks. Wires were color coded as follows: 480 V phases are brown, orange, and yellow; 480 V contactor controls are dark blue; neutral is white; ground is black. The grey wire is 2 conductor 16 AWG plus a shield, used for CT inputs and 12/24V power distribution.

Wiring

A more detailed wiring schematic is included on the following page for clarification.

Figure 18: Wiring Schematic for 480 V Test System

Color coding in Figure 18 was kept in line with that of the actual test system. See the top left of the figure for legend information. In this particular schematic, the system is HRG grounded. All wiring external to the 3-phase GFCI housing is shown.

Feeder Models

To correctly model characteristics of 3-phase cables (or lines, feeders, etc.), discrete component implementations of the following nominal- π three-phase mutually coupled cable network model were simulated and constructed.

Figure 19: Nominal- π 3-Phase Mutually Coupled Cable Network

The model of Figure 19 was verified as the same model used in the original V-HARM studies, and is the same model used in all EMTP simulations (the discrete RLC components are not shown, as the positive and zero sequence impedances are input). In this particular cable model, downstream ground connections (where ground fault current for end of the line faults is directed) are not made to a system or global ground, but rather are returned to the equivalent network of Figure 19 on the far end and then connected to a system or global ground on the near end (that closest to the bus). Such is the reason for

ground wires running to each feeder in Figure 18. This is necessary for inclusion of Z_0 in calculations, lest it be shorted by ground connections on both sides.

Nominal- π cable models with non-equal positive and zero sequence impedances are mutually coupled. Assuming zero sequence values equal to positive sequence values, an uncoupled three-phase network results; brief simulation in EMTP showed an almost twofold error in charging currents on the non-faulted feeders when this assumption was made. Thus the more complete model of Figure 19 is justified.

Components were obtained, matching as closely as possible the parameters used in the original V-HARM studies, with what is commercially available. Positive sequence resistors are within 2% error, capacitors within 10%, and inductors within 20%. Negative sequence resistors are within 3% error, capacitors within 34%, and inductors within 10%. To maintain relative sizes of feeder lengths, parallel and or series combinations of the same components were used to obtain one half or two times multiples of impedance values on various feeders. All series components are rated at 10 A RMS, and all shunt components at 1000 V-peak.

Figure 20: Feeder Model Close-up (Feeder 4 shown)

As shown in Figure 20, components were mounted and wired onto an acrylic board. Terminal lugs provide wire mounting to and from the feeder. The four feeder models are based on 3-phase plus neutral 2/0 copper conductors in steel conduit with lengths of 500 feet, 1000 feet, 2000 feet, and 500 feet with 1 μF surge capacitors, as specified in [9].

Source Transformer

A Hammond 15 kVA 240-480/277 V Δ -Y transformer was used to obtain the 480 V system voltage from the 3-phase 240 V available at the lab bench. 40 A fuses on the bench limits total continuous current to 16 A at 480 V. Thus, maximum allowable complex power draw is 13.3 kVA; hence the 15 kVA rating.

Holes were drilled on both sides and Hampden HB-5 receptacles were attached for quick-release connections to the 240 V source at the bench and the 480 V connections

on the test system. Existing 10 AWG wires with quick-release plugs and spade ends were spliced to provide cabling as needed.

Figure 21: 15 kVA 240-480/277 V Δ -Y Transformer with Internal Wiring and Side Connections

Figure 21 shows the internal 10 AWG wiring and side connections on the transformer. The transformer chassis ground connection is not shown but was later added from the bottom left receptacle to the chassis ground bar.

Contactors

For prototyping purposes, interruption via circuit breakers with shunt trips is unnecessary as motor contactors can provide the same functionality at a lower cost- no internal thermal magnetics are required. Fuji SC-E05 contactors with 480 V auxiliary coils were chosen for opening and closing of the main and feeders, and to initiate faults.

Figure 22: Fuji SC-E05 Contactor

One such contactor is shown in Figure 22. The SC-E05 allows 32 A non-inductive breaking current and is opened and closed by 480 V auxiliary contacts. The high voltage contacts allows a fast contactor open and close response time of 5-16 ms (compared to alternative DC and lower AC coils), with the trade-off of managing 480V AC in the control system. In designing the prototype to reach a 25 ms break time, these contactors will take a majority of the allowable time.

Relay Driver PCB

In order to interface control of the 480 V SC-E05 auxiliary contacts with a General Purpose Input Output (GPIO) pin on a processor, a relay driver PCB with interposing relays was designed. The SC-E05 has a seal-in current of 200 mA and a holding current of 20 mA (at 480 V), while a GPIO pin can be expected to source less than 1 mA at 3.3 V. There is a large disparity in power switching capability. Two interposing relays were chosen - a BJT based inductive load driver IC (Fairchild MDC3105), and a high voltage reed relay (Coto 9104). Later a Meder SIL05-1A85-76D2K reed relay with a higher power rating but same form factor, capable of switching

the coil inrush current, was substituted for the Coto relay. Both reed relays have reverse biased diodes across their switching element to freewheel current from the auxiliary coils inductive kick. Total clearing time for the interposing relays is less than 0.2 ms, and negligible compared to that of the contactor. On- and off- time testing of the contactors using the relay driver PCB verified operation within one cycle for both opening and closing.

Figure 23: Relay Driver PCB Layout

Figure 23 above shows the PCB layout for the relay driver. The design includes a 5 V regulator and layouts for five sets of the two interposing relays; each board allows control of 5 contactors. 0.1" connector pins are supplied for interfacing with GPIO pins on a micro-controller. A 10 mm clearing distance was maintained between 480 V traces and the PCB included a solder mask to ensure isolation.

Figure 24: Assembled Relay Driver PCB

The assembled board is shown in Figure 24, which also shows stacking of the relay driver PCBs so that more than 5 contactors can be controlled in a small footprint.

DC Power Supply

The final solution draws power from the 480 V mains, rather than a nearby wall-wart. A 3-phase 120 W 24 VDC switched mode power supply (SMPS) is used to provide galvanic isolation from the 480 V mains.

Figure 25: 3-Phase 120 W 24 VDC SMPS

The SMPS, shown in Figure 25, supplies power to the 3-phase GFCI. Whenever the mains are on, the relay is operational.

The 24 VDC rail from the SMPS is tied directly into two DC/DC converters, as the processor development board power distribution required a 6-18 V input, and the sensor PCB required split supplies.

Figure 26: System Power PCB Layout

From Figure 26, it is seen the board accepts 24 V from the SMPS, and provides a separate 12 V and +/-12 V rail from a 17 W and 15 W DC/DC converter, respectively. Floating grounds are tied together to ensure noise immunity in digital logic levels. Chassis ground is isolated from the relay ground due to the DC/DC converter's floating output - seen as the separate green wire in Figure 18.

Load Banks

To allow pre-fault load currents and high power low impedance fault resistances, a load bank for each of the four feeders was designed and ordered. Each load bank has three levels, each level containing three 69.3 Ω 1500 W resistors and one 0-5 k Ω 1500 W

slide-adjustable resistor. For reference, a single $69.3\ \Omega$ resistor allows 4 A at 277 V. Acrylic connection panels for each load bank were laser cut, providing a 5-way binding post corresponding to each resistor tap. With stackable banana plug connections, many series or parallel combinations can be realized.

Figure 27: Load Banks with Connection Panels

16 AWG wire was used with banana-jack connectors to create the cables which connect the load banks to the end of the feeders; this is the only instance where 10 AWG was not used in a series element, however the use of banana-jacks required it, and the increased resistance is assumed negligible, compared to the load resistance.

A back panel was made such that wiring from the end of the feeders to the load banks is accomplished via 5-way binding posts as shown in Figure 28.

Figure 28: Back Panel Binding Post Connections

Available connections on the back panel include the three phases, neutral and end-of-feeder ground connection point for each feeder and the main, and access to both terminals on a single phase of the fault initiating contactor.

Neutral Grounding Resistor

The solidly grounded and ungrounded systems are achieved via a simple wire connection of the transformer neutral to the chassis or system ground. The HRG system, on the other hand, requires a separate neutral grounding resistor (NGR). The NGR was manufactured at $138\ \Omega$, the same as the original V-HARM studies' NGR, to limit resistive fault current contribution to 2 A. A 700 W power capability gives 26% over-rating during a long-term solidly grounded fault; no short time thermal ratings are considered.

Figure 29: Neutral Grounding Resistor

Figure 29 shows the 700 W NGR purchased from MF Power Resistors.

Fault Current Generation

A wide range of fault currents, ranging from milliamps up to thousands of amps is desired. In the test system, a 4 A fault limit is imposed so as to accommodate pre-fault load currents with the 16 A limit. Due to the availability of the series power inductances in the line models, only 10 A RMS can be sourced through the feeders. For these medium levels fault currents, the 70 Ω fixed and 0-5 k Ω variable resistors included in the load banks offer a number of series and parallel possibilities which yield large and medium fault currents.

For the smaller fault currents, especially those in the 5 mA region of interest, a small fault current board was assembled.

Figure 30: Small Fault Current Board

The board shown in Figure 30 consists of many 47 k Ω 25 W 5% chassis mount resistors, connection wires with small banana clips for quick inter-changeability, and 5-way binding posts for banana-jack connections. A single 47 k Ω resistor (shown) yields

5.9 mA of fault current at 277 V. Various other fault currents can also be created, with 5% accuracy.

Figure 31: Available Fault Currents from Small Fault Current Board

Figure 31 shows the fault current available from the small fault current board's many series and parallel resistor combinations. The range spans roughly 2 mA to 30 mA. With a little math, this graph is useful for any scenario when a desired current needs to be generated from a non-apparent combination of series and parallel same-value resistances.

To generate very large fault currents on campus, a single-phase 5 kVA 240:12 V Q005ERCF buck-boost transformer was used in isolation mode with the secondary shorted through steel rods.

Figure 32: High Current Test Set

The large surface area and increased resistivity of the steel rods offers a means to limit the current from the 12 V secondary without too much concern for overheating, however fault current was never allowed to flow more than 100 ms. Connections to the steel rods from the transformer secondary taps were made with 4x 10 AWG in parallel, as shown in Figure 32.

It is difficult to estimate the actual secondary fault current generated by the transformer. Due to inrush current, 25 A current probes on the primary became saturated. As shown in Figure 32, two steel rods were used; allowing a larger and smaller current in the 1 kA range. When a single rod was used to limit current, a Fluke 33 true-RMS meter in peak detection mode was used to measure the current through each of the 4 10 AWG wires. Summing up these peaks leads to an estimate 1400 A RMS of fault current, whereas measuring the secondary impedance with an LCR meter (applies a 1 V signal at 60 Hz) estimates 800 A. With two rods, estimates are 831 A and 631 A, respectively. Despite discrepancy in these estimates and an inability to precisely measure the fault current, the generated currents are over a hundred times the rated primary current of the W-20, which is the goal of this set of testing.

Resolution of EMI Issues via Phase Restricted Switching

This section details resolution of EMI induced via the contactor opening, and is included here as a test setup issue. It requires understanding of the 3-phase GFCI hardware capabilities explained in Chapter 6.

Despite the reverse biased diodes across DC coils of the reed relays, there is no inherent means of dealing with the inductive kick generated when interrupting non-zero current in the contactor's auxiliary control AC coils. By forcing a non-zero current through the inductive auxiliary coils to zero when opening the contactors, a tremendous inductive back EMF is induced, which in turn is responsible for emanating EMI along the lines connected back to the transformer. This EMI was coupled to the Blackfin development board and sensor PCB, and somehow collapsed the ground or power rails of the board, at which point the onboard power monitor circuitry shut down the processor. The original development board suffered permanent damage from repeated exposure to EMI. Alternatively, the ADC on the sensor PCB would sometimes crash.

The issue was attempted to be resolved through the use of an Avago HCPL-7710 opto-isolator array to isolate the relay driver PCB and processor control lines, ensuring unidirectional communication. Obviously this was done before a complete understanding of the issue as EMI. After EMI was confirmed as the source, no amount of shielding was able to completely eliminate the problem, although modest improvements were made. Diodes capable of withstanding the 480 V peak voltage are not commercially available, despite anti-parallel zener diodes being the most popular option with AC coils. So, a different approach was taken.

By restricting the contactors to only open near a zero current crossing, the induced EMI can be significantly reduced. A 47 k Ω resistor was placed between the same phases supplying the contactor coils, in order to establish a 10 mA reference current. This current was measured by a CT and input into a spare channel on the sensor PCB. When passed through the digital filter, rather than determining the magnitude, the phase of the extracted 60 Hz phasor was determined. Thus, a reference is created in the processor with which it is possible to control when a contactor is opened.

Due to the 960 Hz sampling rate, phase information is discrete, and so there is a 22.5 degrees minimum phase window over which the contactor can be restricted to open, if opening is to be guaranteed within a half cycle. Delaying more than half a cycle is unnecessary since there should be two optimal phases, 180 degrees apart, during which EMI is a minimum. Because of the way the coil operates, it is not straightforward to calculate the optimal opening angle. The AC coils of the contactor have a shading ring, which guarantees there is some minimum force generated to keep the contactor closed when current in the main coil passes through zero. The current through the shading ring does not add in phase with current in the main coil. As such, after much testing, optimal phase windows were empirically found to be centered about 146.5 and -33.5 degrees with respect to the reference current. Even with such a large window during which the contactor could open, this reduced EMI to the point that shielding was unnecessary and relay operation was never affected by contactor opening. Restrictions were not placed on when the contactors could be closed, as no current is interrupted during closure and EMI issues were restricted to contactor openings.

Intentionally withholding the contactor release in software had no effect on total contactor opening time. Average opening time over 22 trials for software delays of 1.04 through 8.33 ms were 12.6 ms without load current, and 12.1 ms with 2 A of load current. Maximum opening time never exceeded 17 ms.

Verification of System Properties

The following system properties are examined with the complete 3-phase GFCI signal chain, details of which can be found in Chapter 6.

Feeder Energization Error Currents

When a feeder is energized, the line charging currents in each phase begin to flow to ground. In theory these currents will sum to zero, however this was not found to be the case. This section details the adverse effects of feeder energization on the 3-phase GFCI testing.

First, steady-state errors are examined. Due to finite component tolerances (the 1 μ F surge capacitors have a 10% tolerance), there is expected to be a non-zero 60 Hz component present. Indeed, this is cause for great concern, for if the steady-state 60 Hz component of an energized feeder is greater than a few milliamps, the resolution of the system is significantly reduced, and if is greater than 4 mA, shock prevention cannot be implemented.

The extracted 60 Hz component from the digital filter was shown to be no more than 0.5 mA for feeders 1-3, and 2.2 mA on feeder 4. The larger steady-state error on feeder 4 is expected due to the much larger surge capacitors and their varied tolerances.

Figure 33: Steady-State Feeder Energization Errors

Typical steady-state filter outputs scaled to mA for a system in which all feeders are energized is shown in Figure 33. The main (dark blue) shows the sum of feeders 1-4; it is not significantly greater than feeder 4 itself. Due to this error, as will be shown later, resolution is reduced when attempting to detect fault currents near the 5 mA threshold at the end of feeder 4. Peak-to-peak ripple over 400 samples was found to be less than 0.2 mA on all feeders, and feeder 4 exhibits a sub 1 Hz drift ranging from 0.5-2.2 mA.

For the most part, since the steady-state errors are below the 5 mA tripping threshold, they have no significant effect on relay operation. Field testing is needed to confirm whether or not systems with large surge capacitances are an issue, as it is unreasonable to assume the tolerances of such capacitors is low enough to keep the steady-state error significantly below the tripping threshold. In an interview with the author of [9], field tests at the Chevron Richmond refinery indicated this to be a non-issue.

Transient Characteristics of System

While correct steady-state characteristics are necessary, for GFCI prototyping purposes transient characteristics of the test system are most important; the system should accurately model transitions from pre-fault to post-fault conditions.

The transient response of the test system is questionably accurate. The actual line charging current in each phase is highly distorted, as shown in Figure 34 below.

Figure 34: Feeder 4 Phase-A Line Charging Current

The line charging current in phase B and C of feeder 4 are similar.

Figure 35: Feeder 4 Phase-A Energizing Harmonic Content

The harmonic content of Figure 34 is shown in Figure 35, as measured by a PowerSight 3500 meter. Current THD was measured at 8.85%. Any triplen harmonic content shown in Figure 34 or Figure 35 passes through the CBCT as zero-sequence current; a majority of the CBCT output due to feeder energization occurs at 180 Hz. The major source of the harmonic distortion is unknown. All elements of the feeder model are passive and so should not create any harmonics; the series inductors do not saturate until 14 A. A PowerSight 3500 meter found negligible THD in the source voltage. Shorting out all series elements in the feeder models had no effect.

The digital filter successfully rejects all of the harmonics, however it only does so in steady-state, when the filter window is completely filled with the harmonic content. As the 180 Hz progresses through the FIR filter immediately after feeder energization, output is erratic and overshoot occurs.

Figure 36: Filter Transient Response to 180 Hz Feeder 3 Energization

Shown in Figure 36 is the transient response of the signal chain after feeder 3 is energized. A 180 Hz component associated with feeder energization appears, and until the filter settles, filter output overshoots the steady-state value.

Naturally, the overshoot shown in Figure 36 is much larger when energizing feeder 4, and actually surpasses the 5 mA tripping threshold. As discussed later, due to the variable length filter delay, this momentary surpassing of the threshold turns out to be a non-issue.

The main issue encountered with the test system is that, in addition to the steady-state 180 Hz component, there is a large, short-duration inrush current immediately following feeder energization; it is most noticeable on feeder 4.

Figure 37: CT Output Post-Energization

As seen in Figure 37, the moment the feeder is energized a large transient occurs so as to re-orient the charge on the capacitors with that required by the newly applied voltages. Since closing is asynchronous, the severity of the inrush varies. The magnitude of this impulse is much larger than the 180 Hz line charging components, and can so have deleterious effects on the relay, which are unable to be over-come with the variable length filter. It is not clear whether or not this is a true indication of what occurs during cable energization, or if this is a peculiarity of the test setup used. Again, field tests with the 3-phase GFCI would be the appropriate next step to take.

What follows is included in this section as it is assumed to be unique to the test setup, however requires understanding of the 3-phase GFCI operation as detailed in Chapter 6.

When the transient charges or discharges associated with the feeder RLC elements are sampled by the ADC, the filter output response is invariably affected for the following 20 samples. The 960 Hz sampling rate and the 300 μ s averaging per sample make the amount of the short-time transient quantized by the ADC difficult to predict. In practice, it is rarely detected, but nonetheless has occurred.

The basic problem is as follows. A feeder is closed, the transient inrush current happens to be sampled, and it is large enough such that 5 samples later (when the total filter delay is decided) the maximum of all the filter outputs is high enough to cause an immediate tripping logic evaluation. Since the transient impulse and any triplen harmonics are still working their way through the filter, filter output is expected to be over 5 mA; thus by relay logic the breaker is immediately re-opened. Due to high channel-to-channel matching in the ADC, all relevant filter outputs will respond similarly to a sampled transient, and selectivity is not compromised. Thus this does not pose an issue for feeder de-energization: in the event a transient is sampled and a trip incurred, the relay will only attempt to open the breaker whose opening action incited the relay response. The problem is that when a feeder is attempted to be closed-in, detection of an impulse may incorrectly lead to presumption of a pre-existing fault and re-opening of the feeder.

It is not possible to simply withhold tripping action while closing in a feeder, as that would preclude detection of pre-existing faults- for which the same timing

requirements apply. Rather, the logic must be able to detect and remove the happen-chance transients' effect on relay logic, or accept the fact that it may require multiple attempts to close in a feeder with a sufficiently large line-capacitance to ground. In the case of the latter, inability to ever close in the feeder would indicate a true pre-existing fault condition. Since secondary attempts to close a fault-free feeder are statistically very successful, the problem was circumvented in the lab by simply manually reclosing feeder 4 when an incorrect tripping decision was made upon energization.

Next is described a basic approach to intelligently detect and remove the effect of the transient samples, so as to nullify their effect on relay logic. This assumes these transient inrush currents are a problem in a real installation. This approach could be implemented if so desired, but was not further investigated.

When a transient is sampled, due to the relatively slow sampling rate and short-time nature of the inrush current, it is expected to appear as the sample corresponding to the first post-threshold filter output. Differentiating this first input from the remaining 5 inputs, of which all corresponding 6 filter outputs are used to determine the filter delay, can give an indication of whether or not a transient was inadvertently sampled. The problem with simply detecting the transient and withholding tripping was described above; it is necessary to completely remove the effect of the single sample inrush transient.

An algorithm used in the lab with moderate success was to find the 2nd difference amongst all post-threshold currents and return true for an impulse if the largest 2nd difference was greater than the twice the next largest. Other approaches could include

using the pre-threshold zero or DC offset value; or more simply the magnitude of the first post-threshold input.

Nevertheless, once detection of the impulse is confirmed, the effect must be removed. Stated simply- the first post-threshold input is forced to zero, and all filter outputs are reset and re-calculated. As all filter outputs depend on the 20 previous filter inputs, this becomes quite complicated to implement in code. The chosen Blackfin processor would be more than capable of allowing for this additional computation. This would lead to an additional 1-sample delay, even for closing into a pre-existing fault that requires the quickest relay action. If an additional sample delay is required, always recalculating filter outputs with the first sample (responsible for passing the threshold) forced to zero at the filter delay decision point would completely remove susceptibility to inrush currents.

Fault-initiation induced transients can be disregarded since they only increase relay response time, and it turns out the magnitude of the required fault current to create the previously described nuisance trip effect requires an instantaneous trip anyways. Plus, fault current can increase without limit, whereas maximum transient discharge is fixed due to characteristics of the line.

Again, these errors merely prevent proper breaker closure, which is desired for precluding closing into pre-existing faults. Since they only over-estimate actual fault current, selectivity is not compromised and the worst-case scenario is a false-positive of closing into an existing fault. In almost every case the second closure is successful. Whether or not this is even an issue in a real application or is just due to the feeder models is unknown.

Verification of Steady-State Nuisance Tripping Currents

Since EMTP was validated as an acceptable simulation tool due to the agreement with V-HARM results, the test system was modeled in EMTP with nominal component values. No tolerances were accounted for, and the only parasitic element included was the series resistance of the inductors.

Figure 38: ATP-DRAW EMTP Schematic of Test System

Breaking out the feeder models to actual component values shown in Figure 38 allowed tolerances to be varied, but this did not create any appreciable effect or explain the energization error currents. The simulations are overly simplified to an extent.

Prior to GFCI implementation, steady-state nuisance tripping currents in the test system were measured and verified with those expected from EMTP simulations of the test setup. Tables comparing the results are available in Appendix E. Again, results agreed within 5% or 1 mA, whichever is greater, with nuisance tripping currents highlighted in yellow. Comparison between Appendix A and Appendix E show the test systems ability to produce similar fault situations.

Transient filter output buildup for a few fault currents are included here to highlight the nuisance currents and values which are to be passed into the tripping logic. Faults are initiated after feeders are energized and so only steady-state energization errors are present. These are all measured values produced by the test system.

Figure 39: Solidly Grounded System, 700 Ω Fault on Feeder 1

Figure 39 shows tight matching between the main and faulted feeders CTs, and correct fault current level as indicated in Appendix E for the stated fault.

Figure 40: HRG System, 0 Ω Fault on Feeder 3

A bolted fault at the end of feeder 3 on the HRG system provides both feeder 3 and feeder 4 with currents over the 5 mA threshold, as shown in Figure 40. Referencing Appendix E, all feeders are actually over the 5 mA threshold; nuisance tripping currents are confirmed.

Figure 41: HRG System, 47 k Fault on Feeder 4

Figure 41 shows filter buildup to a 5.9 mA fault on Feeder 4. This fault current is too small to cause any nuisance currents, and is off by about a mA due to vector addition with the 2 mA energizing error current.

Other than the inrush current and 180 Hz component of the line charging currents, the test setup performs as desired; nuisance tripping currents are available for the 3-phase GFCI development and prototyping.

CHAPTER 6

DETAILS OF 3-PHASE GFCI DIGITAL RELAY

The components which make up the 3-phase GFCI prototype are explained, and the program flow is summarized.

Hardware

The final prototype hardware assemblage is shown below in Figure 42.

Figure 42: 3-Phase GFCI Prototype Hardware

The reed relays and relay driver PCBs are located outside of the aluminum frame shown in Figure 42, and so are not shown; the figure is otherwise self-explanatory. Shielded wires are grounded to the frame, and the chassis is grounded to the same 120 V

supply as the PC in order to ensure successful UART communications. Star grounding amongst the various PCBs was implemented with the chassis ground connection at the DC/DC converter. No extraordinary attempt was made at miniaturization.

The 3-phase GFCI is composed of the components shown below in Figure 43.

Figure 43: Digital System Block Diagram

A number of control lines are required to interface the processor with the sensor PCB via GPIO pins; see the 6 and 12 inch jumpers in Figure 42, and the AD7606 datasheet for more information. To interface with the contactors, GPIO pins are connected to opto-isolators and then an output enable buffer, as shown in Figure 42. The output buffer directly controls the reed relays, which in turn open and close their respective contactors.

A PC HyperTerminal is used to interface with the user, to print transient results, and to verify functionality. A user is able to initiate faults and control the opening and closing of the contactors, however relay functionality is maintained independently of a PC (e.g., if fault and contactor control were to be implemented elsewhere).

ADSP-BF538F Processor

A processor with exemplary DSP properties was chosen, so as to provide ample computing capability, namely with regards to filtering. Thus, the limiting requirement for selecting a processor within the Analog Devices Blackfin processor family (an arbitrarily chosen product line) was the available number of GPIO pins. While the

prototype utilizes only 6 contactors and monitors only 7 ADC channels, the selected processor is capable of working with up to 15 feeders (e.g., for a typical 16-circuit lighting panel), for which it has sufficient GPIO pins. In this prototype, the chosen ADSP-BF538F processor is implemented on an ADSP-BF538F EZ-KIT Lite development board, reproduced below in Figure 44.

Figure 44: ADSP-B538F EZ-KIT Lite

The processor itself features a 533 MHz core clock, 3x 32-bit timers, UART, SPI with multiple slave selects, 54 GPIO pins, and hardware pin change interrupts. The chip was chosen since it meets the requirements of the system: simultaneous sampling, filtering, and processing of up to 16 channels at a 960 Hz sampling rate. Access to most of this functionality is provided on the development board via .1" pin-headers.

The ADSP-BF538F is easily able to process 16 channels in the required time: running the final code at a 250 MHz core clock and a 100 MHz system clock (about half its maximum speed) the processor idles for approximately 600 μ s, or 2/3 of the total time between samples. Of the used processor time, 350 μ s is spent waiting for the ADC to finish a conversion— time during which the processor could be running calculations if necessary. The time the processor spends transmitting/receiving, sampling, and

processing 7 channels is under 100 μ s, which indicates ample processing capability, even at 16 channels. Compiler optimization was not used.

Sensor PCB design

The sensor PCB was designed to house the ADC and over-voltage detector circuitry, in order to interface CT output with the processor (note, however, anything "over-voltage detector" related can be neglected as it does not appear in the final solution). Due to the sensing requirements of the 3-phase GFCI, the only adequate ADC and analog front end solution necessitated an external IC solution; ADCs typically integrated into micro-controllers are not adequate. The AD7606, an 8-channel, 16-bit, 200 k (simultaneous) sps, bipolar input, data acquisition system was chosen to this end. The chip contains all necessary analog front end circuitry, and with two AD7606s, 16 channels can be monitored. Each AD7606 features on-chip hardware oversampling. Since the sampling rate is relatively low, for each 960 Hz sample, 64 samples are taken over a 250-350 μ s interval and averaged. This greatly improves resolution at the milliamp level, and affords over 96 dB of dynamic range (nominally). Additionally, the chip has integrated 16.5 V overvoltage protection (although this is not utilized due to the 6.8 V CT TVS diode protection).

The sensor PCB which houses the AD7606s was designed on a 21 square inch 4-layer PCB, as shown in below Figure 45.

Figure 45: Sensor PCB CAD Layout with Comments

Figure 45 shows the top copper layer in red, the bottom copper layer in green, and the solder mask in yellow. The power and ground planes are not shown, as they provide no relevant information, other than to note that analog and digital ground planes were roughly isolated to the left and right of the AD7606s. The sensor PCB was designed to accept inputs on up to 16 channels, and output data over SPI with 4 data out lines. In the prototype only 7 channels on a single AD7606 with one data out line was used. Full scale range is hardwired to $\pm 5V$. Pins on the AD7606 are 0.5mm pitch and had to be soldered using a reflow oven.

Software

The variable length filter and C code used for implementing the 3-phase GFCI prototype are explained. Refer to Appendix F for the explicit C code, which includes comments for clarification.

Variable Length Filter Delay

Due to timing restraints, it is not always possible to wait for filter outputs to settle before a trip must be initiated. On the other hand, filter output may momentarily surpass 5mA, even when there is no ground fault current- as an example, the 180 Hz feeder energization currents (shown in Chapter 5, Figure 37) always results in filter output momentarily rising above 5mA on feeder 4. Intelligently adjusting the total filter delay based on an initial estimate of the fault current, such that the total response time is always kept under the UL required curve, allows as much time as possible for a “most-informed” decision to be made by the logic.

The implementation is as follows. First, a maximum filter delay is chosen, so that for the smallest currents the best resolution is obtained. A conservative value for a maximum relay response time of 45.1 ms is determined by the following contributions:

- 1.04 ms (worst-case) to sense the fault- ADC sampling is not synchronous with fault current passing the threshold
- 25 samples or 26.1 ms for the filter to completely settle- chosen intentionally larger than the theoretical 20 samples
- 17 ms for the contactor to open
- 1 ms error

The Class A GFCI tripping curve detailed in Chapter 1 indicates that for faults 175 mA or less, it is allowable to take the maximum 45.1 ms to clear the fault. Although even longer tripping times are allowed for smaller currents, choosing the values above guarantees the filter will be settled and in steady-state; hence there is no point in extending the maximum time.

On the other hand, to meet the 25 ms minimum trip time for the largest faults referenced as 264 mA and greater, the maximum filter delay is constrained to:

$$\begin{aligned} 25 \text{ ms} - 1 \text{ ms (asynchronous)} - 1 \text{ ms (error)} \\ - 17 \text{ ms (breaker opening)} = 6 \text{ ms} \end{aligned} \tag{6.1}$$

Rounding down, (6.1) indicates that for fault currents greater than 264 mA, only 5 samples worth of time are allowed (after a fault has occurred) before a trip decision must be made. The filter delay is implemented so that 5 additional samples can be taken after the filter output surpasses the 5 mA threshold, and a trip decision must be made. Any fault time which accumulates before the threshold is passed is not taken into account; for the small fault currents which slowly build up to 5 mA, such large amounts of time are allowable that the error is negligible.

With the 5 additional samples leeway after the threshold is passed, filter output data is gathered in order to determine the remaining filter delay. The total or remaining filter delay is always decided after the 5th post-threshold sample. Total filter delay then ranges from 5 to 25 samples, where for currents ranging between 175-264 mA, delay is calculated using a linear interpolation.

In order to remain conservative, when determining the total delay, the maximum of all channels' filter outputs is used to (roughly) indicate the magnitude of the fault. Over-estimating the fault current will only increase response time. Set-points are chosen to

clear faults of 120 mA (rather than 175 mA) or less within 45.1 ms, and faults of 200 mA (rather than 264 mA) or greater within 25 ms. Assuming the estimates of the fault current are accurate, relay response time is bounded by the curve shown below in Figure 46.

Figure 46: Ideal 3-Ph GFCI Maximum Response Time

The Class A, UL 943 curve shown in Figure 46 is the maximum allowable tripping time. Assuming that 6-samples worth of data is sufficient to make a correct fault decision, the variable length filter delay ensures Class A timing requirements are met. For complete results, see Chapter 7.

C Code Summary

The program runs entirely in a timer-interrupt delayed loop, aka "round-robin". Each 960 Hz interrupt results in the following program sequence: Data Sampling and Verification, Filtering, Relay Tripping Logic, Updating of Contactor States, and PC Communication. The program flow is illustrated below in Figure 47.

Figure 47: High Level Software Flow

The flow of Figure 47 runs indefinitely. All processes but the Relay Tripping Logic run on a sample-independent basis; they are not affected by their previous outputs, and perform the same operation (as a function of their input arguments) each time through the loop. The Relay Tripper Logic process, on the other hand, can span multiple loop instances, due to the variable length filter delay. As will be explained, the relay tripping logic is examined on a 960 Hz basis, but may result in different operations, depending on previous tripping logic operations; this is how a multi-sample process is implemented in round-robin flow.

When the program enters the 960 Hz loop, it begins in the Data Sampling and Verification stage. The processor initiates an ADC conversion, and the ADC oversamples its 8 channels and stores the results; this takes 250-350 μ s. The processor has many levels of error checking to ensure the ADC is healthy and the conversion occurs in the expected timeframe. Subsequently, the processor twice retrieves the values for each channel over SPI. Any errors (e.g., EMI induced) are mitigated by requiring that

all of the samples from each transmits are identical. If there is an error (i.e. they are not identical), the samples are retransmitted and compared again until successful.

Next, the samples from each channel are input into the cosine filter. Values in arrays holding the most recent filter 16 filter inputs and 5 filter outputs are updated in the style of a circular buffer. There is a pointer keeping track of the beginning and end of the circular buffer, which allows the use of a linear memory array to mimic a circular buffer; moving a pointer takes fewer cycles than shifting every element. The symmetry of the cosine filter coefficients was used to reduce the total number of multiplication operations from 16 to 4. This was done by a preliminary addition or subtraction of the corresponding previous filter inputs prior to the multiply-accumulates. Updated filter outputs are scaled to milliamps, which are then input into the tripping logic. Channel 7 is used to as a reference phasor and it returns the phase rather than magnitude; it is not used in the tripping logic.

From here the program enters the Relay Tripping Logic stage. Each of the filter outputs are checked to see if any are above the 5 mA threshold. If none are, the program continues to the next stage; otherwise, the program initiates the multi-sample variable length filter delay process. If the variable length filter delay process is already running, the new filter outputs are simply passed into the variable length filter, regardless of whether or not they are above the 5 mA threshold. If the variable length filter process has just completed, this portion of the code will again check if one or more of the filter outputs are above the threshold. If so, tripping action is taken via the magnitude comparison logic proposed by Hamer: if the main is within $\pm 40\%$ of the largest feeder, the largest feeder is to be tripped. Otherwise, the main is tripped. In [9], matching was

originally proposed at $\pm 20\%$, however due to errors with the feeder energization error currents significantly altering measurements near the 5 mA threshold and issues with mild remnant flux, this range was expanded to overcome any incorrect bus trips.

After a trip decision is made, tripping is disabled for the next 100 samples (approximately 104 ms) to allow for the contactor to open, the fault current to clear, and the filter outputs to settle to their new steady-state values.

In the Updating Contactor States stage, control over the GPIO pins which determine the contactor states occurs. In the tripping logic (and elsewhere in the program), global variables with desired contactor states are updated. Here, actual states of the contactors are compared to the desired states indicated by the global variables. As described in Chapter 5, contactors are opened only during an opportune phase window. Naturally, the phase varies each time through the software loop. Contactors that need to be closed are done so immediately.

The final stage of the program is PC Communication, useful only in prototyping. User controls are summarized as follows: 'm' toggles the Main contactor; numerals 1-4 toggle the respective feeder contactors; 5 controls the fault contactor; 'h' closes the fault contactor for 100 ms before automatically reopening it (high-current testing); and 'x' opens all contactors (to quickly avoid inopportune feeder opening and resultant EMI when powering the system down). All of the above, excluding 'x', also cause the program to store the next 100 input and output samples. Whenever 'p' is asserted, the program will print these 100 samples associated with the last switching event, along with tripping data such as timing and which contactor tripped, and will pause the system until a 'c' is received from the PC. Relay functionality is lost until 'c' is pressed. When the

system is continuously running, a single filter output for each channel (main, feeders 1-4) is printed to the PC each cycle such that any given filter output is updated on the PC every 6 cycles. This is adequate only for steady-state verification of fault currents.

After completing the PC Communication stage, the processor idles and waits for the next trigger of the 960Hz main while loop. The processor idles for approximately 600 μ s, or 2/3 of the total time between triggers.

Signal Chain Characterization

The following describes the digital filter implementation and prototyping. The filter was originally implemented on a PSOC5 FPGA for initial verification, then the AD7683 and Atmega2560 for characterization of a 16-bit signal chain, and finally on the AD7606 and BF538F development board. Although the full-cycle filter and a variable length filter delay was ultimately used, data regarding the half-cycle filter is included.

Optimized filter code was first verified in a C environment, where outputs could be verified with known inputs. During hardware implementation, desired variables were recorded in memory for a few hundred samples on a per sample basis (yielding pre- and post-fault information), then communicated back serially to a PC hyper-terminal. This allowed transmission of data without the need to additionally meet real-time constraints of chip functionality. A Saleae logic analyzer was used to aid in developing communication between the ADC and processor.

The following graphs, generated from a PSOC5 FPGA, give a good qualitative idea of desired filter responses.

Figure 48: PSOC5 Steady-State Filter Output, 60 Hz Input

Figure 48 indicates the filter properly extracting the peak value of a 60 Hz input signal.

Figure 49: Steady-State Filter Output, 120 Hz Input

Figure 49 indicates the filter properly rejecting a 120 Hz 2nd harmonic.

Figure 50: Filter Transient Response, 60 Hz Input

Figure 50 shows the filter output during the transient state. The output settles after ~24 ms, and reaches 92% of the final value at 14 ms. Output settling time greater than 1.25 cycles in this case is explained in that, referring to the filter input, the first few samples are not from the steady-state sinusoid. This will not be an issue due to use of the variable length filter delay and high channel-channel matching.

No data from the Atmega and AD7683 intermediary tests is included and only information from the final solution is given. The AD7606 was set to oversample and average 64 times during a 300 μ s interval immediately following the 960 Hz sample interrupt, and configured for the +/-5 V full scale. With 16-bit resolution, one least significant bit (LSB) represents 153 μ V.

Due to oversampling, the noise floor of the signal is nearly zero. Figure 51 below shows the sampled values when no primary current flows through the CT.

Figure 51: 500x Sample Noise Floor of AD7606

The DC offset in Figure 51 cannot be explained by improper biasing - .5% resistors were used in the split AA filter (the 2.5 μ A bias current cannot account for 3 LSB of difference). It is unknown why the DC offset exists, and it does vary on each channel, however that it does not change over 500 samples indicates minimal AC noise. An additional 2000 samples were taken with only one 4, a single LSB deviation, appearing. This data, taken on Channel 1, appears as somewhat of a fluke; DC offsets from the remaining channels were found to vary between two LSBs much more often than seen on Channel 1 - but were still restricted to only two distinct values.

An optimal burden is selected with trade-offs made concerning dynamic range and peak-to-peak filter noise at a 5.9 mA input. It was discovered that the peak-to-peak filter output ripple was inversely proportional to the burden. Since for smaller burdens, secondary voltages are smaller, quantization error increases. Additionally, the effect of the oversampling averaging a non-linear sine-wave may contribute to some signal distortion, although this is difficult to measure. The AD7606 datasheet does not mention this effect and simply states a 1.5 kHz corner frequency with 64x oversampling. Table 1 summarizes the results.

Table 1: Burden Selection Parameters

Burden (Ω)	peak-peak filter output ripple in 5.9mA est. (mA)	Est. Max. Current (A)
180	0.17	9.4
301	0.14	5.8
402	0.06	4.4

The maximum current is an estimate; conservatively estimated effective turn ratios are used. Due to a majority of systems exhibiting line charging currents below 5 A, choosing the 402 Ω resistor as a burden allows greater resolution. The effect of the HPF formed by this increased resistance will be accounted for in the effective turn ratio, determined when linearity of the signal chain is measured. Further optimization of the burden can be chosen if needed for systems with higher line charging currents. Data from this point on is assumed with a 402 Ω burden.

Steady-state measurements are investigated to calibrate the accuracy of the signal chain. A primary current was established from the 480 V source and available resistors, whose resistance was measured. Sampled ADC values and both full- and half-cosine filter outputs are included.

Figure 52: Signal Chain Input and Output Data, 0.98mA Primary Current

Figure 52 shows sampled inputs, and both filter outputs scaled to mA using a 620 CT turns ratio. Input current in mA is determined by subtracting the DC offset (as averaged over 400+ samples) and then applying the turn ratio (this produces a symmetric waveform for ease of comparison). Filter outputs are scaled to estimate the RMS value of the 60 Hz input. Quantization error in filter outputs is constricted to an approximately 0.5 mA band for the half-cycle and negligible for the full-cycle, considering the desired end-result mA resolution. As shown in Figure 52, milliamp resolution of the signal chain is verified.

Figure 53: Signal Chain Input and Output Data, 5.9mA Primary Current

Input and filter outputs for a 5.9 mA primary current is shown in Figure 53. By inspection, the steady state filter output is a stable estimate of the 60 Hz input current near the tripping threshold.

A complete characterization of the linearity and filter peak-to-peak output ripple is shown in Table 2 below.

Table 2: Steady-State Filter Characteristics

60Hz Input Current (mA)	Full-Cycle Cosine Filter			Half-Cycle Cosine Filter		
	Average Filter Output (mA)	Filter Output Peak-Peak Ripple (mA)	N-eff	Average Filter Output (mA)	Filter Output Peak-Peak Ripple (mA)	N-eff
0.98	0.98	0.12	619.58	0.99	0.44	614.71
1.96	1.94	0.07	626.99	1.94	0.40	625.55
3.93	3.85	0.06	633.24	3.85	0.40	632.97
4.70	4.63	0.09	628.77	4.64	0.46	628.53
5.90	5.75	0.06	636.18	5.75	0.42	636.03
11.80	11.52	0.10	634.93	11.52	0.44	634.90
23.60	22.95	0.07	637.49	22.95	0.42	637.49
29.50	28.69	0.12	637.60	28.69	0.48	637.60
61.80	59.91	0.11	639.52	59.91	0.43	639.52
96.85	94.43	0.14	635.91	94.43	0.44	635.91
162.00	158.57	0.12	633.42	158.57	0.47	633.42
340.00	334.56	0.57	630.09	334.56	1.16	630.09
1320.00	1283.91	1.20	637.43	1283.91	8.41	637.43
2007.00	1940.96	1.15	641.10	1940.97	17.55	641.09
3890.00	3723.25	0.46	647.77	3723.24	3.56	647.77

Data in Table 2 shows that steady state filter ripple remains within 0.15 mA up to 163 mA primary current for the full-cycle filter and within 0.5 mA for the half-cycle filter. Beyond 150 mA the full-cycle ripple remains within 2 mA, whereas the half-cycle ripple is increased to approximately 18 mA. Average filter outputs were calculated over 400+ samples, and an effective turn ratio of 632.5 is calculated via the 4.7 mA and 5.9 mA effective turn-ratios, since these are closest to the 5 mA trip threshold. This exact number is not extremely critical to overall sensitivity- 620 used in the original estimate led to about 0.2 mA error.

If desired, non-linearity in the transfer function can be accounted for via a curve-fit or a look-up-table, however, since the application is in protection (and not say, metering), absolute measurements are not important, save for at 5 mA, and only relative

currents or channel-to-channel matching matters. The transfer function of the CT at 5 mA is chosen to allow highest accuracy at the tripping threshold.

The characterization of error in the entire signal chain, when calibrated at 632.5 turns, is shown below in Figure 54.

Figure 54: Calibrated Signal Chain Percent Error

As shown in Figure 54, once calibrated the signal chain is linear to within 3%, from 1 mA to 4 A. Variations in channel DC offset do not affect linearity since they do not pass through the filter. Error at 1 mA can be attributed to magnetizing current, and error at the higher currents can be attributed to the CT approaching saturation.

Harmonic rejection is investigated with a function generator producing a voltage approximating a 1 A primary current, which is fed directly to the ADC inputs. Outputs in the following table have been averaged over 400+ consecutive samples.

Table 3: Filter 1 A Harmonic Rejection Data

1 A Input Frequency (Hz)	Full-Cycle Cosine Filter Avg. Output (mA)	Half-Cycle Cosine Filter Avg. Output (mA)
120.0	0.04	620.98
180.0	0.17	0.20
240.0	0.03	263.97
300.0	0.73	0.74
360.0	0.05	154.71
420.0	0.04	0.07
480.0	0.04	117.28

As shown in Table 3, both filters reject their intended harmonics to less than 0.1% of the input (a 60 Hz input confirmed un-calibrated input current of 878 mA). For odd-harmonics, the half-cycle filter output average is not the best numerical representation of the filter output, which is actually sinusoidal, but the most statistically probable. Ripple can account for the variation. Half-cycle rejection of even harmonics is poor at best.

The transient response of the full- and half-cycle cosine filter is characterized for a 60 Hz input. An effective turns ratio of 632.5 is used in the input current definition.

Figure 55: 5.9 mA Transient Response

Figure 55 shows transient response to a 5.9 mA primary. The half cycle filter responds to within 0.3 mA of the steady-state 5.9 mA in 12 samples, and the full cycle in 20 samples. This shows good agreement with expectations and the half cycle's 8 sample or approximately 8.33 ms faster response time.

Figure 56: 4A Transient Response

Figure 56 verifies similar transient operation for a higher 4A current. In this case, the half cycle filter responds in approximately 11 samples, and the full cycle in approximately 20.

CHAPTER 7

HIGH CURRENT AND PREVENTATIVE NUISANCE TRIPPING TEST

RESULTS

The 3-phase GFCI is tested for selectivity and response time at milliamp and 800 A fault current magnitudes, and the effects of remnant flux are investigated. Raw data, although not included, is available for any test case presented in this chapter, including filter input, output, and current probe measured fault duration.

Buck-Boost Transformer High Current Testing

As detailed in Chapter 5, a buck-boost transformer was configured to generate currents in the hundreds of amperes. As there is some uncertainty in the currents produced, to be conservative, currents of 800 A and 631 A are assumed as that generated during testing. The high current test set has a CT dedicated for measuring the large secondary currents. This CT replaced the "main" in the test system and was hooked up to the corresponding "main" input channel on the 3-phase GFCI. All high-current faults are represented as bus fault. High current feeder faults were not possible to implement, since fault currents greater than 10 A cannot be ran through the feeder models, and it is not possible to pass the secondary wires which carry the high current through the CTs used in the test system, since the four additional 10 AWG conductors would not fit through the CT window. Opening the main was then set to open the contactor which energized the buck-boost transformer, in order to simulate fault current interruption.

Clamped CT output, as anticipated from Chapter 4, conserves the volt-second area corresponding to the primary rated current.

Figure 57: Secondary CT Voltage, 800 A Primary Current (No tripping)

Figure 57 shows the rectangular 60 Hz pulse train generated from the CT when the primary current is much larger than the rated current. Estimating the area shown by the cursors yields 17.5 mV-s, whereas (4.1.b) with a 4.3 A rated current yields 20.9 mV-s; as a first-order estimate, the results are in agreement to 17%. This is quite good, considering the primary current in Figure 57 is at 100 times the rated current, and (4.1.b) assumes rated current.

The recorded filter inputs and outputs corresponding to the 800 A fault current of Figure 57 above appear below in Figure 58.

Figure 58: Filter Output (top) and Input (bottom), 800 A Bus Fault

The filter input values shown in Figure 58 are band-limited by the oversampling of the ADC, hence the lower slew rate. Nevertheless, the filter extracts the 60 Hz component, and filter output progresses to a steady-state value of roughly 4 A; since 4 A is much larger than the line charging currents available in the test system, large fault currents are allowed to simply pass through the signal chain.

With filter output progressing as shown in Figure 58, the 3-phase GFCI is able to make the proper tripping decision. As the only high-bandwidth measurement tool available on-hand to measure these high current levels is not able to measure even the primary current in the test set, the tripping times for very large currents are extracted from the ADC data of Figure 58. Tripping times are not overly important at these high fault current levels since they are above personnel levels, but the relay always responded with a minimal filter delay. Timing was not affected by the non-sinusoidal input waveform. Of more importance is that the relay maintains selectivity; see the discussion

on remnant flux at the end of this chapter. Selectivity cannot be thoroughly examined during a bus-fault.

For fault currents of 800 A, 631 A, and 16 A, the relay responded with a "trip main" decision at 5 samples of filter delay, and total tripping times (estimated from ADC data) of 19.8 ms, 27.1 ms, and 28.1 ms. Only single trials were done at these larger currents; statistical cases are included in the general system characterization later on in this chapter.

Figure 59: Filter Input, 800 A Bus Fault, Relay Tripping Enabled

As an example of relay response when the GFCI is enabled, Figure 59 shows the sampled primary current as a 800 A fault is being cleared. Considering that the contactor on the primary of the transformer is interrupting the inrush current plus the ratio current, longer than expected interrupting times may be due to the contactor interrupting a current greater than its 32 A rating. Regardless, the processor responded adequately, with minimum filter delay and proper selectivity.

System Characterization

The following data characterizes the 3-phase GFCI for fault currents ranging from 1 mA to 10 A. First, variability in relay tripping time due to variations in feeder, phase, and fault current magnitude is examined, followed by verification of the measurement resolution and shock prevention at small currents of 1-30 mA. Finally, all of the cases for the solidly grounded and HRG systems which originated in [9] are tested, up to maximum fault currents of 10 A.

Variability in Tripping Time due to Feeder, Phase, and Magnitude of Fault

In order to ensure that both the test setup and the 3-phase GFCI functioned independently of the faulted feeder and phase, faults were initiated on feeder 1 and 4 at 6 mA, 400 mA, and 10 A, and then repeated 20 times on each phase. The effect of closing a feeder into a pre-existing fault was measured only on phase A of feeder 1. Total duration of fault current flow was measured by an AM503B current probe amplifier, which was connected at the fault switch, so as to avoid measuring any feeder energization currents. The following data was collected on a solidly grounded system, and all tests were performed consecutively.

The summarized data table for the three discrete fault currents is shown below in Table 4.

Table 4: Relay Response Time for Variation in Fault Magnitude, Feeder, and Phase

			t-Fault (ms)			
			Average	σ	Min	Max
Feeder 1	6 mA	A	49.9	8.0	33.4	64.0
		B	52.3	9.2	40.2	65.8
		C	49.5	5.7	39.0	60.2
		A, exist.	53.6	4.6	45.2	61.8
	393 mA	A	22.5	3.2	16.6	27.8
		B	22.0	3.1	16.7	27.2
		C	21.3	2.9	17.2	27.2
		A, exist.	21.3	0.5	20.8	23.1
	4 A	A	20.6	2.8	18.0	27.0
		B	21.6	1.9	19.5	29.3
		C	25.0	2.2	23.3	30.1
		A, exist.	23.5	3.7	17.7	33.2
Feeder 4	6 mA	A	45.8	5.0	39.0	58.4
		B	45.4	7.6	36.4	61.6
		C	49.9	6.0	34.2	59.6
	393 mA	A	20.0	1.9	17.0	24.3
		B	20.2	2.2	16.6	23.3
		C	18.9	2.0	16.9	24.4
	4 A	A	19.4	1.8	17.5	24.0
		B	22.1	2.0	20.5	27.1
		C	22.8	4.0	16.0	30.7

As shown in Table 4, "Phase A, exist" is used to differentiate closing into an existing fault from all of the other cases where a fault was initiated on an energized feeder. Although only timing information is shown, in all cases proper selectivity was maintained. By inspection, the variability in relay response time between feeder 1 and 4, and amongst phases, is negligible. As desired, the average tripping time for all cases remained underneath the UL 943 Class A curve. For the 393 mA and 4 A faults, where the quickest 25 ms operating time is desired, maximum tripping times exceeded expectations as highlighted in yellow. The largest measured tripping time was 33.2 ms,

or 8.2 ms greater than the 25 ms limit. The extended delays are assumed due to contactor mechanics.

The fault current waveform for a tripping time greater than 25 ms is shown below in Figure 60.

Figure 60: Feeder 1, Phase A, 4 A Fault Interruption

As shown in Figure 60, a greater than desired total response time of 27.4 ms is measured via the cursors. Note, however, the discontinuity in the last half-cycle of current flow; this indicates the contactor attempting to break the current, but incipient arcing preserving some current flow until the next zero crossing. It is assumed such effects should be included within the contactors rated 16 ms maximum opening time, and it is unknown why there are instance that take longer than the datasheet stated 16 ms for the contactors to open.

When taking the data for Table 4, an issue with Feeder 4 tripping a 6 mA fault on phase B was discovered. Due to the feeder 4 energization error's sub 1-Hz variation from approximately 0.5-2.2 mA (as detailed in Chapter 5), depending on where in this

variation the fault was initiated, the error current could detract from the 6 mA fault current and bring the resultant current below the 5 mA threshold. Within a second or so the energization error would become small enough that the 5 mA threshold was passed and a trip would incur. All instances where this happened were disregarded. As concluded in Chapter 5, the appropriate next step would be to verify if such an energization error current exists in a real system.

Averages from the data of Table 4 super-imposed onto the UL 943 Class A time-to-trip curve are included at the end of the next sub-section.

Small Fault Current Resolution

With the small fault board, fault currents in the range of 1-30 mA are created, and system response is measured. All faults were initiated on feeder 1, phase A, and when a trip occurred, selectivity was again maintained; feeder 1 was always tripped. All feeders were energized prior to fault initiation. Fault current values are calculated based on nominal system voltage and the nominal resistance of the 5% 47 k Ω resistors used.

Table 5: Small Fault Current System Response

I-Fault (mA)	t-Fault (ms)	Trip?
1.2	-	N
2.0	-	N
3.0	-	N
3.9	> 40	N/Y
4.7	53.2	N/Y
5.9	60.8	Y
7.9	53.2	Y
11.8	46.8	Y
14.7	53.2	Y
17.7	54	Y
20.6	44.8	Y
23.6	45	Y
29.5	44.4	Y

Table 5 shows that the 3-phase GFCI begins tripping action at ground currents of 3.93 mA and greater, rather than as desired for currents 5 mA and greater and always within the 4-6 mA window. These errors resulting in a N/Y entry in the table are again due to the feeder 4 energization error currents. As measured at the main CT, the 3.93 mA fault current of feeder 1 is added to the feeder 4 energization error current, and the magnitude of the vector sum surpasses the 5 mA threshold. Subsequently, a trip decision is made by the relay, and feeder 1 is tripped since it is within +/-40% of the main. Data for this case is shown below in Figure 61.

Figure 61: Filter Output, Feeder 4 Energization Error Current Creating Incorrect Trip in Feeder 1 3.93 mA Fault

Figure 61 shows that when the fault on feeder 1 is initiated, it is just enough to bump the main above the 5 mA threshold and cause a trip. Note the constant 2 mA error of feeder 4 which exists before, during, and after the fault. Incidentally, this shows the precision with which the relay trips for currents just over 5 mA.

Removing this maloperation is easy: without feeder 4 energized, the system does not respond with a false trip.

Figure 62: System Response to 3.93 mA Fault without Feeder 4 Energized

With the energization error of feeder 4 removed, Figure 62 shows correct system response (no tripping), and accurate measurement of the 3.93 mA fault. Figure 62 is the expected result, neglecting the energization error currents (hence the N/Y entries Table 5). This same situation results in tripping at the tested 4.7 mA, however it is in within the 4-6 mA range, and therefore acceptable.

For reference, a typical 5.9 mA filter output progression is shown below in Figure 63.

Figure 63: System Response to 5.9 mA Fault, Feeder 1, Phase A

Figure 63 shows filter output rise and fall due to a 5.9 mA fault, and the high degree of stability in the relays estimation of the primary current once the filters are

settled. The small separation in feeder 1 and the main is caused by the feeder 4 energization error current.

The measured current corresponding to Figure 63 is shown below in Figure 64.

Figure 64: Measured 5.9 mA Fault Current, Feeder 1, Phase A

Figure 64 shows the variable length filter delay allowing an increased duration of fault current prior to tripping.

Disregarding the effects of the feeder energization error currents, specifically those due to feeder 4, the 4-6 mA window is easily met. This can be verified by observing the differences between the expected values shown in Table 5 with the steady-state values shown in the previous few figures. The result is that the (possibly) final step in verifying proper shock prevention is verification that the energization error currents do not exist in a real system.

The results of the system characterization so far are super-imposed on top of the UL 943 Class A tripping curve first presented in Chapter 1.

Figure 65: Measured and Ideal Relay Response Times versus UL 943 Class A Requirements

Figure 65 shows the response time of the prototype for the data of Table 4 and Table 5 overlaid onto the UL 943 Class A required curve. Small fault current data is presented in orange, and for each of the 6 mA, 393 mA and 4 A faults of Table 4, the average, maximum and minimum values are presented. These three metrics are representative of 140 fault clearing actions at each of the three current levels. On average, the 3-phase GFCI meets UL 943 Class A tripping requirements; should contactor opening within 16 ms be guaranteed, Class A tripping requirements can be met in all cases.

Test Cases

The 3-phase GFCI prototype test results for the HRG and solidly grounded systems described in tables A-1, A-2, A-5, A-5, and A-7 in [9] are examined; the many test cases allow variations in loading as well as system grounding. SLG faults are initiated after all feeders are energized, and a single test is performed at each fault level.

Case A-1 (corresponding to Table A-1 in [9]) is a solidly grounded system with balanced loading. Feeders 1, 3 and 4 are 4-wire Y-connected with $210\ \Omega$ load resistance per phase, yielding 1.1 kW per feeder. Feeder 2 is loaded at 64 mA per phase, or 52 W total. Summarized results are shown below in Table 6.

Table 6: Selectivity and Interrupting Time, Case A-1 (Solidly Ground System, Balanced Load)

	Fault Impedance	Tripped Feeder	t-Fault (ms)
SLG Fault on Main Bus	$70\ \Omega$	M	19.5
	$700\ \Omega$	M	18.5
	$47\ \text{k}\Omega$	M	56.4
SLG Fault on Feeder 1	$70\ \Omega$	1	19.6
	$700\ \Omega$	1	20.1
	$47\ \text{k}\Omega$	1	52.2
SLG Fault on Feeder 2	$70\ \Omega$	2	27.6
	$700\ \Omega$	2	30.3
	$47\ \text{k}\Omega$	2	48
SLG Fault on Feeder 3	$70\ \Omega$	3	18.5
	$700\ \Omega$	3	18.7
	$47\ \text{k}\Omega$	3	61
SLG Fault on Feeder 4	$70\ \Omega$	4	19.6
	$700\ \Omega$	4	22.6
	$47\ \text{k}\Omega$	4	59.2

Table 6 shows proper selectivity and Class A tripping times in all cases except for the larger faults on Feeder 2, which exceed the 25 ms mark by 2.6 and 5.2 ms. No bolted faults were performed; a $70\ \Omega$ fault impedance yields 4 A of fault current. This was necessarily limited to 4 A by the existing load current and available system kVA.

The same tests were performed again as case A-4, this time without loading, and the results are shown below in Table 7.

Table 7: Selectivity and Interrupting Time, Case A-4 (Solidly Grounded System, No Load)

	Fault Impedance	Tripped Feeder	t-Fault (ms)
SLG Fault on Main Bus	27.7 Ω	M	24.8
	700 Ω	M	21.8
	47 k Ω	M	47
SLG Fault on Feeder 1	27.7 Ω	1	23.4
	700 Ω	1	26
	47 k Ω	1	60.4
SLG Fault on Feeder 2	27.7 Ω	2	28
	700 Ω	2	19.4
	47 k Ω	2	58.2
SLG Fault on Feeder 3	27.7 Ω	3	20
	700 Ω	3	18.2
	47 k Ω	3	54
SLG Fault on Feeder 4	27.7 Ω	4	19.4
	700 Ω	4	26
	47 k Ω	4	51.4

More kVA is available without load current, and so the largest fault current created for case A-4 was 10 A at 27.7 Ω . Table 7 results are similar to Table 6, as expected, since load current should have no noticeable effect. Again, selectivity is maintained, and Class A timing requirements are met in all but 3 instances; response times are always kept within 30 ms.

In case A-7, feeder 3 and 4 are 4-wire Y-connected with 1.1 kW of balanced load. Feeder 1 is the same but connected with just 3 wires, foregoing the neutral, and feeder 2 is a delta-connected load of 109 mA per phase or 159 W total. Results are summarized below in Table 8.

Table 8: Selectivity and Interrupting Time, Case A-7 (Solidly Grounded System, Varied Load)

	Fault Impedance	Tripped Feeder	t-Fault (ms)
SLG Fault on Main Bus	70 Ω	M	19.7
	700 Ω	M	19.2
	47 k Ω	M	44.4
SLG Fault on Feeder 1	70 Ω	1	19.4
	700 Ω	1	20.4
	47 k Ω	1	60.2
SLG Fault on Feeder 2	70 Ω	2	22.9
	700 Ω	2	18.6
	47 k Ω	2	54.2
SLG Fault on Feeder 3	70 Ω	3	19.5
	700 Ω	3	21.9
	47 k Ω	3	52.2
SLG Fault on Feeder 4	70 Ω	4	19.7
	700 Ω	4	23.2
	47 k Ω	4	60.4

Again, results are similar in that load current has no effect on relay operation, despite how the load is connected. Selectivity is maintained, and as shown in Table 8, operating times all meet Class A requirements. Unfortunately, previous data has shown tripping in 25 ms for the larger faults to not always occur.

Case A-1 is repeated with the same loading but a system grounding as a HRG system- the largest fault currents are now bolted faults, although they are limited to 2 A; this is case A-2.

Table 9: Selectivity and Interrupting Time, Case A-2 (HRG System, Balanced Load)

	Fault Impedance	Tripped Feeder	t-Fault (ms)
SLG Fault on Main Bus	0 Ω	M	23.2
	700 Ω	M	21.4
	47 k Ω	M	47.4
SLG Fault on Feeder 1	0 Ω	1	20.2
	700 Ω	1	23.6
	47 k Ω	1	60.6
SLG Fault on Feeder 2	0 Ω	2	20.6
	700 Ω	2	20.2
	47 k Ω	2	61.6
SLG Fault on Feeder 3	0 Ω	3	19.2
	700 Ω	3	18.8
	47 k Ω	3	56.2
SLG Fault on Feeder 4	0 Ω	4	18.6
	700 Ω	4	21
	47 k Ω	4	62

Table 9 shows proper selectivity and operating time on par with Class A standards. Finally, the HRG system is tested with no load; this is case A-5.

Table 10: Selectivity and Interrupting Time, Case A-5 (HRG System, No Load)

	Fault Impedance	Tripped Feeder	t-Fault (ms)
SLG Fault on Main Bus	0 Ω	M	20.2
	700 Ω	M	20.6
	47 k Ω	M	59.4
SLG Fault on Feeder 1	0 Ω	1	22.6
	700 Ω	1	26
	47 k Ω	1	56
SLG Fault on Feeder 2	0 Ω	2	20.6
	700 Ω	2	24.8
	47 k Ω	2	56
SLG Fault on Feeder 3	0 Ω	3	20.6
	700 Ω	3	19.2
	47 k Ω	3	60
SLG Fault on Feeder 4	0 Ω	4	25.2
	700 Ω	4	25
	47 k Ω	4	59.2

Proper selectivity is maintained in Table 10, and all faults requiring 25 ms tripping time are tripped within 26 ms. Referring to Appendix E, nuisance tripping currents are very prevalent on a HRG system, and so proper operation of the 3-phase GFCI is confirmed.

The prototype successfully demonstrates the concept originally detailed in [9] for faults up to 10 A, and the system is capable of measuring fault currents up to the short-time thermal limits of the CTs. Although faults greater than 10 A were not tested in an environment with nuisance tripping currents present, it is assumed the relay would respond with proper selectivity and timing in a kA fault current situation; the results of this chapter indicate this is not a significant leap of faith.

Other than for the remarks regarding the energization error currents and remnant flux (which follows immediately), the prototype of the 3-phase GFCI has proved the feasibility of the concept.

Remnant Flux

The effects of remnant flux in worst-case conditions can result in nuisance tripping of the main during a faulted feeder condition. Nuisance trips are forced in lab, and a remedial degaussing circuit is proposed but not implemented.

Remnant flux can act only to decrease CT output during the first cycle or so of fault current. If all CTs were to be affected by remnant flux in the same manner, channel-to-channel matching and thus selectivity would be preserved, and only slower relay response time would be an issue. However, remnant flux only appears on CTs through which significantly large fault current is interrupted, and so over time and many fault occurrences, CTs can be left in a variety of remnant flux states. Two CTs with

varying degrees of remnant flux density will initially produce different outputs for the same current; due to the sub-cycle response required of the 3-phase GFCI, this becomes an issue. When CT outputs vary significantly during a fault (when they should otherwise match), the disparity passes through the signal chain and enters the tripping logic. If either output is diminished so much by remnant flux that it does not fall within the +/- 40% range required by the tripping logic, the main is tripped. Nuisance tripping of this type is detrimental to the 3-Phase GFCI feasibility.

In order to test worst case conditions and create an opportunity for disparity in CT output, two CTs were left with opposing senses of maximum remnant flux density prior to a fault. A 10 A DC current was applied and abruptly removed from the main and feeder 1 CTs; the wire carrying this current was fed through the CT windows in opposite directions. Furthermore, control over the fault inception angle was accomplished by varying the allowable phase window during which the fault contactor was closed. Fault inception angles were centered around 0° , 45° , and 90° , and results indicated 0° to produce the largest disparity in CT outputs, as theory would indicate. CT output for both positive and negative remnant states at a 0° and 90° fault inception angle are shown below.

Figure 66: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 4 A Fault (yellow), 0° Fault Inception Angle

Comparing the left and right CT output curves in Figure 66, CT output is significantly diminished depending on the sense of remnant flux and fault inception angle. The difference between the green or secondary voltage curves in Figure 57 turns out to create the largest disparity in filter output. Note that switching the sense of the remnant flux is equivalent to shifting the fault inception angle by 180° .

Figure 67: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 4 A Fault (yellow), 90° Fault Inception Angle

Figure 67 is identical to Figure 66, except for the 90° fault inception angle, and qualitatively shows that disparity in CT output is lessened as the fault inception angle is moved away from 0° ; further data at 45° confirmed this. Thus, 0° represents the worst-case fault inception angle for producing output disparity.

For currents smaller than 4 A, the effects of remnant flux are not as drastic, due to the smaller required minor hysteresis loop. A 293 mA fault current with the worst-case sense of remnant flux and a 0° fault inception angle proved to have no effect on relay response time; the total delay remained at 5 samples. Waveforms for a 293 mA fault are shown below in Figure 68.

Figure 68: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 293 mA Fault (yellow), 0° Fault Inception Angle

Shown in Figure 68 is the negligible difference in CT output under the same conditions that provided the large differences observed in Figure 66. This difference can be explained in that the minor hysteresis loop required for a 293 mA signal is small compared to the major hysteresis loop required for a primary rated current. Naturally, as the fault current is increased, distortion becomes more prevalent, as shown below in Figure 69 for a 1 A fault.

Figure 69: CT Output (green) for +/- (left/right) Maximum Remnant Flux Density, 1 A Fault (yellow), 0° Fault Inception Angle

CT distortion shown in Figure 69 is in between that seen for the 400 mA and 4 A faults of Figure 68 and Figure 66, as expected.

Since there are sufficient volt-seconds available in the core to produce a secondary voltage corresponding to a 293 mA current despite a maximum remnant flux condition, then these volt-seconds are always available to produce some amount of CT output. It is already confirmed the relay acts fast enough at a 293 mA fault current; at higher current levels, the secondary waveform produced by these available volt-seconds will become distorted, approaching a rectangular pulse of a smaller width than described previously for high currents in steady-state. Even this waveform is properly passed through the filter, since the impulse response of the filter is such that the relay will still respond with an immediate trip. Therefore, remnant flux has no measureable effect on relay response time.

On the other hand, selectivity is not always maintained. Imagine the following case: a bolted fault on feeder A occurs and is interrupted such that the maximum negative remnant flux “-Br” remains on the faulted feeder A and the main. Sometime later, a bolted feeder fault occurs, this time on a *different* feeder, call it B, and is interrupted such that the maximum positive remnant flux “+Br” remains on the main and feeder B. At this point, feeder A and the main share completely opposite senses of remnant flux- which has been shown to be the direst conditions for transient filter output disparity, and can thus represent the worst case scenario for remnant flux adversely affecting relay selectivity. So, for a third bolted fault on feeder A, filter outputs between feeder A and the main will track very poorly, depending on the fault inception angle.

If the fault inception angle is such that the output of the main is retarded, and feeder A is not, filter input and output will progress as follows (where data is taken as previously described):

Figure 70: Filter Output (top) and Input (bottom), 4 A Feeder 3 Fault with Main Output Diminished due to Remnant Flux

As shown in Figure 70, the main CT's output is diminished over the first half-cycle. Due to the inability of the main to produce a proportionally accurate signal, 5 samples after the threshold is passed the most recent filter output of the main is just 37% that of the faulted feeder. Thus, the values passed into the tripping logic result in an incorrect tripping of the main (the faulted feeder and the main do not match within +/- 40%).

This false trip can possibly be remedied; the only reason a feeder measurement can ever be larger than the main measurement is due to either: contributions to the fault current from the non-faulted feeders line charging currents (which don't pass through the main), or remnant flux retarding the output of the main. In both cases the fault always does occur on the feeder, and so there is no need to set an upper bound on which the largest faulted feeder must stay under with respect to the main. Feeder selectivity can be simplified to "trip the largest feeder if the feeder is greater than or equal to x% of the

main” (rather than “... is within x% of the main”). This would avoid nuisance tripping of the main when the main suffers from remnant flux.

On the other hand, if the fault inception angle is such that the output of feeder 3 is retarded, and the main is not, filter inputs and outputs will progress as follows:

Figure 71: Filter Output (top) and Input (bottom), 4 A Feeder 3 Fault with Feeder 3 Output Diminished due to Remnant Flux

In this case, when the relay makes a tripping decision, the faulted feeder is only 35% of the main, and so again results in an incorrect tripping of the main. However, it is not possible to differentiate a feeder filter output which is smaller than the main due to remnant flux from one which is only contributing a large capacitive or nuisance tripping current, especially in HRG systems. Variances in system design preclude simply choosing a percentage of the fault current measured at the main for which one can say line charging currents will always remain underneath. Since line charging currents can exceed the 35% disparity shown in Figure 71, remnant flux and nuisance currents do not have mutually exclusive ranges of magnitude.

Unfortunately, this indicates there is no simple software solution to dealing with the large disparities possible in CT outputs due to remnant flux. Remnant flux does not have any effect on relay response times, but when filter output disparity exists due to adverse combinations of remnant flux and fault inception angle, the entire bus may be tripped. In reality, this nuisance tripping was not observed during relay characterization, however it has been demonstrated as a possibility; the likelihood of such an occurrence is unknown. Only 4 A faults were tested after worst-case conditions were prepared; the disparity in filter output would become complicated to predict at fault current levels above the rated current.

In order to completely avoid such a mishap, it is recommended that a degaussing circuit be implemented. CTs should be degaussed immediately after every fault occurrence, if not periodically as well. The simplest solution would be to use an ice-cube type relay (capable of switching the 480 V line-to-line voltage at 10 A) which switches current through a positive temperature coefficient (PTC) thermistor and all windows of the CTs. The PTC thermistor is chosen with the correct resistance vs. time curve to sufficiently degauss all CTs, and the relay is controlled in a similar manner as the contactors or shunt trips. Degaussing should occur so infrequently that the low duty cycle required with thermistors is a non-issue.

CHAPTER 8

CONCLUSIONS

The basic concept of the 3-phase GFCI was successfully demonstrated, and the proof of concept prototype is ready to be tested on an operational 480 V system. With the appropriate changes in software and additional hardware, up to 15 feeders (encompassed by W-20 cores) can be accommodated. Access to a 480 V system will allow confirmation of relay operation at kA level fault currents, and further investigation of the errors associated with feeder energization will confirm the ability of the relay to operate with milliamp resolution .

As the project was funded by the patent-holders of the tripping logic, most of the design work was done with a larger end goal in sight: commercialization of a 3-phase GFCI type device. The envisioned solution comprises a W-20 like CT core embedded in a 3-phase circuit breaker with a shunt trip accessory, and a standalone digital relay. The space constraints due to implementing such a device in a typical 42-circuit lighting panel necessitates the CT cores being integrated into the breakers, however it is unknown if any breaker manufacturers are familiar with the design requirements of the digital portion of the relay, which could make it difficult to do everything in-house.

The processor, analog front end, power supplies, and output drivers would be consolidated onto a single PCB and housed in a properly shielded chassis. Combining all the components of the prototype would require a much more complicated PCB design, however it is nothing new and miniaturization in this case would not be overly difficult.

A degaussing circuit should be implemented in the final solution, for assurance that CT output during a fault is always valid. Further research could be done regarding the upper extent of SCCRs that a core which can obtain milliamp resolution is capable of handling, with regards to short-time thermal ratings. It is likely that an optimal core would be designed for this specific application.

A large amount of variability in the measured tripping times can be attributed to the contactors mechanical action. Accounting for asynchronous sampling, the prototype always responded with a correct tripping decision in under 6.5 ms; all the remaining time was allotted to waiting for the contactors to break the fault current. The speed of shunt-trips on 3-phase circuit breakers needs to be investigated. If interruption within 1 cycle can be guaranteed, Class A timing requirements will not be difficult to meet; perhaps even the 20 ms suggested by UL 943C is possible.

As a final note, the amount of release delay (implemented so as to avoid EMI) may have exceeded the desired half-cycle, lengthening the total time taken for a contactor to open: the 22.5° phase window may have been too constrained, due to finite errors in sampling and quantization of the reference phasor. It is unknown whether this is responsible for some of the greater than 25 ms tripping times, since release delay was only measured initially over about 40 samples. It is however deemed unlikely.

BIBLIOGRAPHY

- [1] C.F. Dalziel, "Effects of Electric Shock on Man," *IRE Transactions on Medical Electronics*, vol. PGME-5, pp. 44-62, July 1956.
- [2] E. W. Roberts, *Overcurrents and Undercurrents: All about GFCIs, AFCIs, and Similar Devices*. Mystic, CT: Reptec, 2000.
- [3] W.B. Kouwenhoven; W.R. Milnor, "Field Treatment of Electric Shock Cases-I," *Power Apparatus and Systems, Part III. Transactions of the American Institute of Electrical Engineers*, vol. 76, no. 3, pp. 82-84, April 1957.
- [4] R.E. Nabours, "Dalziel revisited," *IEEE Industry Applications Magazine*, vol.15, no. 3, pp. 18-21, May-June 2009.
- [5] *Effects of current on human beings and livestock - Part 1: General aspects*, IEC/TS 60479-1, 2005.
- [6] J. Sottile; S.J. Gnappagasam; T. Novak; J.L. Kohler, "Detrimental Effects of Capacitance on High-Resistance-Grounded Mine Distribution Systems," *IEEE Transactions on Industry Applications*, vol. 42, no. 5, pp. 1333-1339, Sept.-Oct. 2006.
- [7] P. S. Hamer. (2010, Aug. 6) *IEEE PCIC-2008-02 GFCIS-3Ph-Rev2 with Notes* [Online]. Available e-mail: PaulHamer@Chevron.com Message: FW: Advice to Intern - Three-phase GFCI background
- [8] Powell Electrical Group. (2011, Aug 14). *High Resistance Grounding Units* [Online]. Available: http://www.powellind.com/main/Uploadpdf/12037_hrg_85x11_v0.pdf
- [9] P.S. Hamer, "The Three-Phase Ground-Fault Circuit-Interrupter System - a novel approach to prevent electrocution," *Petroleum and Chemical Industry Technical Conference, 2008. 55th IEEE PCIC*. pp.1-13, 22-24 Sept. 2008
- [10] P. S. Hamer, "Ground-fault circuit-interrupter system for three-phase electrical power systems," U.S. Patent 7 301 739, Nov. 27, 2007.
- [11] R.G. Bruce; A. Wright, "Remanent flux in current-transformer cores," *Proceedings of the Institution of Electrical Engineers*, vol. 113, no. 5, pp. 915-920, May 1966.
- [12] S. E. Zocholl; D. W. Smaha. (1992). *Current Transformer Concepts* [Online]. Available: <http://www.selinc.com/WorkArea/DownloadAsset.aspx?id=2436>

- [13] L. Underwood (2005). *Current Transformer Selection Criteria for Relays With Adaptive Overcurrent Elements* [Online]. Available:
<http://www.selinc.com.cn/ResourceFolder/Articlefiles/2008-5/1334316.PDF>
- [14] S. E. Zocholl; G. Benmouyal. *How Microprocessor Relays Respond to Harmonics, Saturation, and Other Wave Distortions* [Online]. Available:
www2.selinc.com/techpprs/6059.pdf
- [15] E.O. Schweitzer, III; D. Hou, "Filtering for protective relays," *WESCANEX 93. 'Communications, Computers and Power in the Modern Environment.'* Conference Proceedings. pp.15-23, 1993
- [16] M. Savostianik. (2000, May). *Lowering the Limits for Earth-Fault Detection* [Online]. Available:
<http://www.startco.ca/library/papers/Lowering%20the%20Limits.pdf>
- [17] S. D. Moreton, "A Simple Method for the Determination of Bushing-Current-Transformer Characteristics," *Transactions of the American Institute of Electrical Engineers*, vol. 62, no. 9, pp. 581-585, Sept. 1943.

APPENDICES

Appendix A - EMTP vs. V-HARM Verification Simulations

All values in this appendix are rounded to the nearest mA, or .1 A if in kA, unless close to V-HARM and otherwise rounded down. Nuisance tripping currents are highlighted in yellow, undetectable fault currents in green.

A-1, Solidly Grounded System, Balanced Load

TABLE A-1	Fault Impedance	Main			Feeder 1		
		EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
SLG Fault on Main Bus	0Ω	10 kA	10 kA	-	2	2	0
	700Ω	382	384	2	0	0	0
	46,000Ω	6	4	2	0	0	0
SLG Fault on Feeder 1	0Ω	2.4 kA	2.4 kA	-	2.4 kA	2.4 kA	-
	700Ω	372	375	3	372	375	3
	46,000Ω	6	6	0	6	6	0
SLG Fault on Feeder 2	0Ω	1.3 kA	1.3 kA	-	0	1	1
	700Ω	361	366	5	0	0	0
	46,000Ω	6	6	0	0	1	1
SLG Fault on Feeder 3	0Ω	.63 kA	.65 kA	-	0	1	1
	700Ω	336	348	12	0	0	0
	46,000Ω	5	5	0	0	0	0
SLG Fault on Feeder 4	0Ω	2.35 kA	2.4 kA	-	1	1	0
	700Ω	372	372	0	0	1	1
	46,000Ω	6	4	2	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
4	4	0	8	8	0	105	108	3
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1
1	1	0	2	2	0	25	25	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1
1.3 kA	1.3 kA	-	1	1	0	13	14	1
361	366	5	0	0	0	0	1	1
6	6	0	0	0	0	0	1	1
0	1	1	.63 kA	.65 kA	-	7	7	0
0	0	0	336	349	13	0	1	1
0	0	0	5	5	0	0	0	0
1	1	0	2	2	0	2.35 kA	2.4 kA	-
0	0	0	0	0	0	372	375	3
0	0	0	0	0	0	6	5	1

A-2, HRG System, Balanced Load

TABLE A-2	Fault Impedance	Main			Feeder 1		
		EMT P	V-HARM	Abs Diff	EMT P	V-HARM	Abs Diff
SLG Fault on Main Bus	0Ω	1940	1981	-	6	6	0
	700Ω	316	323	7	1	1	0
	46,000Ω	6	6	0	0	1	1
SLG Fault on Feeder 1	0Ω	1885	1899	-	1915	1931	-
	700Ω	307	309	2	312	315	3
	46,000Ω	6	5	1	6	6	0
SLG Fault on Feeder 2	0Ω	1827	1854	-	6	6	0
	700Ω	309	302	7	1	2	1
	46,000Ω	5	6	1	0	0	0
SLG Fault on Feeder 3	0Ω	1701	1770	-	5	6	1
	700Ω	278	288	10	1	1	0
	46,000Ω	5	5	0	0	0	0
SLG Fault on Feeder 4	0Ω	1885	1901	-	6	6	0
	700Ω	307	308	1	1	1	0
	46,000Ω	6	4	2	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMT P	V-HARM	Abs Diff	EMT P	V-HARM	Abs Diff	EMT P	V-HARM	Abs Diff
12	12	0	24	24	0	309	311	2
2	2	0	4	4	0	50	50	0
0	0	0	0	0	0	1	2	1
12	11	1	24	24	0	300	303	3
2	2	0	4	4	0	49	50	1
0	0	0	0	0	0	1	1	0
1855	1882	-	23	23	0	291	295	4
302	307	5	4	3	1	47	48	1
6	5	1	0	0	0	1	1	0
11	11	0	1725	1795	-	271	282	11
2	2	0	281	293	12	44	47	3
0	0	0	5	5	0	1	1	0
12	11	1	24	24	0	1885	1903	-
2	2	0	4	4	0	307	309	2
0	0	0	0	0	0	6	5	1

A-3, Ungrounded System, Balanced Load

TABLE A-3	Fault Impedance	Main			Feeder 1		
		EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
SLG Fault on Main Bus	0Ω	0	359	-	6	6	0
	700Ω	0	263	263	5	5	0
	46,000Ω	0	6	6	0	0	0
SLG Fault on Feeder 1	0Ω	0	2	-	336	345	-
	700Ω	0	2	2	247	252	5
	46,000Ω	0	1	1	6	5	1
SLG Fault on Feeder 2	0Ω	0	1	-	6	6	0
	700Ω	0	2	2	4	5	1
	46,000Ω	0	1	1	0	0	0
SLG Fault on Feeder 3	0Ω	0	1	-	5	5	0
	700Ω	0	1	1	4	4	0
	46,000Ω	0	0	0	0	0	0
SLG Fault on Feeder 4	0Ω	0	1	-	6	6	0
	700Ω	0	0	0	4	5	1
	46,000Ω	0	1	1	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
12	11	1	24	24	0	309	310	1
9	8	1	18	18	0	227	227	0
0	0	0	0	0	0	5	5	0
12	11	1	24	24	0	300	304	4
7	8	1	17	18	1	221	223	2
0	0	0	0	0	0	5	5	0
320	332	-	23	24	1	291	296	5
236	242	6	17	17	0	215	217	2
5	5	0	0	0	0	5	5	0
11	10	1	287	303	-	272	283	11
8	8	0	212	222	10	200	206	6
0	0	0	5	5	0	5	3	2
12	11	1	24	24	0	41	46	-
9	9	0	17	17	0	30	34	4
0	1	1	0	0	0	1	2	1

A-4, Solidly Grounded System, No Load

TABLE A-4	Fault Impedance	Main			Feeder 1		
		EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
SLG Fault on Main Bus	0Ω	10.1 kA	10 kA	-	2	3	1
	700Ω	396	396	0	0	0	0
	46,000Ω	6	6	0	0	1	1
SLG Fault on Feeder 1	0Ω	2.46 kA	2.3 kA	-	2.46 kA	2.4 kA	-
	700Ω	396	393	3	396	396	0
	46,000Ω	6	6	0	6	6	0
SLG Fault on Feeder 2	0Ω	1.35 kA	1.3 kA	-	0	1	1
	700Ω	396	396	0	0	0	0
	46,000Ω	6	6	0	0	0	0
SLG Fault on Feeder 3	0Ω	.7 kA	.7 kA	-	0	1	1
	700Ω	396	395	1	0	0	0
	46,000Ω	6	6	0	0	1	1
SLG Fault on Feeder 4	0Ω	2.46 kA	2.4 kA	-	1	1	0
	700Ω	396	396	0	0	1	1
	46,000Ω	6	7	1	0	1	1

Feeder 2			Feeder 3			Feeder 4		
EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
4	4	0	8	9	1	107	109	2
0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	1	1
1	1	0	2	2	0	26	26	0
0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0
1.35 kA	1.3 kA	-	1	1	0	14	14	0
396	396	0	0	0	0	0	0	0
6	6	0	0	0	0	0	1	1
0	1	1	.7 kA	.7 kA	-	7	8	1
0	0	0	396	396	0	0	0	0
0	0	0	6	6	0	0	1	1
1	1	0	2	2	0	2.46 kA	2.4 kA	-
0	0	0	0	0	0	396	396	0
0	0	0	0	0	0	6	6	0

A-5, HRG System, No Load

TABLE A-5	Fault Impedance	Main			Feeder 1		
		EMT P	V-HARM	Abs Diff	EMT P	V-HARM	Abs Diff
SLG Fault on Main Bus	0Ω	2008	2035	-	6	7	1
	700Ω	327	332	5	1	2	1
	46,000Ω	6	6	0	0	1	1
SLG Fault on Feeder 1	0Ω	2007	2001	-	2039	2038	-
	700Ω	327	326	1	332	332	0
	46,000Ω	6	7	1	6	7	1
SLG Fault on Feeder 2	0Ω	2006	2005	-	6	7	1
	700Ω	327	328	1	1	2	1
	46,000Ω	6	6	0	0	0	0
SLG Fault on Feeder 3	0Ω	2003	2002	-	6	7	1
	700Ω	327	328	1	1	2	1
	46,000Ω	6	7	1	0	1	1
SLG Fault on Feeder 4	0Ω	2007	2005	-	6	7	1
	700Ω	327	326	1	1	1	0
	46,000Ω	6	7	1	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMT P	V-HARM	Abs Diff	EMT P	V-HARM	Abs Diff	EMT P	V-HARM	Abs Diff
13	12	1	25	25	0	320	320	0
2	2	0	4	4	0	52	52	0
0	0	0	0	0	0	1	1	0
13	12	1	25	25	0	310	319	9
2	2	0	4	4	0	52	51	1
0	0	0	0	0	0	1	1	0
2036	2032	-	25	25	0	319	319	0
332	332	0	4	4	0	52	52	0
6	6	0	0	0	0	1	1	0
13	12	1	2032	2027	-	319	319	0
2	2	0	332	331	1	52	52	0
0	0	0	6	6	0	1	1	0
13	12	1	25	25	0	2008	2005	-
2	2	0	4	4	0	327	327	0
0	0	0	0	0	0	6	6	0

A-6, Ungrounded System, No Load

TABLE A-6	Fault Impedance	Main			Feeder 1		
		EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
SLG Fault on Main Bus	0Ω	0	366	-	6	8	2
	700Ω	0	270	270	5	5	0
	46,000Ω	0	6	6	0	1	1
SLG Fault on Feeder 1	0Ω	0	3	-	357	362	-
	700Ω	0	1	1	263	265	2
	46,000Ω	0	1	1	6	7	1
SLG Fault on Feeder 2	0Ω	0	1	-	6	7	1
	700Ω	0	1	1	5	5	0
	46,000Ω	0	1	1	0	0	0
SLG Fault on Feeder 3	0Ω	0	1	-	6	7	1
	700Ω	0	2	2	5	5	0
	46,000Ω	0	1	1	0	0	0
SLG Fault on Feeder 4	0Ω	0	1	-	6	7	1
	700Ω	0	1	1	5	4	1
	46,000Ω	0	1	1	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
13	12	1	25	25	0	320	320	0
9	9	0	18	18	0	235	236	1
0	0	0	0	1	1	6	7	1
13	12	1	25	25	0	320	320	0
9	9	0	19	19	0	235	234	1
0	0	0	0	1	1	5	5	0
351	354	-	25	25	0	320	319	1
259	261	2	19	19	0	235	237	2
6	6	0	0	1	1	5	6	1
13	12	1	339	341	-	320	320	0
9	9	0	249	252	3	235	237	2
0	0	0	6	6	0	5	6	1
13	12	1	25	25	0	44	46	-
9	9	0	19	19	0	32	34	2
0	0	0	0	1	1	1	1	0

A-7, Solidly Grounded System, Unbalanced Load (Feeder 4 Disconnected)

TABLE A-7	Fault Impedance	Main			Feeder 1		
		EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
SLG Fault on Main Bus	0Ω	10.1 kA	10 kA	-	2	3	1
	700Ω	394	389	5	0	1	1
	46,000Ω	6	6	0	0	0	0
SLG Fault on Feeder 1	0Ω	2.5 kA	2.5 kA	-	2.5 kA	2.5 kA	-
	700Ω	390	380	10	390	378	12
	46,000Ω	6	6	0	6	6	0
SLG Fault on Feeder 2	0Ω	1.4 kA	1.4 kA	-	0	1	1
	700Ω	393	387	6	0	0	0
	46,000Ω	6	7	1	0	0	0
SLG Fault on Feeder 3	0Ω	.72 kA	.75 kA	-	0	1	1
	700Ω	385	369	16	0	1	1
	46,000Ω	6	7	1	0	0	0
SLG Fault on Feeder 4	0Ω	-	-	-	-	-	-
	700Ω	-	-	-	-	-	-
	46,000Ω	-	-	-	-	-	-

Feeder 2			Feeder 3			Feeder 4		
EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff	EMTP	V-HARM	Abs Diff
4	4	0	8	8	0	-	-	-
0	1	1	0	1	1	-	-	-
0	1	1	0	1	1	-	-	-
1	1	0	2	2	0	-	-	-
0	1	1	0	1	1	-	-	-
0	1	1	0	0	0	-	-	-
1.4 kA	1.4 kA	-	1	1	0	-	-	-
393	385	8	0	0	0	-	-	-
6	6	0	0	0	0	-	-	-
0	1	1	.72 kA	.75 kA	-	-	-	-
0	1	1	385	366	19	-	-	-
0	1	1	6	6	0	-	-	-
-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-

Appendix B - Ungrounded System Tripping Logic Solution

If directionality or phase of the measured currents during a fault on an ungrounded system are known, the phase (rather than the magnitude) can be used to determine the faulted feeder. Non-faulted feeders will contribute currents in the opposite direction of actual fault current; the phases of the faulted and non-faulted feeder will be 180 degrees out of phase. During a bus fault, all measured waveforms will be in-phase, and during a feeder fault, the faulted feeder will be 180 degrees out of phase with the non-faulted feeder and main. Since no resistive ground current flows, phases of measured currents will always be either in phase or 180 degrees out of phase; in HRG and solidly grounded systems, this is not guaranteed, and so use of phase information would be more difficult.

Figure 72: EMTP Schematic Corresponding an Ungrounded System

The circuit of Figure 72 is simulated with a 700 Ω SLG fault on feeder 1 and the main bus to examine the instantaneous zero sequence current waveforms as measured by the CBCTs.

Figure 73: Instantaneous Zero Sequence Current Sensed by CBCTs, 700 Ω SLG fault on Feeder 1

Figure 73 indicates opposite polarity of the current in the faulted feeder (feeder 1; red), indicating a fault on feeder 1.

Figure 74: Instantaneous Zero Sequence Current Sensed by CBCTs, 700 Ω SLG fault on Main Bus

For a fault on the bus, Figure 74 shows all feeder currents in-phase; this only occurs for bus faults. Therefore, knowing the phase of all waveforms in an ungrounded system is sufficient knowledge in and of itself to determine the faulted feeder.

Due to the need for tripping decisions to be made prior to filter settling, transient phase buildup should be studied and verified as acceptable. Phase information can be extracted in a similar manner as used in the phase-restricted contactor opening controls. The best solution would require combining both magnitude and phase information, e.g. using phase information only once the magnitude has surpassed the tripping threshold.

Appendix C - CT Saturation Characteristics

This appendix details how a CT without over-voltage protection would act in cases of extreme saturation.

CT Behavior in Extreme Saturation

The following is based off [12] in which the CT is modeled as a volt-time switch. This model simplifies the CT to transitioning back and forth between positive and negative saturation, so that the maximum slope occurs between transition states, and the magnetizing current can be ignored. The equivalent circuit is shown below.

Figure 75: Simplified CT Schematic for Volt-Second Preservation Analysis [12]

As shown in Figure 75, the switch is open during the transition period, so that all the secondary current flows to the burden, and off when saturated, so that all the secondary current flows into the magnetizing path.

For a given core and burden, a minimum volt-time area is established during which the CT will faithfully reproduce an output signal over a half-cycle,

$$A_{min} = \int_0^{T/2} V_{S,peak-em.sat.} \sin(\omega t) dt$$

Where the secondary peak voltage just prior or eminent to saturation is defined as,

$$V_{S,peak-em.sat.} = \frac{I_{P,peak-rated}}{N} \cdot Z_B = I_{S,peak-rated} \cdot Z_B$$

In other words, the eminent saturation voltage is the highest peak voltage where a sinusoid is maintained on the secondary. It is a function of core characteristics and burden impedance. From this definition of a minimum volt-second area, it is possible to determine the length of time an output will be maintained on the secondary burden under a fault current many times the rating of the CT.

$$\int_0^{t_{sat}} V_{s,peak-fault} \sin(\omega t) dt = A_{min}$$

$$V_{s,peak-fault} = M \cdot V_{s,peak-em.sat.}$$

$$M = \frac{I_{Fault}}{I_{P,rated}}$$

$$V_{s,peak-sat} = V_{s,peak-fault} \sin(\omega t_{sat})$$

A few notes before proceeding:

- The above is only valid for $M > 2$, since this guarantees saturation of the core will occur prior to $T/4$, so that the secondary voltage is always increasing.
- $V_{s,peak-fault}$ is the peak voltage the secondary would reach as a result of a highly saturating fault current were there no maximum volt-time area to prevent the signal from reaching this peak value (i.e. an ideal CT). Its value is in defining the curve the secondary voltage follows prior to saturation.
- $V_{s,peak-sat}$ is the peak secondary voltage reached prior to saturation acting to reduce the output to zero. It is the maximum secondary voltage.

Since the saturation status threshold is determined at a minimum secondary voltage magnitude, the time interval of interest is the difference between the loss of secondary output due to saturation, and the occurrence of this minimum threshold. This is the amount of time available to measure and therefore preserve the saturation status.

$$t_{op} = t_{sat} - t_{th}$$

The threshold voltage V_{th} is chosen according to the desired pickup level and defined via

$$V_{th} = V_{S,peak-fault} \sin(\omega t_{th})$$

Simplifying the above equations to determine the minimum operating time results in

$$t_{op} = \frac{1}{\omega} \left\{ \cos^{-1} \left(\frac{M-2}{M} \right) - \sin^{-1} \left(\frac{V_{th}}{M \cdot V_{S,peak-em.sat.}} \right) \right\}$$

The operating time is dependent on the magnitude of the fault current and the threshold voltage relative to the eminent saturation voltage. The effects of these parameters are shown in Figure 76 below.

Figure 76: Operating Time for CT during Large Fault ($f = 60$ Hz)

Figure 76 shows calculated operating times during a heavy fault. At 10,000x rated current the minimum operating time is greater than 50 μ s, and since the region near 10,000x sets an outer limit at 480 V, it can be concluded variations in the relative threshold voltage do not have an effect on the minimum operating time. Most importantly, 50 μ s is ample time for detection using high-speed comparators configured in an over-voltage detector with outputs connected to an interrupt service routine on a μ -controller. Further research could confirm the times suggested in Figure 76 by investigating how the short-time thermal ratings of the CT are derived. The above results make sense, as the datasheet specified 2.4kA/s, 6kA/40ms ratings must come from some secondary current flow during heavily saturated conditions. In practice however, CT datasheet thermal ratings may be empirically derived. This approach is somewhat conservative as volt-time area should increase slightly under very large faults due to continued coupling through the air.

Appendix D - W-20 Characterization Data

The W-20 CT core is characterized for measurements from 1mA to a 1A and shown to be acceptably linear.

Figure 77: W-20 Core and Winding

Linearity of the Core

The linearity of the W-20 is measured at various burdens over the range .001-1 A RMS primary current. Measurements are made on the U3402A DMM in Fast Mode, measuring AC coupled signals only (RMS measurements with an oscilloscope may vary up to 15%). Two different sources are used to generate primary current, the Agilent 33210A Function Generator (FG), and the California Instruments 850 Precision Oscillator.

Figure 78: Linearity of W-20 for 179, 427, and 978 Ω Burdens with FG Source, 4-30 mA Input

Examination of the linear regressions in Figure 78 shows strong linearity (near unity R^2 values), and less than 150 μA of offset for the three burdens over currents from 4-30 mA. As desired, this indicates minimal ratio error due to excitation current at the tripping threshold of 5 mA. If excitation current can be neglected, the CT will then be linear up until saturation. The transfer function of the CT, in mV/mA, is determined empirically from the slopes of the regressions in Figure 78, and can alternately be derived as:

$$G = \frac{V_{out}}{I_{in}} = \frac{V_s}{I_p} = \frac{V_s N}{I_s} = R_{burden} N \quad (4.3)$$

However, over-burdening the CT invalidates this model. A high-pass filter (HPF) formed by the self inductance of the windings and burden resistance attenuates the desired 60 Hz signal when the burden is large enough. This effect is accounted for when measuring the overall gain of the CT, as in Figure 78, which includes the effective turn's

ratio, and any other errors in the theoretical gain. The more accurate empirically determined turns ratios are used in signal chain calibration.

Table 11: Turns Ratio Correction

R-burden (Ω)	G (mV/mA)	Offset (mV)	N_{eff}
179	0.308	0.131	581
427	0.686	0.005	622
978	1.412	0.123	663/693

Table 11 shows the effective turn ratios of the W-20, rounded to the nearest integer. These values agree with the datasheet stated 598 turns, and increase with the burden to express filter attenuation at 60 Hz. The reason for two N_{eff} values with the 978 Ω burden are explained further on.

The high-pass filter phenomenon is confirmed through a rudimentary LTSpice CT model utilizing a Chan core with B-H characteristics defined at 3.4 A RMS primary current. Burdens of 180, 427, and 1000 Ω are input across the CT with the following frequency response.

Figure 79: LTSpice CT Frequency Response (magnitude is solid) for 180 Ω (pink), 427 Ω (yellow), and 1000 Ω (blue) Burdens

In order to examine errors in the transfer function as a function of burden, Figure 79 shows the secondary current for various burdens (note this normalizes the secondary voltage response maximum value; corner frequencies are of the only interest here). As shown in Figure 79, there is minimal signal attenuation at 60 Hz for a 180 Ω burden, however the 427 Ω and 1000 Ω burdens significantly shift the corner frequency, decreasing the magnitude response at 60 Hz. For purposes of resolving a larger secondary voltage at a given input current, this increase in effective turns ratio is overshadowed by the increase in the secondary voltage, and harmonics passing through at a larger magnitude (than fundamental) is a non-issue, due to the digital filter.

With the precision oscillator sourcing the primary current, CT characteristics are observed for a 978 Ω burden. A large burden is chosen to increase the small current secondary voltage measurements, and to relax the amount of primary current needed to saturate the core.

Figure 80: CT Gain, FG vs. Precision Oscillator Source, 1-30 mA Input

As shown in Figure 80, the gain of the CT is different for each source; this suggests a conflict between the sources, not with the CT itself, since measurement tools and procedures were not otherwise changed. The reason for this discrepancy is unknown - perhaps the FG signal includes high frequency switching, which slightly saturates the core. The error may also lie in retaining auto-scaling of primary current RMS measurements in the U3402A. The trends still indicate linearity with low offset, and suggest an effective turn ratio somewhere between the two values shown in Table 11.

The primary current is then extended to 1 A RMS.

Figure 81: CT Gain, Precision Oscillator, .001-1 A Input

Figure 81 indicates high linearity of the CT through 1A. Calculated slope increases only 0.2% from that of Figure 80. Offset calculated from the linear regression is 1.5 mA, however a scalar transfer function based on the slope at 5 mA will be implemented in software, reducing error from offset to the 150 μ A levels seen in Figure 78. Effects of this at higher current measurements will be filtered out as common mode error by the tripping logic, and are negligible since absolute certainty is only required at the tripping threshold.

The effective turn ratio must be known in order to characterize the CT; see results of Table 11. Since only the 60 Hz component will be recovered, it is of no consequence what the frequency dependent transfer function of the CT is, except at 60 Hz. Even then, this is best determined empirically when the entire signal chain is complete, for which the above analysis can give an expected result to aid in calibration.

Saturation Current and Burden

Saturation current is estimated by triggering the scope on the zero-crossing of the secondary voltage, and noting when the following peak of the output waveform begins drifting away from the $T/4 = 4.17$ ms expected for a 60 Hz sinusoid. Primary currents are generated by a 120 V variac and 3x 25 W power resistors.

Table 12: W-20 Saturation Current for Various Burdens

Burden (Ω)	Sat. Current (A-rms)
180	5.7+
300	5.7+
400	5.3
510	4.1
750	2.6
1000	2

The data of Table 12 gives a rough idea of measureable current ranges for various burdens, but is not very accurate.

Excitation Curve

In order to better approximate saturation currents for various burdens, the excitation curve of the CT was characterized using the California Instruments Series 850T Precision Oscillator as a source, and either 2x U3402A DMMs or an Amprobe 5XP-A and DS03202A oscilloscope to measure secondary RMS current and voltage, respectively. In both cases, voltage measurements were AC coupled, and current measurements were AC+DC coupled.

Figure 82: W-20 Excitation Curve

The curves of Figure 82 agree for the upper portion of the curve (past the “knee”), but digress below, when current is in the sub-mA region and increasingly difficult to accurately measure. Measured current is actually magnetizing current, and so is erratic in nature with a high crest factor; this coupled with μA measurements can explain differences and errors in the measurements. It is assumed the U3402A is more accurate- the Amprobe is only a handheld multi-meter.

Data from Figure 82 could be used to fit a curve such that the procedure outlined in [17] could be algorithmically run to produce a current dependent transfer function:

- A secondary voltage can be assumed, a burden chosen, and the secondary current calculated via Ohm’s Law.
- This is then added in quadrature with the excitation current found from the graph, and reflected to the primary via the ideal turn ratio.

- The effective turn ratio is finally found via the ratio of this primary current with the secondary current calculated before the excitation current was added to it.

This approach would be useful in analyzing operation outside of the core's linear region, where a fixed gain is assumed. Unfortunately, due to the logarithmic nature of the data, it is not an easy task to fit a curve.

MATLAB can however easily interpolate between the given data points using the `splinetx` function, so for currents falling within the range measured in Figure 82 a ratio correct factor curve [17] can be plotted.

Figure 83: Interpolated Excitation Curve for U3402A Data in Figure 30

The interpolated data shown in Figure 83 is used to analyze the expected CT response via the ratio correction factor at the burdens and effective turns ratios previously tested and shown in Table 12.

Figure 84: RCF for a) 179 Ω Burden, 582 N-eff, b) 427 Ω Burden, 622 N-eff, c) 980 Ω Burden, 678 N-eff

The ratio correction factor data of Figure 84 show non-linearity at the large primary currents in good agreement with the saturation values measured by sight and described in Table 12. Unfortunately, the excitation curve data is still not extrapolated to extreme CT conditions; it is only interpolated. Even then, this range is limited due to inability to measure low secondary currents (as described previously), and inability to measure larger secondary currents, so as to avoid over-heating of the 28 AWG secondary magnetic wire.

In theory it should be straight forward to manually implement data points for the higher currents, since this is where the CT is saturated and relative permeability is unity; the CT is then linear. MATLAB can then interpolate between manually chosen largely spaced out linear points (large enough to accommodate the maximum desired primary input) as necessary. This may give insight into CT behavior at very large currents, from a theoretical perspective. It may be best to prove the concept empirically.

Modeling of a Core

Can a CT be satisfactorily characterized by a traditional voltage transformer core model approach? If so, core characteristics can be modeled in simulation, allowing examination of properties under extreme saturation.

LTSpice is able to implement a Chan inductor to model a non-linear core. The Chan inductor requires three magnetic parameters, from which it fits the major B-H curve:

- Remnant flux density (B_r)
- Coercive force (H_c)
- Saturated flux density (B_s)

And three geometric parameters to interface with the electric circuit quantities:

- Turns ratio (N)
- Core cross-sectional area (A_c)
- Mean magnetic length (l_m)
- Length of air gap (l_g)

However, a CT has no major hysteresis loop; CTs are driven from constant current sources, not constant voltage sources. This becomes evident in that while attempting to establish a major hysteresis loop, H_c is observed as a function of primary current. Thus, H_c increases as the peak primary current increases, without limit. For this reason, CTs are not normally characterized by a B-H curve, but rather by an excitation curve.

Nevertheless, a major hysteresis loop was analyzed such that a qualitative CT model could be implemented in LTSpice. This was achieved using the 240 VAC bench source, a $70\ \Omega$ current limiting resistor, a current probe amplifier to measure primary current, and a 10:1 scope probe to measure integrated voltage via a $297\ \text{k}\Omega$ $2.28\ \mu\text{F}$ passive integrator.

Figure 85: W-20 B-H Curve for 3.4 A-rms Input

Figure 85 shows the B-H loop for a 3.4 A RMS primary input current which was used to determine a major hysteresis loop for the Chan core inductor; this allows a rudimentary LTSpice CT model. If it is possible to determine and implement the relationship between H_c and I_p in LTSpice, an accurate simulation model can be created.

Figure 86: Chan Core Based LTSpice W-20 Model

Figure 86 shows the defunct LTSpice CT model schematic. In addition to modeling only a single major hysteresis curve, the model does not account for excitation current. At the very least, simulation shows a large, ~80%, remnant flux. EMTP CBCT measurements can be directly input into LTSpice as a piecewise linear text file; this would prove a powerful tool if the CT model were valid.

Appendix E - EMTP Simulations vs. Test System Measurements

All values in this appendix are rounded to the nearest mA. Effects of load current were not accounted for in these measurements; as such only tables A-4, A-5, and A-6 are present. Only measured nuisance tripping currents are highlighted in yellow. Undetectable fault currents are highlighted in green.

A-4, Solidly Grounded System, No Load

TABLE A-4	Fault Impedance	Main			Feeder 1		
		EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff
SLG Fault on Main Bus	27.72Ω	9747	5412	-	0	0	0
	700Ω	396	392	4	0	0	0
	47,000Ω	6	7	1	0	0	0
SLG Fault on Feeder 1	27.72Ω	9711	5489	-	9708	5501	-
	700Ω	396	389	7	396	385	11
	47,000Ω	6	6	0	6	6	0
SLG Fault on Feeder 2	27.72Ω	9675	5463	-	0	0	0
	700Ω	396	388	8	0	0	0
	47,000Ω	6	6	0	0	0	0
SLG Fault on Feeder 3	27.72Ω	9601	5540	-	0	0	0
	700Ω	396	390	6	0	0	0
	47,000Ω	6	6	0	0	0	0
SLG Fault on Feeder 4	27.72Ω	9711	5363	-	0	0	0
	700Ω	396	390	6	0	0	0
	47,000Ω	6	6	0	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff
0	0	0	0	0	0	4	3	1
0	0	0	0	0	0	0	2	2
0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	4	2	2
0	0	0	0	0	0	0	2	2
0	0	0	0	0	0	0	1	1
9672	5478	-	0	0	0	4	2	2
396	389	7	0	0	0	0	1	1
6	6	0	0	0	0	0	1	1
0	0	0	9598	5538	-	4	2	2
0	0	0	396	390	395	0	2	2
0	0	0	6	6	0	0	1	1
0	0	0	0	0	0	9711	5389	-
0	0	0	0	0	0	396	391	5
0	0	0	0	0	0	6	7	1

* CTs can only measure up to ~4A, so the 10A measurement is invalid as the ADC clips

A-5, HRG System, No Load

TABLE A-5	Fault Impedance	Main			Feeder 1		
		EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff
SLG Fault on Main Bus	0 Ω	2000	1956	44	7	8	1
	700 Ω	327	321	6	1	1	0
	47,000 Ω	6	7	1	0	0	0
SLG Fault on Feeder 1	0 Ω	1998	1955	43	2030	1997	33
	700 Ω	327	321	6	332	326	6
	47,000 Ω	6	6	0	6	6	0
SLG Fault on Feeder 2	0 Ω	1997	1954	43	7	8	1
	700 Ω	327	322	5	1	1	0
	47,000 Ω	6	6	0	0	0	0
SLG Fault on Feeder 3	0 Ω	1994	1953	41	7	8	1
	700 Ω	327	323	4	1	1	0
	47,000 Ω	6	7	1	0	0	0
SLG Fault on Feeder 4	0 Ω	1998	1956	42	7	8	1
	700 Ω	327	320	7	1	1	0
	47,000 Ω	6	7	1	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff
14	13	1	28	26	2	320	339	19
2	2	0	5	4	0	52	55	3
0	0	0	0	0	0	1	1	0
14	13	1	28	25	2	319	330	10
2	2	0	5	4	0	52	54	2
0	0	0	0	0	0	1	1	0
2027	1984	43	28	25	2	319	306	13
332	328	4	5	4	0	52	53	1
6	6	0	0	0	0	1	1	0
14	13	1	2022	1993	29	319	323	5
2	2	0	331	327	4	52	49	3
0	0	0	6	6	0	1	2	1
14	13	1	28	25	2	1999	1958	41
2	2	0	5	4	0	327	321	6
0	0	0	0	0	0	6	7	1

A-6, Ungrounded System, No Load

TABLE A-6	Fault Impedance	Main			Feeder 1		
		EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff
SLG Fault on Main Bus	0Ω	0	1	1	7	8	1
	700Ω	0	1	1	5	6	1
	47,000Ω	0	0	0	0	0	0
SLG Fault on Feeder 1	0Ω	0	1	0	362	370	8
	700Ω	0	1	1	265	256	9
	47,000Ω	0	0	0	6	6	0
SLG Fault on Feeder 2	0Ω	0	1	0	7	8	1
	700Ω	0	1	1	5	6	1
	47,000Ω	0	0	0	0	0	0
SLG Fault on Feeder 3	0Ω	0	1	0	7	8	1
	700Ω	0	1	1	5	6	1
	47,000Ω	0	0	0	0	0	0
SLG Fault on Feeder 4	0Ω	0	1	0	7	8	1
	700Ω	0	1	1	5	6	1
	47,000Ω	0	0	0	0	0	0

Feeder 2			Feeder 3			Feeder 4		
EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff	EMTP	Meas.	Abs Diff
14	13	1	28	26	2	321	321	0
10	10	0	20	19	1	235	235	0
0	0	0	0	1	0	5	5	1
14	13	1	28	26	2	321	333	12
10	10	0	20	19	1	235	229	6
0	0	0	1	0	0	5	5	0
355	373	18	28	26	2	321	338	18
260	249	11	20	19	1	235	224	11
6	5	0	0	1	0	5	4	1
14	13	1	341	325	16	321	303	18
10	10	0	250	238	11	235	222	13
0	0	0	6	5	0	5	5	1
14	13	1	28	25	3	48	48	0
10	10	0	20	19	1	35	36	1
0	1	0	0	1	1	1	2	2

Appendix F - Prototype C Code

```

/*****
 * 3Phase GFCI.c
 *****/

// 10/13/2011
// OVD disabled
// contactor recloser removed
// impulse checking removed

// ****

// System connections:
// ADC channel 1 input - Main
// ADC channel 2 input - Feeder 1
// ADC channel 3 input - Feeder 2
// ADC channel 4 input - Feeder 3
// ADC channel 5 input - Feeder 4
// Note: feeder channels are interchangeable
// ADC channel 6 input - Ground Current CT (will not trip)
// ADC channel 7 input - Reference Phasor

// ****

#include <stdlib.h>
#include <math.h>
#include <services/services.h>
#include <cdefBF538.h>
#include <ccblkfn.h>
#include <sys\exception.h>
#include <stdio.h>

#define TRIP_THRESHOLD 5.0 // sets sytem to trip at 5mA current threshold (to meet 4-6mA tripping with 1mA
resolution)
#define OVD_US_REQUIRED_FOR_EVENT 25 // TMR2 @ 500kHz; Set to (desired us)/2 (shown is 50us min time)
```

```

#define PHASE_DISABLE 0
#define PHASE_ENABLE 1

/* //OVD disabled
EX_INTERRUPT_HANDLER (PF_ISR); // ISR prototype for PF_ISR
EX_INTERRUPT_HANDLER (Timer2_ISR); // ISR prototype for Timer_ISR
void Timer2_init (void);
*/

void Clock_init (void);
void SPI0_init (void);
void GPIO_inits (void);
void Timer0_init (void);
void Timer1_init (void);
void PinChange_init(void);
void ADC_init(void);
void UART_init(void);
void SPI0_Convst(void);
void UART_Transmit(unsigned char data);
void UART_TransmitString (const char *str);
void ResetADC(void);
void delay_us(int delay);
void Contactor_controls(void);
int Trip_threshold(double Ym, double Y1, double Y2, double Y3, double Y4);
int TrippingLogic(double Ym, double Y1, double Y2, double Y3, double Y4);
int FindMax(double Y1, double Y2, double Y3, double Y4);
int SPI0_Recieve1(void);
int Convert2sComp(int num);
int Return_2nd_Max(int Y0, int Y1, int Y2, int Y3);
double FullCycleFilter(int adcValue, int X[], double Y[], int X_start, int Y_start, int ph);
double ReturnMax(double Y0, double Y1, double Y2, double Y3, double Y4);

// Global variables
int Contactor_M = 0; //states to set contactors to
int Contactor_1 = 0;
int Contactor_2 = 0;
int Contactor_3 = 0;

```

```

int Contactor_4 = 0;
int Contactor_F = 0;
int release_delay_count = 0; //counts delay of contractor opening due to phase angle
int release_delay = 0;
int record = 0; //enables storing of next 100 samples
int record_count = 0;
double phase_trip = 0; //used to track and control phase angle contactors open at
double V_AB_phase = 0;

/* OVD disabled
// OVD global variables
int OVD_initiated_trip = 0;
int OVD_initiated_trip_record = 0; // Sticky bit indicating source of last trip
int OVD_M = 0;
int OVD_1 = 0;
int OVD_2 = 0;
int OVD_3 = 0;
int OVD_4 = 0;
int OVD_i = 0;
int OVD_j_m = 0;
int OVD_j_1 = 0;
int OVD_j_2 = 0;
int OVD_j_3 = 0;
int OVD_j_4 = 0;
*/

int main(void)
{
    // *** Initializations ***
    Clock_init(); // Init processor clocks: cclk = 250MHz, sclk = 100MHz
    UART_init(); // Init UART: 8 data bits, 1 stop bit, no parity, 921600 bps
    Timer0_init(); // Trigger @ 960Hz, controls main program loop/sampling rate
    Timer1_init(); // Trigger @ 1MHz, used for a programmable delay (in us)
    SPI0_init(); // 16 data bits, msb 1st, cpol=1, cpha=1, sck=8.33MHz
    ADC_init(); // Init AD7606 and controlling GPIOs
    GPIO_inits(); // Init contactor control GPIOs + misc.
    /* //OVD disabled

```

```

Timer2_init(); // Trigger @ 500kHz
PinChange_init(); // Init PF7 pin change interrupt, update ISR (toggles PD2 now)
*/

// *** Variable declarations ***
int i=1; //counter for printing filter outputs during regular loop program flow
int j=0; //counter for ~200ms delay on startup for rails to stabilize
int k=100; //global trip enable delay counter (~100ms delay after trip decision)
//int l=0;
int m=0; //print loop counter
//int n=0;

int adcValue1 = 0; //inputs from ADC
int adcValue2 = 0;
int adcValue3 = 0;
int adcValue4 = 0;
int adcValue5 = 0;
int adcValue6 = 0;
int adcValue7 = 0;
int adcValue8 = 0;

int adcValue1_2 = 0; //duplicate inputs from ADC to verify no transmit errors
int adcValue2_2 = 0;
int adcValue3_2 = 0;
int adcValue4_2 = 0;
int adcValue5_2 = 0;
int adcValue6_2 = 0;
int adcValue7_2 = 0;
int adcValue8_2 = 0;

int X1[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}; //filter input arrays
int X2[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
int X3[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
int X4[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
int X5[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
int X6[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
int X7[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};

```

```

double Y1[5] = {0,0,0,0,0}; //filter output arrays
double Y2[5] = {0,0,0,0,0};
double Y3[5] = {0,0,0,0,0};
double Y4[5] = {0,0,0,0,0};
double Y5[5] = {0,0,0,0,0};
double Y6[5] = {0,0,0,0,0};
double Y7[5] = {0,0,0,0,0};

double Y_mag1 = 0; //filter magnitude output (mA estimate of primary current)
double Y_mag2 = 0;
double Y_mag3 = 0;
double Y_mag4 = 0;
double Y_mag5 = 0;
double Y_mag6 = 0;
double Y_mag7 = 0;
double Y_mag1_store = 0; //storeage of filter magnitude output
double Y_mag2_store = 0;
double Y_mag3_store = 0;
double Y_mag4_store = 0;
double Y_mag5_store = 0;
double Y_mag6_store = 0;
double Y_mag7_store = 0;

int X_start = 0; //filter starting indicies (linear arrays mimicking circular buffer)
int Y_start = 0;

double Ymag_data1[150]; //filter magnitude output storeage
double Ymag_data2[150];
double Ymag_data3[150];
double Ymag_data4[150];
double Ymag_data5[150];
double Ymag_data6[150];
double Ymag_data7[150];

int ADC_data1[150]; //ADC input storeage
int ADC_data2[150];
int ADC_data3[150];
int ADC_data4[150];

```

```

int ADC_data5[150];
int ADC_data6[150];
int ADC_data7[150];
int phase_data[150]; //phase data storage
int OVD_ISR_count_array[150];

int timercount = 0;
int trip_enable_global = 1;
int trip_enable_filter = 1;
int filter_delay = 0;
int filter_delay_enable = 0;
int CB_trip = 99; // contactor to trip (initialized to 99 but otherwise sticky)
int tripped_filter_delay = 0;
int tripped_filter_delay_count = 0;
int initial_threshold_passed_count = 0;
int trip_decision_made_count = 0;
int currently_delaying = 0;
int filter_delay_max = 99;

int XM_post_threshold[6] = {0,0,0,0,0,0}; //stores 6 inputs following current passing threshold
int X1_post_threshold[6] = {0,0,0,0,0,0};
int X2_post_threshold[6] = {0,0,0,0,0,0};
int X3_post_threshold[6] = {0,0,0,0,0,0};
int X4_post_threshold[6] = {0,0,0,0,0,0};
double YM_post_threshold[6] = {0,0,0,0,0,0}; //stores 6 outputs following current passing threshold
double Y1_post_threshold[6] = {0,0,0,0,0,0};
double Y2_post_threshold[6] = {0,0,0,0,0,0};
double Y3_post_threshold[6] = {0,0,0,0,0,0};
double Y4_post_threshold[6] = {0,0,0,0,0,0};
double YM_max = 0;
double Y1_max = 0;
double Y2_max = 0;
double Y3_max = 0;
double Y4_max = 0;
double Y_max = 0;

int high_fault_count = 1000;
int print = 0;

```

```

int SPI_data_good = 0;
int ADC_status = 0;

// *** Start-up/one-time run ***

//init UART variables and transmit test char
unsigned char UARTdata = '~';
char outputstring[7];
sprintf(outputstring, "\r\n%c\r\n", UARTdata);
UART_TransmitString(outputstring);

while (j < 200) // ~200ms delay for ADC rails to power up and establish references, filters to settle
to DC offsets
{
    while(!(*pTIMER_STATUS & 0x0001));
    *pTIMER_STATUS |= 0x0001;
    j++;
}
*pPORTDIO_SET = 0x0002; // Enables opto-coupler buffer output long after 3.3V rails are up
//*pPORTFIO_INEN = 0x0F80; // Enable PF7-PF11 as input //OVD disabled

// *****
// *** Main program, round-robin flow ***
// *****

while(1)
{
    // *** Idles/wait for 960 Hz interrupt ***
    while(!(*pTIMER_STATUS & 0x0001));
    *pTIMER_STATUS |= 0x0001; // W1C (write 1 to clear) TMR0 interrupt bit
    release_delay_count++; // Keeps track of delays due to contactor withholding while waiting for
phase to open at

```



```

// *****
// *** Data Sampling and Verification ***
// *****

// *** ADC conversion and transmit *** - verifies data transmit is valid to mitigate EMI
while(!ADC_status)
{
    SPI0_Convst(); //ADC start conversion

    adcValue1 = Convert2sComp(SPI0_Recieve1()); //recieve channel 1 output; convert 2s
compliment to signed int
    adcValue2 = Convert2sComp(SPI0_Recieve1());
    adcValue3 = Convert2sComp(SPI0_Recieve1());
    adcValue4 = Convert2sComp(SPI0_Recieve1());
    adcValue5 = Convert2sComp(SPI0_Recieve1());
    adcValue6 = Convert2sComp(SPI0_Recieve1());
    adcValue7 = Convert2sComp(SPI0_Recieve1());
    adcValue8 = Convert2sComp(SPI0_Recieve1());

    // Verifies non-corrupt SPI transmission from ADC
    while(!SPI_data_good)
    {
        adcValue1_2 = Convert2sComp(SPI0_Recieve1());
        adcValue2_2 = Convert2sComp(SPI0_Recieve1());
        adcValue3_2 = Convert2sComp(SPI0_Recieve1());
        adcValue4_2 = Convert2sComp(SPI0_Recieve1());
        adcValue5_2 = Convert2sComp(SPI0_Recieve1());
        adcValue6_2 = Convert2sComp(SPI0_Recieve1());
        adcValue7_2 = Convert2sComp(SPI0_Recieve1());
        adcValue8_2 = Convert2sComp(SPI0_Recieve1());
        if (adcValue1_2 == adcValue1 && adcValue2_2 == adcValue2 && adcValue3_2 ==
adcValue3 && adcValue4_2 == adcValue4 && adcValue5_2 == adcValue5 && adcValue6_2 == adcValue6 &&
adcValue7_2 == adcValue7)
            SPI_data_good = 1; // data is valid if both transmits are identical,
otherwise shift and transmit again
        else
        {

```

```

        adcValue1 = adcValue1_2;
        adcValue2 = adcValue2_2;
        adcValue3 = adcValue3_2;
        adcValue4 = adcValue4_2;
        adcValue5 = adcValue5_2;
        adcValue6 = adcValue6_2;
        adcValue7 = adcValue7_2;
        adcValue8 = adcValue8_2;
    }
}
SPI_data_good = 0; //reset flag

// Resets ADC if all inputs are zero (DC offsets prevent this under normal operation)
if (adcValue1 == 0 && adcValue2 == 0 && adcValue3 == 0 && adcValue4 == 0 && adcValue5 ==
0 && adcValue6 == 0 && adcValue7 == 0)
    ResetADC();

else
    ADC_status = 1;
}
ADC_status = 0;

// *****
// *** Filtering ***
// *****

Y_mag1 = FullCycleFilter(adcValue1, X1, Y1, X_start, Y_start, PHASE_DISABLE);
Y_mag2 = FullCycleFilter(adcValue2, X2, Y2, X_start, Y_start, PHASE_DISABLE);
Y_mag3 = FullCycleFilter(adcValue3, X3, Y3, X_start, Y_start, PHASE_DISABLE);
Y_mag4 = FullCycleFilter(adcValue4, X4, Y4, X_start, Y_start, PHASE_DISABLE);
Y_mag5 = FullCycleFilter(adcValue5, X5, Y5, X_start, Y_start, PHASE_DISABLE);
Y_mag6 = FullCycleFilter(adcValue6, X6, Y6, X_start, Y_start, PHASE_DISABLE);
Y_mag7 = FullCycleFilter(adcValue7, X7, Y7, X_start, Y_start, PHASE_ENABLE); //only calc phase
angle for channel 7

```

```

// Shift filter starting indicies
X_start--; //mimics circular buffer of size 16
if(X_start== -1)
    X_start = 15;

Y_start--; //mimics circular buffer of size 5
if(Y_start == -1)
    Y_start = 4;

// *****
// *** Relay Tripping Logic ***
// *****

// OVD was found to be unnecessary and is disabled
if (trip_enable_global) //cleared after trip decisions to implement ~100ms delay before next
trip
{
    /*
    // OVD tripping disabled - system found able to detect high currents though CTs
    // Wait until OVD goes high, OR we get filter output above 5mA
    // OVD always has higher priority than filter outputs
    if (OVD_initiated_trip) // True when OVD event is confirmed via ISR
    {
        // Since here we know we're going to trip something, examining filter outputs for
tripping
        // will be abandoned, and its associated variables reset in the general post-trip
reset loop

        // Determine the contactor to trip by passing 1's and 0's into the tripping logic
(2b resolution)
        CB_trip = TrippingLogic(OVD_M,OVD_1,OVD_2,OVD_3,OVD_4);

        // Update the necessary contactor states for clearing of the fault
        if (CB_trip == 0)
            Contactor_M = 0;

```

```

else if (CB_trip == 1)
    Contactor_1 = 0;

else if (CB_trip == 2)
    Contactor_2 = 0;

else if (CB_trip == 3)
    Contactor_3 = 0;

else if (CB_trip == 4)
    Contactor_4 = 0;

trip_enable_global = 0;
k=0;
OVD_initiated_trip = 0; // Clear flag, HW interrupts will not be enabled until

global reset

// set timing variables
release_delay_count = 0;
initial_threshold_passed_count = record_count+1;
trip_decision_made_count = record_count+1;
OVD_initiated_trip_record = 1; // sticky bit indicating source of last trip

}
else if(trip_enable_filter) //if OVD is enabled
*/

if (trip_enable_filter) //always high, filter tripping is never disabled
{
    // From steady-state, any filter output must pass the 5mA threshold
    if (Trip_threshold(Y_mag1,Y_mag2,Y_mag3,Y_mag4,Y_mag5) && !currently_delaying)
    {
        currently_delaying = 1; //indicates 5-25ms (depending on current magnitude)
delay after threshold is passed to ensure it was due to fault current
        filter_delay = -1;
        initial_threshold_passed_count = record_count+1; // Add +1 since record_count
starts from 0
    }
}

```

```

// Post-threshold detection, pre-tripping action
if (currently_delaying)
{
    //delay at least 5 samples to ensure fault current (can delay 5 and still
trip w/in 25ms)

    filter_delay++; //counts delay from initial threshold passed to trip decision

    // store the threshold passed sample and 5 following ones
    if (filter_delay <= 5)
    {
        YM_post_threshold[filter_delay] = Y_mag1;
        Y1_post_threshold[filter_delay] = Y_mag2;
        Y2_post_threshold[filter_delay] = Y_mag3;
        Y3_post_threshold[filter_delay] = Y_mag4;
        Y4_post_threshold[filter_delay] = Y_mag5;
        XM_post_threshold[filter_delay] = adcValue1;
        X1_post_threshold[filter_delay] = adcValue2;
        X2_post_threshold[filter_delay] = adcValue3;
        X3_post_threshold[filter_delay] = adcValue4;
        X4_post_threshold[filter_delay] = adcValue5;

        if (filter_delay == 5)
        {
            // set the remaining number of samples to delay before the
            // i.e. for large current trip immediately, so filter_delay_max
            //find maximum output of all channels since threshold was passed
            YM_max =
ReturnMax(YM_post_threshold[1],YM_post_threshold[2],YM_post_threshold[3],YM_post_thres
hold[5]);
            Y1_max =
ReturnMax(Y1_post_threshold[1],Y1_post_threshold[2],Y1_post_threshold[3],Y1_post_thres
hold[5]);

```

```

        Y2_max =
ReturnMax(Y2_post_threshold[1],Y2_post_threshold[2],Y2_post_threshold[3],Y2_post_threshold[4],Y2_post_thres
hold[5]);
        Y3_max =
ReturnMax(Y3_post_threshold[1],Y3_post_threshold[2],Y3_post_threshold[3],Y3_post_threshold[4],Y3_post_thres
hold[5]);
        Y4_max =
ReturnMax(Y4_post_threshold[1],Y4_post_threshold[2],Y4_post_threshold[3],Y4_post_threshold[4],Y4_post_thres
hold[5]);
        Y_max = ReturnMax(YM_max,Y1_max,Y2_max,Y3_max,Y4_max);

        //set remaining filter delay
        if (Y_max <= 30)
            filter_delay_max = filter_delay+20; // wait an additional
20 samples; total delay is 25 samples

        else if (Y_max <= 50)
            filter_delay_max = -1*(Y_max-50)+5; // wait between 5-25
samples; linear interpolation

        else
            filter_delay_max = filter_delay+0; // trip immediately;
total delay is 5 samples
    }
}

// trip if reached set max delay
if (filter_delay >= filter_delay_max)
{
    if (Trip_threshold(Y_mag1,Y_mag2,Y_mag3,Y_mag4,Y_mag5)) // Verify last
outputs are above the trip threshold
    {
        // Tripping decision
        CB_trip = TrippingLogic(Y_mag1,Y_mag2,Y_mag3,Y_mag4,Y_mag5);

        if (CB_trip == 0)
            Contactor_M = 0;
    }
}

```

```

else if (CB_trip == 1)
    Contactor_1 = 0;

else if (CB_trip == 2)
    Contactor_2 = 0;

else if (CB_trip == 3)
    Contactor_3 = 0;

else if (CB_trip == 4)
    Contactor_4 = 0;

k=0;
trip_enable_global = 0;

// set timing variables
release_delay_count = 0;
trip_decision_made_count = record_count+1;
// OVD_initiated_trip_record = 0; // sticky bit indicating source
of last trip //OVD disabled
}

filter_delay = -1; // At the end of the filter delay, ALWAYS reset the
delay count
currently_delaying = 0; // At the end of the filter delay, ALWAYS reset
the currently delaying flag
}
} // end if(trip_enable_filter)
} // end if(trip_enable_global)

// *** Reset tripping logic variables ***

k++;
if (k == 100) // Instructions occurs 100 samples after reset (k=0); allows contactor opening
and filter settling
{

```

```

        /* //OVD disabled
        // OVD specific
        if (OVD_initiated_trip_record) // This is a sticky bit, so do not reset it. Filter will
reset it if need be.
        {
            OVD_i = 0;
            OVD_j_m = 0;
            OVD_j_1 = 0;
            OVD_j_2 = 0;
            OVD_j_3 = 0;
            OVD_j_4 = 0;
            *pPORTFIO_MASKA_SET = 0x0080; // Re-enable the HW OVD interrupt
        }
        */

        // Tripping logic general
        filter_delay = -1;
        currently_delaying = 0;
        trip_enable_global = 1;
        trip_enable_filter = 1;

        // Insert code to run de-gaussing circuit here, if necessary

    }
    else if (k > 200)
        k--; // keeps k from incrementing past 200

    // *** High fault contactor control ***

    // Opens contactor 5 100ms after closed (used for safe testing of high fault currents)
    high_fault_count++;
    if (high_fault_count == 100)
    {
        Contactor_F = 0;
    }
    else if (high_fault_count > 100)
        high_fault_count--; //keeps from incrementing past 100

```



```

// *****
// *** Update desired contactor states (if at adequate phase angle) ***
// *****

// checks if contactor states matched desired settings, only opens contactors if at valid phase
Contactor_controls();

// *****
// *** PC Communication and Print Outs ***
// *****

if(*pUART0_LSR & 0x0001) // Check for new UART data from PC
{
    UARTdata = *pUART0_RBR; // Grab new data from recieve register

    if(UARTdata == 'r') // used to reset ADC when crashes due to EMI
        ResetADC();

    // Prints last 100 sample data set recorded (puts relay functionality on hold)
    else if(UARTdata == 'p')
        print = 1;

    else if(UARTdata == 'm')
    {
        if (Contactor_M)
            release_delay_count = 0; //counts time between trip decision and actual trip
when opening(waiting for phase angle)
        Contactor_M = !Contactor_M; //toggle contactor Main
    }

    else if(UARTdata == '1')
    {
        if (Contactor_1)

```

```

        release_delay_count = 0;
        Contactor_1 = !Contactor_1;
    }

else if(UARTdata == '2')
{
    if (Contactor_2)
        release_delay_count = 0;
    Contactor_2 = !Contactor_2;
}

else if(UARTdata == '3')
{
    if (Contactor_3)
        release_delay_count = 0;
    Contactor_3 = !Contactor_3;
}

else if(UARTdata == '4')
{
    if (Contactor_4)
        release_delay_count = 0;
    Contactor_4 = !Contactor_4;
}

else if(UARTdata == '5')
{
    if (Contactor_F)
        release_delay_count = 0;
    Contactor_F = !Contactor_F;
}

// x to open all contactors prior to system shutdown
else if(UARTdata == 'x')
{
    Contactor_M = 0;
    Contactor_1 = 0;
    Contactor_2 = 0;

```

```

        Contactor_3 = 0;
        Contactor_4 = 0;
        Contactor_F = 0;
    }

    else if(UARTdata == 'h')
    {
        Contactor_F = 1;
        high_fault_count = 0; //automatically opens high fault current contactors after
100ms
        record = 1;
    }

    // Every switching action will store the next 100 samples
    if ((UARTdata == 'm') || (UARTdata == '1') || (UARTdata == '2') || (UARTdata == '3') ||
(UARTdata == '4') || (UARTdata == '5') || (UARTdata == 'h'))
    {
        record = 1;
        CB_trip = 99;
    }
} //end if(new UART data)

if (record)
{
    if (record_count >= 100)
    {
        record = 0;
        record_count = 0;
    }
    else //record current outputs and inputs for 100 times through the loop
    {
        Ymag_data1[record_count] = Y_mag1;
        Ymag_data2[record_count] = Y_mag2;
        Ymag_data3[record_count] = Y_mag3;
        Ymag_data4[record_count] = Y_mag4;
        Ymag_data5[record_count] = Y_mag5;
        Ymag_data6[record_count] = Y_mag6;
        Ymag_data7[record_count] = Y_mag7;
    }
}

```

```

        ADC_data1[record_count] = 0.1698*1.414*adcValue1; // scales the instantaneous ADC
samples to instantaneous mA
        ADC_data2[record_count] = 0.1698*1.414*adcValue2;
        ADC_data3[record_count] = 0.1698*1.414*adcValue3;
        ADC_data4[record_count] = 0.1698*1.414*adcValue4;
        ADC_data5[record_count] = 0.1698*1.414*adcValue5;
        ADC_data6[record_count] = 0.1698*1.414*adcValue6;

        record_count++;
    }
}

if (print) //prints 100 stored samples and pauses sytem
{
    sprintf(outputstring, "\n\rFilter Output:\n\r");
    UART_TransmitString(outputstring);

    for(m=0; m<100; m++)
    {
        sprintf(outputstring, "%.1f\t", Ymag_data1[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%.1f\t", Ymag_data2[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%.1f\t", Ymag_data3[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%.1f\t", Ymag_data4[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%.1f\t", Ymag_data5[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%.1f\r\n", Ymag_data6[m]);
        UART_TransmitString(outputstring);
    }

    sprintf(outputstring, "Filter Input:\n\r");
    UART_TransmitString(outputstring);
    for(m=0; m<100; m++)

```

```

    {
        sprintf(outputstring, "%d\t", ADC_data1[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%d\t", ADC_data2[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%d\t", ADC_data3[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%d\t", ADC_data4[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%d\t", ADC_data5[m]);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "%d\r\n", ADC_data6[m]);
        UART_TransmitString(outputstring);
    }

    if (!(CB_trip == 99)) // Only print out tripping metrics if the last switching action
caused a trip
    {
        /* //OVD disabled
        if (OVD_initiated_trip_record)
        {
            sprintf(outputstring, "OVD initiated instantaneous trip\n\r");
            UART_TransmitString(outputstring);
            sprintf(outputstring, "6.26A saturation threshold passed sometime before
sample: \t%d\n\r", initial_threshold_passed_count);
            UART_TransmitString(outputstring);
            sprintf(outputstring, "Trip decision made right after sample: \t%d\n\r",
trip_decision_made_count);
            UART_TransmitString(outputstring);
            sprintf(outputstring, "Tripped contactor: \t%d\n\r", CB_trip);
            UART_TransmitString(outputstring);
        }

        else
        */
        {
            sprintf(outputstring, "Filter initiated delayed trip\n\r");
            UART_TransmitString(outputstring);

```

```

        sprintf(outputstring, "5.0mA threshold passed sometime before sample:
\t%d\n\r", initial_threshold_passed_count);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "Total filter delay (samples): \t%d\n\r",
filter_delay_max);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "Trip decision made right after sample:
\t%d\n\r", trip_decision_made_count);
        UART_TransmitString(outputstring);
        sprintf(outputstring, "Tripped contactor: \t%d\n\r", CB_trip);
        UART_TransmitString(outputstring);
    }
}

else
{
    sprintf(outputstring, "No trip from last switching action\n\r");
    UART_TransmitString(outputstring);
}

/*
//print data about phase angle tripping and delay due to it - disabled
sprintf(outputstring, "Release delay (samples): \t%d\n\r", release_delay);
UART_TransmitString(outputstring);
sprintf(outputstring, "Phase of release: %.1f degrees\n\r", phase_trip);
UART_TransmitString(outputstring);
sprintf(outputstring, "YM_max = %.1f\n\r", YM_max);
UART_TransmitString(outputstring);
sprintf(outputstring, "Y1_max = %.1f\n\r", Y1_max);
UART_TransmitString(outputstring);
sprintf(outputstring, "Y2_max = %.1f\n\r", Y2_max);
UART_TransmitString(outputstring);
sprintf(outputstring, "Y3_max = %.1f\n\r", Y3_max);
UART_TransmitString(outputstring);
sprintf(outputstring, "Y4_max = %.1f\n\r", Y4_max);
UART_TransmitString(outputstring);
*/

```

```

        sprintf(outputstring, "End transient data, press 'c' to continue\n\r");
        UART_TransmitString(outputstring);
        while(UARTdata != 'c') //waits for 'c' from UART
        {
            if((*pUART0_LSR & 0x0001))
                UARTdata = *pUART0_RBR;
        }
        print = 0;
    }

    // Prints out every 6th sample in real-time; for steady-state verification only

    if(i == 7)
        i = 1;

    if(i == 1)
    {
        Y_mag1_store = Y_mag1; //store simultaneous filter outputs to be transmitted 1 per cycle
for next 6 cycles
        Y_mag2_store = Y_mag2;
        Y_mag3_store = Y_mag3;
        Y_mag4_store = Y_mag4;
        Y_mag5_store = Y_mag5;
        Y_mag6_store = Y_mag6;
    }
    switch(i)
    {
        case 1:
            sprintf(outputstring, "%.1f\t", Y_mag1_store);
            UART_TransmitString(outputstring);
            break;

        case 2:
            sprintf(outputstring, "%.1f\t", Y_mag2_store);
            UART_TransmitString(outputstring);
            break;

        case 3:

```

```

        sprintf(outputstring, "%.1f\t", Y_mag3_store);
        UART_TransmitString(outputstring);
        break;

        case 4:
        sprintf(outputstring, "%.1f\t", Y_mag4_store);
        UART_TransmitString(outputstring);
        break;

        case 5:
        sprintf(outputstring, "%.1f\r\n", Y_mag5_store);
        UART_TransmitString(outputstring);
        break;

        case 6: //ground current CT output disabled
        //sprintf(outputstring, "%.1f\t", Y_mag6_store);
        //UART_TransmitString(outputstring);
        //sprintf(outputstring, "%d\r\n", recloser_count);
        //UART_TransmitString(outputstring);
        break;
    }
    i++;
} // end while(1)
} // end main

// *****

void Contactor_controls(void)
{
    // Opens contactors only at a valid phase angle to avoid EMI

    // Compares GPIO data register with global contactor_x variable
    // If they're different, and global var = 0 (contactor to be opened), it checks if in the right phase
margin, and if so opens
    // Otherwise simply closes it (no EMI during closing so no phase restriction)
    // Repeat for all contactors
    // Bounds det. empirically "best by test": 146.5+/-11.25, (146.5-180)+/-11.25

```



```

    if (!(Contactor_M && (*pPORTCIO_SET & 0x0020)) && (Contactor_M || (*pPORTCIO_SET & 0x0020))) //if
states are different
    {
        if (*pPORTCIO_SET & 0x0020) //if trying to open
        {
            if ((V_AB_phase <= 157.75 && V_AB_phase >= 135.25) || (V_AB_phase <= -22.25 && V_AB_phase
>= -44.75))
            {
                *pPORTCIO_CLEAR = 0x0020;
                release_delay = release_delay_count;
                phase_trip = V_AB_phase;
            }
        }
        else
        {
            *pPORTCIO_SET = 0x0020;
        }
    }

    if (!(Contactor_1 && (*pPORTCIO_SET & 0x0040)) && (Contactor_1 || (*pPORTCIO_SET & 0x0040)))
    {
        if (*pPORTCIO_SET & 0x0040)
        {
            if ((V_AB_phase <= 157.75 && V_AB_phase >= 135.25) || (V_AB_phase <= -22.25 && V_AB_phase
>= -44.75))
            {
                *pPORTCIO_CLEAR = 0x0040;
                release_delay = release_delay_count;
                phase_trip = V_AB_phase;
            }
        }
        else
        {
            *pPORTCIO_SET = 0x0040;
        }
    }
}

```

```

if (!(Contactor_2 && (*pPORTCIO_SET & 0x0080)) && (Contactor_2 || (*pPORTCIO_SET & 0x0080)))
{
    if (*pPORTCIO_SET & 0x0080)
    {
        if ((V_AB_phase <= 157.75 && V_AB_phase >= 135.25) || (V_AB_phase <= -22.25 && V_AB_phase
>= -44.75))
        {
            *pPORTCIO_CLEAR = 0x0080;
            release_delay = release_delay_count;
            phase_trip = V_AB_phase;
        }
    }
    else
    {
        *pPORTCIO_SET = 0x0080;
    }
}

if (!(Contactor_3 && (*pPORTCIO_SET & 0x0100)) && (Contactor_3 || (*pPORTCIO_SET & 0x0100)))
{
    if (*pPORTCIO_SET & 0x0100)
    {
        if ((V_AB_phase <= 157.75 && V_AB_phase >= 135.25) || (V_AB_phase <= -22.25 && V_AB_phase
>= -44.75))
        {
            *pPORTCIO_CLEAR = 0x0100;
            release_delay = release_delay_count;
            phase_trip = V_AB_phase;
        }
    }
    else
    {
        *pPORTCIO_SET = 0x0100;
    }
}

if (!(Contactor_4 && (*pPORTCIO_SET & 0x0200)) && (Contactor_4 || (*pPORTCIO_SET & 0x0200)))
{

```

```

        if (*pPORTCIO_SET & 0x0200)
        {
            if ((V_AB_phase <= 157.75 && V_AB_phase >= 135.25) || (V_AB_phase <= -22.25 && V_AB_phase
>= -44.75))
            {
                *pPORTCIO_CLEAR = 0x0200;
                release_delay = release_delay_count;
                phase_trip = V_AB_phase;
            }
        }
        else
        {
            *pPORTCIO_SET = 0x0200;
        }
    }

    if (!(Contactor_F && (*pPORTDIO_SET & 0x0001)) && (Contactor_F || (*pPORTDIO_SET & 0x0001)))
    {
        if (*pPORTDIO_SET & 0x0001)
        {
            if ((V_AB_phase <= 157.75 && V_AB_phase >= 135.25) || (V_AB_phase <= -22.25 && V_AB_phase
>= -44.75))
            {
                *pPORTDIO_CLEAR = 0x0001;
                release_delay = release_delay_count;
                phase_trip = V_AB_phase;
            }
        }
        else
        {
            *pPORTDIO_SET = 0x0001;
        }
    }
}

int Convert2sComp(int num)
{
    // Converts from 16b 2s complement (ADC output) to 16b signed int (filter input)

```

```

    if(num > 32767)
        return (-(65536 - num));

    else
        return num;
}

int TrippingLogic(double Ym, double Y1, double Y2, double Y3, double Y4)
{
    // This function assumes the input values are large enough s.t. a trip is warranted
    // If main is within +/- 40% of largest feeder trip feeder, otherwise trip main
    // Returns 1-4 for respective feeder trip, 0 for main

    double percent_threshold = 0.4;

    switch(FindMax(Y1,Y2,Y3,Y4))
    {
        case 1:
            if((Ym > (1-percent_threshold)*Y1) && (Ym < (1+percent_threshold)*Y1))
                return 1;
            else
                return 0;

        case 2:
            if((Ym > (1-percent_threshold)*Y2) && (Ym < (1+percent_threshold)*Y2))
                return 2;
            else
                return 0;

        case 3:
            if((Ym > (1-percent_threshold)*Y3) && (Ym < (1+percent_threshold)*Y3))
                return 3;
            else
                return 0;

        case 4:
            if((Ym > (1-percent_threshold)*Y4) && (Ym < (1+percent_threshold)*Y4))

```

```

        return 4;
    else
        return 0;
    }
    return 0;
}

int FindMax(double Y1, double Y2, double Y3, double Y4)
{
    // Returns index of the maximum of the input arguments

    double value = Y1;
    int max = 1;

    if(Y2 > value)
    {
        value = Y2;
        max = 2;
    }
    if(Y3 > value)
    {
        value = Y3;
        max = 3;
    }
    if(Y4 > value)
    {
        value = Y4;
        max = 4;
    }
    return max;
}

double ReturnMax(double Y0, double Y1, double Y2, double Y3, double Y4)
{
    // Returns the maximum value of the input arguments

    double max = Y0;

```

```

    if(Y1 > max)
        max = Y1;

    if(Y2 > max)
        max = Y2;

    if(Y3 > max)
        max = Y3;

    if(Y4 > max)
        max = Y4;

    return max;
}

int Return_2nd_Max(int Y0, int Y1, int Y2, int Y3)
{
    // Returns the 2nd largest value of the input arguments
    // Assumes all input arguments are positive - true since they are magnitudes of the 2nd difference

    // Determines the index of the largest input
    int value = Y0;
    int index = 0;

    if(Y1 > value)
    {
        value = Y1;
        index = 1;
    }
    if(Y2 > value)
    {
        value = Y2;
        index = 2;
    }
    if(Y3 > value)
    {
        value = Y3;

```

```

        index = 3;
    }

    // Determines the largest magnitude, precluding the largest input
    int max = 0;

    if(Y0 > max && !(index == 0))
        max = Y0;

    if(Y1 > max && !(index == 1))
        max = Y1;

    if(Y2 > max && !(index == 2))
        max = Y2;

    if(Y3 > max && !(index == 3))
        max = Y3;

    return max;
}

void GPIO_inits (void)
{
    // Initializes GPIOs for contactor control and misc
    // Note: only set each FER/DIR once

    *pPORTCIO_FER = 0x03E0; // PC5-9 (main, fdr1-4 contactor control) GPIO inits
    *pPORTCIO_DIR = 0x03E0;

    *pPORTDIO_FER = 0x000F; // PD0 (fault contactor), PD1 (n/a), PD2 (timing), PD3 (?) GPIO inits
    *pPORTDIO_DIR = 0x000F;
}

void Clock_init (void)
{
    // Sets processor clocks: cclk=250MHz, sclk=100MHz

```

```

    *pPLL_CTL = 0x02800; // Set VCO = 20x CLKIN = 500MHz (CLKIN = 25MHz from EZKIT)
    unsigned int temp; // Put processor into idle to change VCO
    temp = cli();
    idle();
    sti(temp);
    *pPLL_DIV = 0x0015;      // Set SCLK = VCO/5 = 100MHz, CCLK = VCO/2 = 250MHz
}

void Timer0_init (void)
{
    // Configures Timer0 which controls the 960Hz sampling and round-robin progression of program

    *pTIMER0_CONFIG = 0x001D; // Emu run off, irq enable, pwm_out mode, output pad enable
    *pTIMER0_PERIOD = 104167; // 959.99 Hz nominal period for 100MHz sclk
    *pTIMER0_WIDTH = 52083; // 50% duty cycle (half of period)
    *pTIMER_ENABLE = 0x0001; // Enable timer0
}

void Timer1_init (void)
{
    // Configures Timer1 to 1MHz for variable length us delays

    *pTIMER1_CONFIG = 0x001D; // Emu run off, irq enable, pwm_out mode, output pad enable
    *pTIMER1_PERIOD = 100; // 1MHz nominal period for 100MHz sclk (activates every 1us)
    *pTIMER1_WIDTH = 50; // 50% duty cycle (half of period)
    //note: dont enable timer here; timer is enabled/disabled in each delay function for a constant delay
}

void ADC_init(void)
{
    // Configures and initializes communications with the sensor PCB

    *pPORTEIO_FER = 0x00DC; // W1 for GPIO functionality: reset convst !stdby, busy1 busy2 (PE2-PE4, PE6-7)
    *pPORTEIO_DIR = 0x001C; // W1 for output: reset convst !stdby (PE2-PE4)
    *pPORTEIO_INEN = 0x00C0; // W1 for input: busy1 busy2 (PE6-PE7)
    *pPORTFIO_DIR = 0x0060; // W1 for output: !cs1 !cs2 (PF6-PF5); default 0 for input: PF7-PF11

    // Brings ADC into full shutdown mode

```



```

    *pPORTEIO_CLEAR = 0x0010; // Set !stdby low (active low)
    delay_us(1);
    *pPORTEIO_SET = 0x0010; // Set !stdby high (active low)
    delay_us(1);

    ResetADC();
}

void ResetADC(void)
{
    // Sends reset pulse to ADC

    *pPORTEIO_SET = 0x0004;
    delay_us(1);
    *pPORTEIO_CLEAR = 0x0004;
}

void UART_init(void)
{
    // Initializes UART communications with PC hyperterminal

    *pUART0_LCR = 0x0083; // Enable access to DLL,DLH; 8bit,1 stop bit, no parity
    *pUART0_DLH = 0; // Divisor latch registers to set baud rate
    *pUART0_DLL = 7; // 7 for 921600 bps at 100MHz sclk (54 for 115200 bps at 100MHz sclk)
    *pUART0_MCR = 0x0000; // Modem control set to 0 - loopback disabled
    *pUART0_GCTL = 0x0001; // Enable uart clocks
    *pUART0_LCR = 0x0003; // Enable access to THR,RBR,IER; 8bit,1 stop bit, no parity
}

void UART_Transmit(unsigned char data)
{
    // Transmits a char over UART

    while(!(*pUART0_LSR & 0x0020)); // Wait for empty transmit buffer (THR)
    *pUART0_THR = data; // Put data into buffer, sends the data
}

```

```

void UART_TransmitString (const char *str)
{
    // Transmits a string of chars over UART

    while (*str)
    {
        UART_Transmit(*str);
        str++;
    }
}

void SPI0_init (void)
{
    // Initializes SPI0 for use in communicating with sensor PCB

    *pSPI0_BAUD = 6; // SCK = 100MHz/(2*SPI0_BAUD); 6 = 8.33MHz (4 = 12.5MHz)
    *pSPI0_CTL = 0x5D2D; // cpol=1, cpha=1, msb first, 16 bits, start when write SPI0_TDBR
    *pSPI0_FLG = 0x0060; // Enable SPI0SEL6 SEL5
    *pSPI0_FLG = 0x6060; // Enable SPI0SEL6 SEL5, set high (active low)
}

void SPI0_Convst(void)
{
    // Used to command AD7606 to begin a conversion
    // Includes ADC health checks with automatic restart of conversion if necessary

    int i = 0;
    int ADC_status = 0;
    while (!ADC_status)
    {
        delay_us(1);
        if (!(*pPORTEIO & 0x0040)) // Proceed only if busy is low; no pre-existing conversion
        {
            *pPORTEIO_CLEAR = 0x0008; // Clear convst
            delay_us(1);
            *pPORTEIO_SET = 0x0008; // Set convst (rising edge starts conversion)
            delay_us(1);
        }
    }
}

```

```

        if (!(*pPORTEIO & 0x0040)) // Verify busy went high, reset if not
            ResetADC();
        else
        {
            delay_us(350); // Wait 350us to allow for variations in conversion time
            if (!(*pPORTEIO & 0x0040)) //proceede if busy is low (done with conversion)
                ADC_status = 1;
            else
                ResetADC();
        }
    }

    else
        ResetADC();
}

}

int SPI0_Recieve1(void)
{
    // Used to recieve 16b number from AD7606

    *pSPI0_FLG = 0x2060; // Set SPI0SEL6 (!cs1) low (active low), leave SEL5 high (2nd AD7606 chip
disabled)
    *pSPI0_TDBR = 0xABAB; // Start transmission by sending dummy variable
    delay_us(1);
    while(!(*pSPI0_STAT & 0x0001)); // Wait for tranmission end (SPIF)
    *pSPI0_STAT = 0x0001; // Reset SPIF (spi finished) bit
    *pSPI0_FLG = 0x6060; // Set SPI0SEL6 high (active low), leave SEL5 high
    return *pSPI0_RDBR;
}

void delay_us(int delay)
{
    // Used to create delays in micro-second increments

    *pTIMER_ENABLE = 0x0002; //enable timer here for a constant delay - ensures timer not in mid cycle
when start delay

```

```

int i = 0;
while(i < delay)
{
    while(!(*pTIMER_STATUS & 0x0002)); // Waits for next interrupt
    *pTIMER_STATUS |= 0x0002; // W1C interrupt bit
    i++;
}
*pTIMER_DISABLE = 0x0002; //disable timer when completed delay
}

double FullCycleFilter(int adcValue, int X[], double Y[], int X_start, int Y_start, int ph)
{
    // Full cycle cosine filter attenuates power system harmonics and DC offset, passes 60Hz component
    // Returns mA estimate of primary current
    // Updates global variable V_AB_phase if ph=1 (used in ZCD)

    // 16 cosine filter coefficients can be reduced to 5 by cos symmetry, 4th coefficient=0 is omitted
    double cCFC[4] = {0.125,0.115484942,0.088388348,0.047835429}; // cCFC[4] = 0
    double Y_magnitude;

    X[X_start] = adcValue; // Append new input
    Y[Y_start] = 0; // Reset old filter output

    // Computes new filter output via MACs (multiply accumulates)
    switch(X_start)
    {
        case 0:
            Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8]) +
                cCFC[1]*(X[X_start+1]-X[X_start+7]-X[X_start+9]+X[X_start+15]) +
                cCFC[2]*(X[X_start+2]-X[X_start+6]-X[X_start+10]+X[X_start+14]) +
                cCFC[3]*(X[X_start+3]-X[X_start+5]-X[X_start+11]+X[X_start+13]); // cCFC[4] = 0
            break;

        case 1:
            Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8]) +
                cCFC[1]*(X[X_start+1]-X[X_start+7]-X[X_start+9]+X[X_start+15-16]) +
                cCFC[2]*(X[X_start+2]-X[X_start+6]-X[X_start+10]+X[X_start+14]) +
                cCFC[3]*(X[X_start+3]-X[X_start+5]-X[X_start+11]+X[X_start+13]);
    }
}

```

```

break;

case 2:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7]-X[X_start+9]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6]-X[X_start+10]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5]-X[X_start+11]+X[X_start+13]);
break;

case 3: //case 3 = case 4 because 4th coefficient=0
case 4: //x[12-16]*0 -> same as case 3
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7]-X[X_start+9]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6]-X[X_start+10]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5]-X[X_start+11]+X[X_start+13-16]);
break;

case 5:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7]-X[X_start+9]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6]-X[X_start+10]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5]-X[X_start+11-16]+X[X_start+13-16]);
break;

case 6:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7]-X[X_start+9]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6]-X[X_start+10-16]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5]-X[X_start+11-16]+X[X_start+13-16]);
break;

case 7:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7]-X[X_start+9-16]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6]-X[X_start+10-16]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5]-X[X_start+11-16]+X[X_start+13-16]);
break;

```

```

case 8:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8-16]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7-16]-X[X_start+9-16]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6-16]-X[X_start+10-16]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5-16]-X[X_start+11-16]+X[X_start+13-16]);
break;

case 9:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8-16]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7-16]-X[X_start+9-16]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6-16]-X[X_start+10-16]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5-16]-X[X_start+11-16]+X[X_start+13-16]);
break;

case 10:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8-16]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7-16]-X[X_start+9-16]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6-16]-X[X_start+10-16]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5-16]-X[X_start+11-16]+X[X_start+13-16]);
break;

case 11: //case 11 = case 12 because 4th coefficient=0
case 12: //x[4-16]*0 -> same as case 11
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8-16]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7-16]-X[X_start+9-16]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6-16]-X[X_start+10-16]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3]-X[X_start+5-16]-X[X_start+11-16]+X[X_start+13-16]);
break;

case 13:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8-16]) +
  cCFC[1]*(X[X_start+1]-X[X_start+7-16]-X[X_start+9-16]+X[X_start+15-16]) +
  cCFC[2]*(X[X_start+2]-X[X_start+6-16]-X[X_start+10-16]+X[X_start+14-16]) +
  cCFC[3]*(X[X_start+3-16]-X[X_start+5-16]-X[X_start+11-16]+X[X_start+13-16]);
break;

case 14:
Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8-16]) +

```

```

        cCFC[1]*(X[X_start+1]-X[X_start+7-16]-X[X_start+9-16]+X[X_start+15-16]) +
        cCFC[2]*(X[X_start+2-16]-X[X_start+6-16]-X[X_start+10-16]+X[X_start+14-16]) +
        cCFC[3]*(X[X_start+3-16]-X[X_start+5-16]-X[X_start+11-16]+X[X_start+13-16]);
        break;

    case 15:
        Y[Y_start] = cCFC[0]*(X[X_start]-X[X_start+8-16]) +
        cCFC[1]*(X[X_start+1-16]-X[X_start+7-16]-X[X_start+9-16]+X[X_start+15-16]) +
        cCFC[2]*(X[X_start+2-16]-X[X_start+6-16]-X[X_start+10-16]+X[X_start+14-16]) +
        cCFC[3]*(X[X_start+3-16]-X[X_start+5-16]-X[X_start+11-16]+X[X_start+13-16]);
        break;
}

// Computes phasor magnitude; calibrated factor = 0.169775107 [mA-rms/LSB] (assumes sinusoid)
if(Y_start+4 < 5)
    Y_magnitude = 0.1698*sqrt(Y[Y_start]*Y[Y_start] + Y[Y_start+4]*Y[Y_start+4]);
else
    Y_magnitude = 0.1698*sqrt(Y[Y_start]*Y[Y_start] + Y[Y_start-1]*Y[Y_start-1]);

// Computes phasor phase
if (ph)
{
    if(Y_start+4 < 5)
        V_AB_phase = 180/3.1415926*atan2(Y[Y_start+4],Y[Y_start]);

    else
        V_AB_phase = 180/3.1415926*atan2(Y[Y_start-1],Y[Y_start]);
}

return Y_magnitude;
}

int Trip_threshold(double Ym, double Y1, double Y2, double Y3, double Y4)
{
    // Checks whether inputs exceed minimum threshold
    // Returns 1 if threshold is exceeded, 0 if not

```

```

        if (Ym > TRIP_THRESHOLD || Y1 > TRIP_THRESHOLD || Y2 > TRIP_THRESHOLD || Y3 > TRIP_THRESHOLD || Y4 >
TRIP_THRESHOLD)
            return 1;
        else
            return 0;
    }

/*
//OVD disabled

void Timer2_init (void)
{
    // Configures Timer2 to 500kHz to generate software ISRs for OVD event verification

    *pTIMER2_CONFIG = 0x001D; // Emu run off, irq enable, pwm_out mode, output pad enable
    *pTIMER2_PERIOD = 200; // 500 kHz nominal period for 100MHz sclk - 2us interrupts
    *pTIMER2_WIDTH = 100; // 50% duty cycle (half of period)
    *pSIC_IMASK0 |= 0x00040000; // Enables at SIC level
    register_handler(ik_ivg11, Timer2_ISR);
}

void PinChange_init(void)
{
    // Configures hardware ISR for initial OVD event detection on main CBCT
    // Note PF7-11 set as input in ADC_init(), only set FER/DIR once

    *pPORTFIO_POLAR = 0xFFFF; // Sets PF inputs to active low/falling edge
    *pPORTFIO_EDGE = 0x0000; // Sets PF inputs to level sensitive
    *pPORTFIO_BOTH = 0x0000; // Sets edge sensitive inputs to single edge (not both)
    *pPORTFIO_MASKA_SET = 0x0080; // Sets interrupt mask A for PF7
    *pSIC_IMASK0 |= 0x00080000; // Unmasking at SIC level (for interrupt)
    register_handler(ik_ivg12, PF_ISR); // Assigning ISR to EVT 12: PFIsr is ISR name
}

EX_INTERRUPT_HANDLER(Timer2_ISR)
{
    // Interrupt triggers off Timer2 at 500kHz (every 2 us) after enabled by hardware pinchange interrupt

```



```

    // Checks the status of the PF7 pin (main OVD) for OVD event verification - pin stays low for 50us
for OVD
    // Simultaneously keeps track of Fdr1-4 OVD inputs for tripping logic selectivity
    // Note (*pPORTFIO & 0x0080) returns 1 if PF7 is low, as expected during OVD event

    *pTIMER_STATUS = 0x0004; // W1C the TIMIL2 bit; prevents endless looping of this ISR
    // *pPORTDIO_SET = 0x0004; // For monitoring total time spent in ISR timing purposes via logic
analyzer

    OVD_i++; // Following logic initializes with this at 1, as already triggered once in the HW ISR
    if (OVD_i < OVD_US_REQUIRED_FOR_EVENT)
    {
        // Keeps track of how many times the main goes high, with possible abortion
        if (!(*pPORTFIO & 0x0080)) // Check if main is actually high- should be low for actual OVD
event
    {
        OVD_j_m++; // count number of times is high - allow 5 to mitigate EMI
        if (OVD_j_m >= 5) // after 5 high occurrences decide there is no fault on main (and
therefore any feeder)
        {
            // No EVD event. Reset everything
            OVD_i = 0;
            OVD_j_m = 0;
            OVD_j_1 = 0;
            OVD_j_2 = 0;
            OVD_j_3 = 0;
            OVD_j_4 = 0;
            *pTIMER_DISABLE = 0x0004; // Disable Timer2 and this interrupt (reenabled after HW
interrupt triggers)
            *pPORTFIO_MASKA_SET = 0x0080; // Re-enable the HW pinchange interrupt
            OVD_initiated_trip = 0;
        }
    }

    // This keeps track of how many times Fdr1 is high (no possible abortion), for relay
selectivity
    if (!(*pPORTFIO & 0x0100))
        OVD_j_1++;

```

```

selectivity // This keeps track of how many times Fdr2 is high (no possible abortion), for relay
if (!(*pPORTFIO & 0x0200))
    OVD_j_2++;

selectivity // This keeps track of how many times Fdr3 is high (no possible abortion), for relay
if (!(*pPORTFIO & 0x0400))
    OVD_j_3++;

selectivity // This keeps track of how many times Fdr4 is high (no possible abortion), for relay
if (!(*pPORTFIO & 0x0800))
    OVD_j_4++;
}
else // Signal passes for an OVD fault
{
    *pTIMER_DISABLE = 0x0004; // Disable this interrupt immediately (opening the faulted breaker
will take up to 17ms)

    // Update the sense of all OVD inputs OVD_x
    OVD_M = 1; // By default the main has detected an OVD event

    if (OVD_j_1 < 5)
        OVD_1 = 1;
    else
        OVD_1 = 0;

    if (OVD_j_2 < 5)
        OVD_2 = 1;
    else
        OVD_2 = 0;

    if (OVD_j_3 < 5)
        OVD_3 = 1;
    else
        OVD_3 = 0;

```

```

        if (OVD_j_4 < 5)
            OVD_4 = 1;
        else
            OVD_4 = 0;

        OVD_initiated_trip = 1; // Sets the flag for the 960Hz tripping logic to take action
    }
    // *pPORTDIO_CLEAR = 0x0004;
}

EX_INTERRUPT_HANDLER(PF_ISR)
{
    // Hardware pinchange interrupt triggers off of PF7
    // This ISR enables a 2us periodc Timer2-based ISR for OVD event verification

    *pPORTFIO_MASKA_CLEAR = 0x0080; // Disables further hardware interrupts on PF7, reenabled in Timer2
ISR
    *pTIMER_ENABLE = 0x0004; // Enables checking of PF7 via Timer2 based interrupt
}
*/

```