ACCELERATED AGING AND HEALTH MONITORING OF IGBT(s) SWITCHES IN MOTOR DRIVERS

by

Paul Richard Valentine O'Connor

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Electrical Engineering

Charlotte

2014

Approved by:

_____
Dr. Robert W. Cox

_____
Dr. Yogendra Kakad

_____
Dr. Madhav Manjrekar

ABSTRACT

PAUL RICHARD VALENTINE O'CONNOR. Accelerated aging and health monitoring of IGBT(s) in Motor Drives. (Under the direction of DR. ROBERT W. COX)

There has been a proliferation in the use of power electronic drives in applications such as ships, aircrafts, and industrial machines. And as such, power electronic converters are playing a more crucial role in mission critical operations. Maintainability or extended operation, are often required during mission critical operations are thus becoming more valuable. The operation of these devices must be achieved with the utmost safety, reliability and repeat-ability and thus the need arises for these assets to be monitored. Diagnostic algorithms have been developed to use certain key performance indicators (KPIs) to detect incipient failures in electric machines and drives. And with new advancements in digital systems and cloud based analysis, it is becoming increasingly more likely that one can acquire the signals needed to perform incipient fault detection and offload these to computing platforms that can perform deep predictive analytics. It is with this knowledge base that this thesis proposes the development and implementation of a proof-of-concept offline incipient fault detection - OFID scheme for Insulated Gate Bipolar Transistors - IGBT(s).The thesis demonstrates a general algorithm that can be used for fault detection. The data acquisition process and the fault detection algorithm being tested on a working prototype AC drive are also presented. Considered also, is the creation of faults using accelerated aging schemes. With the OFID scheme proposed in this thesis, poof of viability of an online incipient fault detection scheme for IGBT(s) is given. And it is with this, potential benefits of lifetime extension of assets, improved controllability, safety and reductions in operating and maintenance costs can be attained.

DEDICATION

I would like to dedicate this thesis to my two guiding stars my Mom Adel E. O'Connor, my Dad Paul R. V. O'Connor Sr. and the rest of my family.

ACKNOWLEDGMENTS

I would first like to thank all the members of my committee for their assistance in helping me achieve this goal. I would like to emphatically thank Dr. Robert Cox for his work as my mentor, teacher and advisor. I greatly appreciate all the hard work he has put in order to help me become a better engineer. The success of this thesis would not have been, if not for his efforts. I would like to thank Stephanie for putting up with my constant confusion about all grad. School deadlines. To all my lab mates, I would like to extend my thanks, because my success is due in part your camaraderie. The simple tasks of brain storming with me, give me advice or simply just making me get out of the lab to clear my head. These acts of kindness often times gave me the motivation to continue and for this I give my thanks. Last but definitely not least I would like to thank mom Adel O'Connor, my dad Paul O'Connor, my brothers Peter and Demisco O'Connor and my girlfriend Nachella Neal. My whole success is in no small part due to you all. Thank you for the guidance, love and support that you all, always provide. I am forever indebted.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

The power electronics market is one of the world's most vibrant industry sectors, with some estimates projecting a compound annual growth rate of 7.74% from 2014 to 2020 [1]. This growth is often attributed to the fact that power electronics are a critical enabling technology, central to the performance of large and complex systems. Efficient and effective power conversion is essential to the adoption of renewables, the advent of electric propulsion, and the growth of "Big Data". Failures in such systems, can be catastrophic, and thus designers have developed schemes in which power converters are reconfigured once hard faults (i.e. shorts/opens) have been detected. If faults can be detected at an earlier or incipient stage, then reconfiguration can occur under conditions that allow for safer mode transitions. With new advancements in digital systems and cloud based analysis, it is becoming increasingly more likely that one can acquire the signals needed to perform incipient fault detection and offload these to computing platforms that can perform deep predictive analytics.

This thesis assumes that such trends will continue and thus proposes a fault detection (FD) system designed to exploit new digital systems to sense incipient faults in Insulated Gate Bipolar Transistors (IGBTs), which are the most common semiconductor switches in high power drives. The thesis itself reviews common IGBT failure mechanisms and demonstrates how fault indicators can be extracted using advanced systems. It also demonstrates a general algorithm that can be used for fault detection. The

data acquisition process and the FD alg. are tested on a working prototype AC drive. The creation of faults using accelerated aging schemes is also considered. The remainder of this chapter explains the motivation and, proposed approach of this thesis.

1.1 Motivation

Power electronics are critical to growth in a number of sectors, including, electric propulsion, power generation, and data centers. Failures in such systems can thus have a devastating impact, but failure mechanisms are only recently being understood. Two recent studies have attempted to catalogue failures in power controllers, with both showing that power device and associated soldering failures account for roughly 1/3 of all issues (See Figure 1.1). Power and thermal cycling are the primary cause, as most converters have dynamic loads that cause thermal expansion and contraction. This phenomenon stresses internal boundaries between material layers and ultimately causes devices to wear out. As the wear out occurs, electrical and thermal characteristics change. Voltage losses and therefore power losses also increase. Thermal resistances also rise, leading to higher junction temps. Such changes can be observed, but they are only slight and thus measurements can be difficult and expensive.

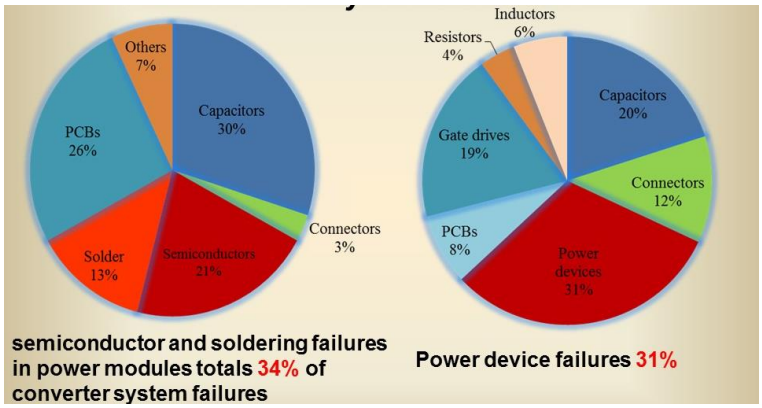


Figure 1.1: Pie charts outlining the Power converter failure distribution [85] [23].

Consider gate charge, for example, which can help to identify naturally occurring gate-oxide degradation [12], [13]. Figure 1.2 shows the noise corrupting the gate-current measurements in a 208V motor drive. Clearly, switching noise makes it difficult to consistently extract the injected gate charge. Another issue apparent in Fig. 1.2 is that the signals of interest change rapidly and thus must be sampled at very high rates and/or carefully conditioned using well-designed analog circuits. Given the noise issues, such sampling and conditioning must be performed near the switch. [2]

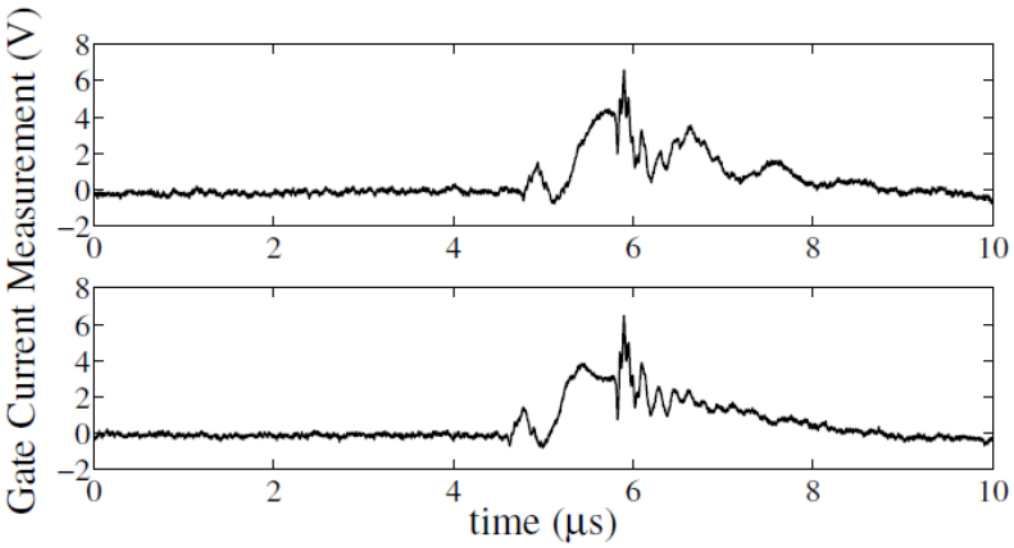


Figure 1.2: Output of a differential amplifier in a prototype drive measuring the voltage across a 10 gate resistor during two different turn-on instances. Note that the gate current has a slightly different shape during the two instances as a result of switching noise. [2]

To deal with the possibility of switching failures, big efforts have been put into fault tolerant drive systems. The two guiding principles are hard fault detection and active reconfiguration. In most such drives, faulty switching components are detected, and the appropriate switches are then isolated and disconnected. See Refs. [18] [19]. Reconfiguration thus happens only after hard faults such as open or short circuits have occurred. Although such approaches are desirable, they can still cause problems.

Consider, for instance, an IGBT that slowly wears over its lifetime because of thermal cycling. Ultimately, such a device will short circuit, which is desirable in a drive with N + 1 component. Eventually, however, the bond wires burn away, leading to an indeterminate failure state that may cause an arc flash and subsequent collateral damage to the rest of the circuit or to nearby humans [3].

Recent advancements in the use of digital signal processors (DSPs) and field-programmable gate arrays (FPGAs) close to the power semiconductor [14]-[17] provide the opportunity to overcome the aforementioned issues to perform early stage fault detection in real time. Intelligent FPGA-based controllers close to the switches have been used to measure switch terminal variables at very high speeds [16][17]. These measurements can be used to extract useful health-related features, including threshold voltage, gate charge, and on-state voltage and resistance [2].

Figure 1.3 shows a 3-∅, full bridge converter with the ability to sample all of the necessary terminal variables (i.e.V_{GE}, V_{CE}, I_C, and I_G). The key feature is the advanced gate-drive concept that has now been described in several works [14]-[17]. FPGA-based gate drives have been used to acquire switch terminal variables at rates as high as 100MHz. with emphasis on their use in optimizing turn-on and turn-off performance [16][17]. The signals measured by such devices can also be used to extract meaningful health-related features, such as on-state voltage and resistance. Feature extraction can be performed locally at the gate-drive unit, and features can be transmitted back to a central controller over fiber-optic cables at a much lower data rate [2].

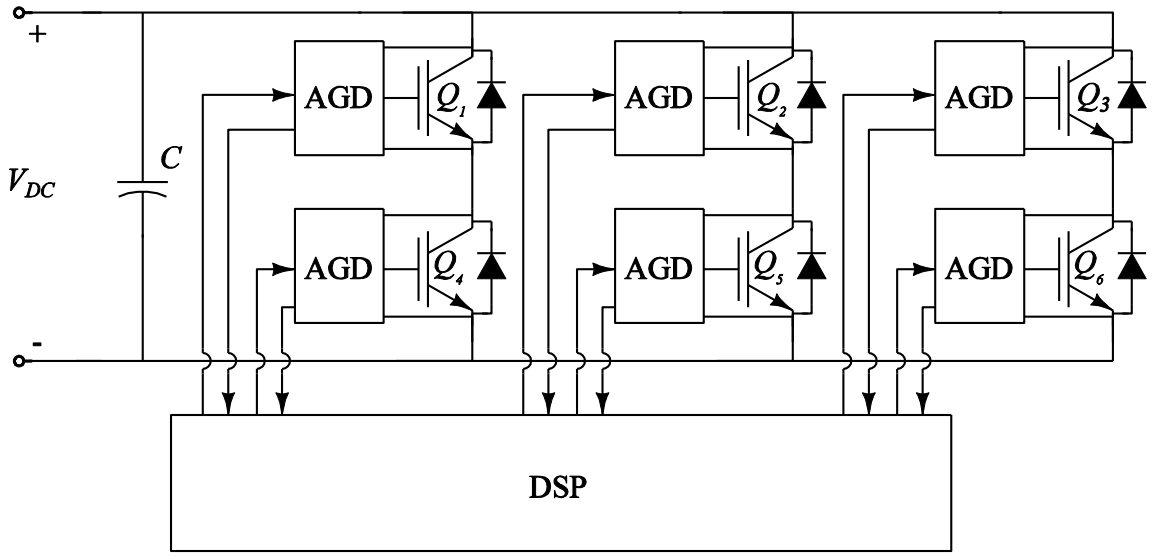


Fig. 1.3: A potential smart drive architecture including FPGA-based gate drives such as those discussed in [14]-[17]. These devices are labeled here as advanced gate drivers (AGDs). The AGDs provide the controller with appropriate measurements. These connections would be fiber optic [2].

Work in the area of advanced gate drives (see Refs [9]-[11]) continues to be focused heavily on real-time thermal modeling (see Refs [4]-[8]) and control of switching (i.e. *dv/dt* and *di/dt*). Although in their early stages, digital approaches can provide optimization between switching losses, reverse recovery current, and overvoltage at turn-off by carefully controlling gate current. Cost is a concern with such advanced devices, but their ability to act as a platform for diagnostics could potentially add to their value.

1.2 Proposed Approach

This thesis assumes that interest in advanced gate drives will only continue to grow, and that these devices should be exploited or early–stage fault detection. The focus is placed on an off-line proof-of-concept using the system shown in Figure 1.4. A custom 3-Ø motor drive was built and tested, and various sensing circuits were also constructed to focus on demonstrating the concept rather than engineering design, these measurement systems were not optimized for real time implementation, although other authors have

demonstrated the ability to monitor the necessary quantities in real-time systems. The focus here is placed on demonstrating the ability to detect faults using an algorithm adopted form the field of facial recognition. The necessary hardware and software are discussed in the later chapters of this thesis.

To demonstrate the proposed fault detection scheme, two specific early-stage faults are considered, namely latch-up and gate dielectric breakdown. Methods for inducing or simulating such failures were developed and used in chapter 5.

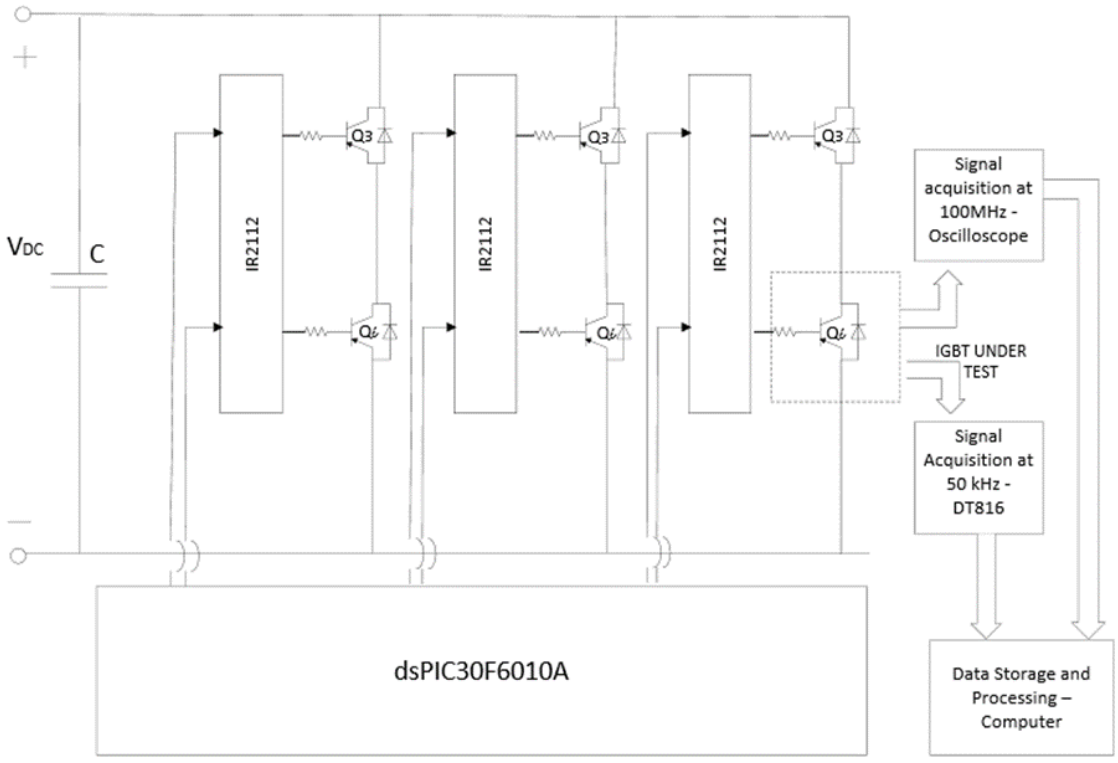


Fig. 1.4: A custom 3-Ø motor drive built and tested with various sensing circuits. This device was constructed to focus on demonstrating the off-line proof-of-concept.

1.3 Thesis Outline

The following chapters of this thesis describe the background methodologies and works pertaining to those ideas previously stated. Chapter 2 describes failure mechanisms affecting IGBT(s), chapter 3 describes the method for fault detection and diagnosis.

Chapter 4 discusses the system designed for testing, chapter 5 presents experiment results and chapter 6 includes conclusions.

CHAPTER 2: IGBT FAILURE MECHANISMS

This chapter describes various failure mechanisms associated with the IGBT, several of which are targets of the proposed health-monitoring algorithm. Since failures are directly related to the physical structure of the device, Section 2.1 first presents a brief description of the device's structure and operation. Section 2.2 then details intrinsic failure mechanisms that occur within the device itself, and Section 2.3 describes extrinsic faults associated with packaging and placement.

2.1 IGBT Structure and Operation

The Insulated Gate Bipolar Transistor (IGBT) is a three terminal power semiconductor device commonly used in high power switch-mode amplifiers and drives. Figure 2.1 shows its circuit symbol and indicates the names of each of its three terminals. The IGBT combines the characteristics of both the Bipolar Junction Transistor (BJT) and the Power Metal-Oxide-Semiconductor Field Effect Transistor (Power MOSFET). Specifically, the IGBT has input characteristics like a power MOSFET (i.e. high gate impedance), and output characteristics like the BJT. The IGBT thus acts like a voltage-controlled bipolar device. It was designed specifically for power switching applications, with the goal of being simple to drive (like a MOSFET) with a wide safe operating area (SOA) (like a BJT). Although far more information about IGBTs can be found in the literatures [21][24], it is essential for the purposes of this thesis to review the basic device structure and its principles of operation.

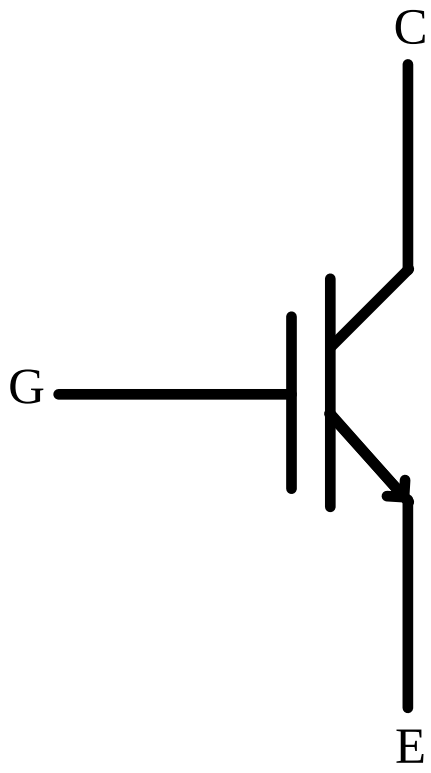


Figure 2.1: IGBT Circuit Symbol. Terminal connections are as follows, G - Gate, C – collector and E-Emitter.

The most common IGBTs are n-channel devices built using a Double Diffused Metal-Oxide-Semiconductor (DMOS) process. Figure 2.2 shows one of several common structures. The silicon cross-section is similar to that of a vertical power MOSFET, with a Metal-Oxide-Semiconductor (MOS) gate structure and N+ source regions embedded in P wells at the emitter. The P+ layer at the bottom is the drain or collector. The doping profile in each layer can be reversed to form a p-channel device, but these are far less commonly found in practice [21].

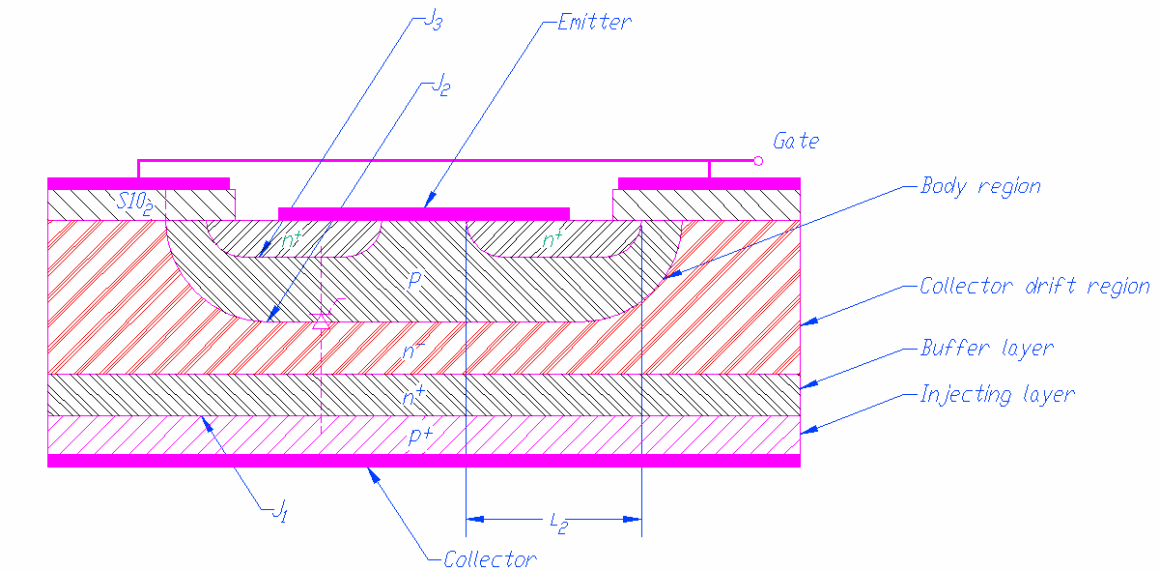


Figure 2.2: Cross section of a generic N-Channel IGBT. Each layer that is discussed in this chapter is labeled [24]. (From [21]).

The cross-section in Fig. 2.2 can be used to understand the operation of the device. When a positive voltage is applied between the collector and emitter terminals with the emitter and gate both connected to ground, the device enters the forward blocking mode. Although denoted as a forward operating region, no current is conducted between the collector and emitter terminals. As the gate voltage is increased above the emitter voltage, the p-type silicon below the gate (known as the base region) becomes inverted. A conduction channel forms, connecting the n^+ emitter to the n-type drift region. Electrons thus begin to flow into the drift region. This flow of electrons lowers the potential of the drift region, and the *pn* junction between the drift region the p^+ collector becomes forward-biased. Under this condition, a high density of minority carrier holes is injected into the drift region from the p^+ collector. Once the injected carrier concentration is much larger than the background concentration, a condition defined as a plasma of holes builds up in the n^- drift region. This plasma of holes attracts electrons from the emitter contact to maintain local charge neutrality. As a result, the

concentrations of excess holes and electrons gathered in the drift region are approximately equal. These excess concentrations drastically enhance the conductivity of drift region, phenomenon known as conductivity modulation [3]. More information about the forward blocking and conduction modes can be found in [21][24] [25][26].

When a negative voltage is applied across the collector-emitter terminals, the junction between the n^- buffer and P+ collector becomes reverse-biased and its depletion layer extends into the n^- drift region. The resulting breakdown voltage is determined by the open-base BJT formed by the p^+ collector, N- drift, and p-type base regions. The device is prone to punch-through if the N--drift region is very lightly doped. The desired reverse voltage capability can be obtained by optimizing the resistivity and thickness of the N - drift region [21].

The device described above is known as a punch-through (PT) IGBT. Devices without the n^+ buffer layer shown in Fig. 2.2 are known as Non-Punch Through (NPT) IGBTs. With a properly selected doping level and thickness, this buffer layer can improve device performance. Thought the IGBT's structure more closely resembles that of a power MOSFET, the conductivity modulation caused by the minority carrier injection means that its operation is more like that of a power BJT. Far more details about these phenomena can be found in various references [21]-[27].

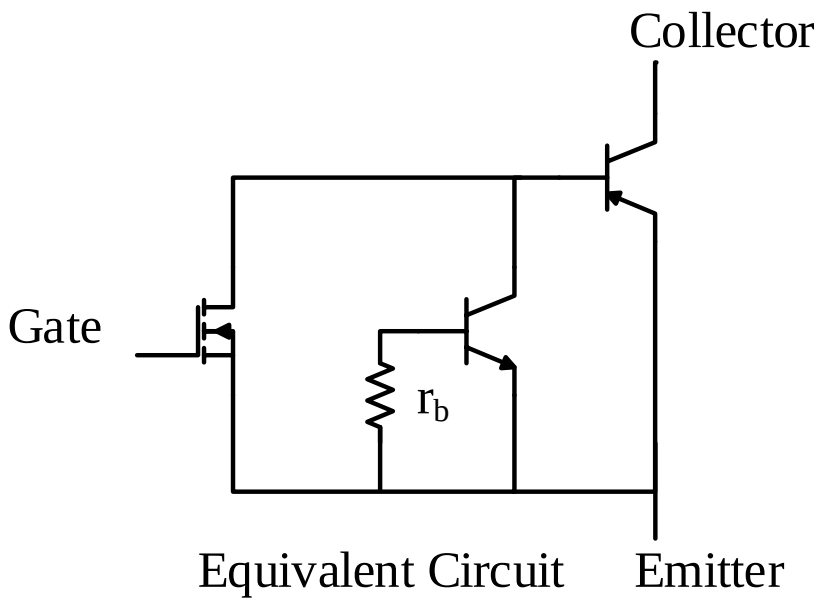


Figure 2.3: Equivalent circuit model for an IGBT.

A careful consideration of device operation yields the equivalent circuit model shown in Fig. 2.3. This circuit combines a shorting resistor *rb*, a MOSFET, and PNP and NPN transistors. The terminal called collector is actually the collector of a PNP transistor. The MOSFET drives the base of the PNP and determines the turn-on speed of the IGBT and its voltage drop The base of the PNP device is directly connected to the collector of the NPN transistor, and the base of the NPN device is connected to the resistor *rb* [84]. The NPN and PNP transistors form a parasitic thyristor. Recall that such devices have a regenerative feedback loop in which increasing device current only causes greater device current to flow, a phenomenon known as latch-up. The resistor *rb* represents the shorting of the base-emitter junction of the NPN transistor, which helps to prevent latch up. Turn-on of this thyristor can occur, and the result can be a device failure. More details about this model can be found in [21].

2.2 Intrinsic IGBT Failure Mechanisms

An intrinsic failure mechanism is one that results directly from an alteration of the semiconductor. Several important examples are covered in this section, namely latching, dielectric breakdown, time dependent breakdown, and electron migration.

2.2.1 Latching Failure

It was previously noted that the layering of the IGBT resembles that of a thyristor. This thyristor is parasitic and is normally prevented from turning on by limiting the current gain of the two transistors and reducing the value of the parasitic resistance R_B. During conditions when the collector current is particularly high (i.e. when the load is demanding excess current), problems can occur as the device is turning off. During this time, the MOSFET channel is reducing and thus higher than normal current flows through R_B. The excess voltage drop across this resistance increases the collector current in the NPN transistor, and more current is thus drawn from the base of the PNP device. The resulting positive feedback loop can drive the IGBT into a latch condition during which the device cannot be controlled from its gate. The collector current at which latch-up occurs is called the latching current. The magnitude of this value reduces with increasing device temperature. Hence, the IGBT is more susceptible to latch-up as temperature increases [27][29].

2.2.2 Catastrophic Dielectric Breakdown

Dielectric breakdown occurs when a strong electric field creates a current channel in an insulating medium. During conduction, breakdown can occur between the gate and collector terminals or between the gate and emitter terminals. Catastrophic breakdown of the gate oxide typically results from severe thermal or electrical over-stress (i.e.

electrostatic discharge, junction over-voltage, etc.) [30][2]. Key indicators of the breakdown are a lowering of the IGBT's thermal resistance and a corresponding rise in the junction temperature beyond safe operating values. Note that breakdown can also occur between the collector and the emitter when the device is in a blocking state.

2.2.3 Time Dependent Dielectric Breakdown

This a failure caused by chronic defect accumulation in the SiO_2 insulator during standard operation [13]. Such defects naturally result from a combination of at least three different mechanisms. These include impact ionization, hot carrier injection, and so-called trap creation attributed to the redistribution of hydrogen within the device. Before causing a complete failure, these naturally-occurring phenomena affect various device parameters. For instance, they can change the gate leakage current, a phenomena considered later in this thesis. Similarly, any charges that become trapped in the gate oxide affect the threshold voltage *VT* and the transconductance *gm* [13]. The deterioration of the silicon dioxide insulator also changes various device capacitances and thus changes device turn-on times [2].

2.2.4 Electron Migration

Electron migration is the transport of mass from metals when stressed at high current densities. This phenomenon occurs as the result of two different forces. The first is the electric field force, and the second is a force exerted on the metal ions in the direction of electron flow. This latter force is caused by the momentum exchanged as electrons collide into the activated metal ions [31]. Under such conditions, voids can form in the metal and the result is an increase in resistance or a complete open-circuit. This type of failure mechanism is dependent upon the occurrence of high collector

currents in the IGBT [2]. Key parameters be affected by such deterioration are the device resistance and on-state voltage.

2.3 Extrinsic Failure Mechanisms for IGBT

An extrinsic failure mechanism is one related to device packaging or placement. Some of the key failures explored here include wire bond lifting, solder-die detachment, and contact migration. Such failures are important to consider because dynamic thermal conditions are common in power-handling circuits. Such conditions cause solder and metal connections to be alternately expanded and contracted and thus weaken.

2.3.1 Wire Bond Lifting

Bonding wires are subjected to tensile stress as a result of power cycling [33]. As such cycling occurs, the junction temperature swings. Because the different materials within the device have different Coefficients of Thermal Expansion (CTEs), the adjacent materials expand and contract at different rates. Over time, the uneven rates of expansion and contraction can ultimately cause the bond wires to lift from the die and the die-attach layers to deteriorate. Held et al. concluded that power cycling is directly related to junction temperature and found that the number of cycles to failure can be characterized by the empirical relationship

$$N_f \propto \left(\Delta T_J\right)^{\alpha} e^{\frac{K_T}{T_J}} \qquad 2.1$$

where alpha and *kt* are real-valued constants, T_J is the actual junction temperature, and ΔT_J is the change in junction temperature [34].

Bond-wire lift eventually leads to higher junction temperatures (T_J), and therefore it affects parameters such as on-state resistance and on-state V_{CE}. Because increases in these parameters cause greater internal losses, they only act to further increase the

junction temperature. The result is a positive feedback loop that eventually leads to complete device failure [2].

2.3.2 Solder Die Detachment

Solder attaching the silicon die to the package heat sink can develop cracks and voids due to dissimilar thermal expansion in the two materials [36]. The junction-to-case thermal impedance thus rises, which causes a higher junction temperature. A positive feedback mechanism is thus created once again. As in the case of bond-wire failures, this mechanism affects parameters such as the on-state resistance and on-state V_{CE} [35][2].

2.3.3 Contact Migration

Contact Migration is related to electro-migration, which occurs when voids between external metal contacts and silicon cause metal to diffuse into the semiconductor. Ultimately, this diffused metal can short-circuit internal p-n junctions. Before causing a complete failure, this mechanism impacts parameters such as on-state resistance and on-state V_{CE} [35][2].

CHAPTER 3: METHOD FOR FAULT DETECTION AND DIAGNOSIS

This chapter briefly details the current state-of the art with respect to hard faults detection in IGBT drives, and then describes the proposed approach for detecting the early stage indicators that would prevent such faults occurring. Section 3.1 describes approaches for detecting the three most common hard switch faults, namely opens, shorts, and gate failures. Section 3.2 then focuses on the details of the proposed incipient fault detection scheme. Ideally, such a scheme would become the first line of defense for fault detection, and approaches such as the ones indicated in Section 3.1 would become a last resort.

3.1 Methods for Fault Detection for IGBT in Motor Dives

Voltage-source inverters (VSIs) are commonly used in motor drive and utility applications. It is estimated that about 38% of the faults in variable-speed ac drives, in industry are due to failures of power devices [40]. Even though drives can use a multitude of semiconductor switching devices, IGBTs are mostly used due their capability for high voltage and current ratings and their capacity to withstand short-circuit currents for periods exceeding 10 μs [41]. Though IGTB(s) are known for their sturdiness, they are not immune to failure. These failures are often due to excess electrical and thermal stress which arises in various applications. IGBT failures can be broadly categorized as open-circuit faults, short-circuit faults, and intermittent gate misfiring faults. And for these categorizations there have been, the development of various protection methods. The rest

of this section will then proceed to define each fault's category and catalog the protection diagnostic/detection methods used for them.

3.1.1 IGBT Open Circuit Faults

A gate-drive open-circuit fault may happen due to lifting of bonding wires caused by thermic cycling. It may be caused by a driver fault or a short-circuit-fault-induced IGBT rupture. Open-circuit faults lead to dc current offset in both the faulty and healthy phase. The interaction between the dc component and the field generates a pulsating torque at the stator current frequency, which may substantially reduce the maximum average torque available to the drive [42]. Unequal stress is also generated in the upper and lower transistors due to the dc currents. These effects may cause secondary faults in the inverter, motor or load. The voltages and currents carry the fault signatures and hence can be analyzed to detect and locate the fault. Open circuit faults generally do not cause system shutdown, but degrade its performance [41]. Listed in Table 3.1 are the diagnostic methods that can be used for open fault detection as well as the literature in which can more details can be found. Also listed are particulars about effectiveness, resistivity, detection time, implementation time, detection parameter and threshold dependence on detection variable with respect to each other.

Table 3.1: Open circuit fault detection methods from [41].

Methods	**Effectiveness**	**Resistivity**	**Detection Time**	**Implementation Time**	**Timing Effort**	**Detection Parameter**	**Threshold Dependence on Detection Variable**
Park's Vector Method [43]	Ambiguous at small circuits	Poor at small currents	>20 ms	Medium	High	3-phase currents	High
Normalized DC Current Method [44][45]	Poor at small currents	Poor as multiple conditions may	18.4 ms	Low	Low	3-phase currents	Independent

		satisfy					
Modified Normalized DC Current Method [45]	Good	Good	18.4 ms	Low	Low	3-phase currents	Independent
Slope Method [46]	Poor at small currents	Poor	38.3 ms	Low	High	3-phase currents	High
Simple DC Method [44][45]	Poor at small currents	Medium	Load dependent	Low	Medium	3-phase currents	High
AC Instantaneous Frequency Method [46]	Cannot locate faulty transistor	Medium	20 ms	Low	Low	3-phase currents	Low
Comparison of Actual and Reference Quantity Method [40][47]	Needs, 2-4 sensors for effective detection	Good	5 ms	High	Medium	Inverter pole voltage, phase voltages and neutral voltage	Independent
Sensing Voltage across the lower Switch [48]	Good	Good	2.7 ms	Medium	-		Low
Centroid Based Fault Detection [49]	Good	Good	-	Medium	-	3-phase currents	N/A
Current Pattern Recognition Method [50]	Many parameters to consider	Medium	10 ms	High	Medium	3-phase currents	High
Converter Behavior Rules [51]	Not yet proven online	Good	-	High	None		Independent
Spectrum Analysis Method [52]	-	-	-	Medium	High	3-phase currents	High
Current Deviation Method [53]	Good	Medium	2 cycles	High (Requires change in inverter control algorithm)	Low	Normalized 2-phase currents	Independent
Wavelet – Fuzzy Method [54]	Good if the fuzzy rules are carefully designed	Good	5 cycles	High	Medium	3-phase currents	Low
Wavelet-Neural Network	Diagnosis error < 5%	Good if NN is trained	-	High due to NN training	Low	3-phase currents	N/A

Method [55]		thoroughly					
Wavelet-ANFI Method [56]	-	Good as the NN is trained for noisy conditions	-	High due to NN training	Low	DC link current	N/A
Clustering-ANFI System [57]	Worst- error <2.26%	-	-	High	Low	3-phase currents	N/A
Subtractive Clustering Based Mean Current Vector Method [58]	Good	-	¼ cycle	Medium	Low	3-phase currents	N/A
Model Based ANN Method [59]	Prediction rate of around 75%	-	-	High	-	3-phase voltages, currents and torque	N/A
Rules Based Expert Systems [20][21]	Offline diagnosis tool	-	-	High	-	User Input	N/A
Bond Graph Method [22]	-	-	-	High	-	Switch voltages	High

3.1.2 IGBT Short Circuit Faults

A transistor short circuit in the motor drive system causes a nonzero dc component current flow in the stator winding and the machine experiences dynamic braking with other base drives disabled. In many short-circuit failure conditions; the time between the fault initiation and the device failure is very short. The IGBTs can withstand abnormal currents up to around 10 μs. Therefore, detection and protection should happen dynamically with the changing conditions. Detection methods based on mathematical transformations can detect the damaged device, but they are not fast enough to cause any protective action. Other methods that are based on voltage level change are fast but need too many measurements for fault localization [63][41]. Detection methods that have been

used for detection of short circuit faults are listed in Table 3.2. For a more expansive look in to each method the literature review for each is provided in the first column beside the identified method

Table 3.2: Short circuit fault detection methods from [41].

Methods	Parameters Requirements	Turn-off	Dynamic	Implementation Effort	Reliability	Comments
De-saturation detection Method [64]	Collector voltage	Abrupt	No	Low	Medium	Device turn-off not assured
Current Mirror Method [64]	Device current	Abrupt	No	Low	Medium	Very expensive
Gate Voltage Sensing Method [63]	Gate voltage	Abrupt	No	Low	Low	Requires complicated protection circuitry
Di/dt Feedback Control Method [64]	Device voltage	Soft	Yes	High	Medium	Stray inductance hard to control
Protection by Gate Voltage Limiting [65]	Gate voltage	Soft	Yes	Medium	Low	Interferes with normal operation
Protection by Snubbers and Clamp Circuits [66]	Device voltage	N/A	Yes	High	Low	Very expensive for fault protection
Protection by Slow Turn-off Using Additional Parameters [66]	Gate voltage	Soft	Yes	High	Low	Require complex circuitry
Protection by Two-Step Gate Pulse [67]	Gate voltage	Soft	No	High	High	Reliable shutdown of the device, di/dt
Average Current Park's Method [43]	Phase currents	N/A	N/A	Medium	High	Reliable detection but offer no protection
Vector Composition of the Inverter Output Voltage [68]	Inverter output voltage	N/A	N/A	Medium	Low	Slow response as it uses filters

3.1.3 IGBT Intermittent Gate-Misfiring Faults

A catastrophic breakdown of the device can be caused by IGBT gate-misfiring faults, if the faults remain undetected. It may be a result of driver open circuit, control circuit element deterioration, degraded electromagnetic compatibility, etc. Inverters can operate for a considerable period of time even with sustained gate misfiring. However, this is accompanied by degraded output voltage and overstress on other switching devices and dc bus capacitors. Gate misfiring may also lead to a short-circuit fault of a device. This may happen when the device fails to turnoff causing shoot-through. In most cases, gate misfiring is intermittent in nature. Online monitoring is important in safety-critical applications. Detection methods in frequency domain are not suitable for intermittent faults, since the steady-state measurements do not contain information about the fault after the system has recovered. Therefore, time-domain techniques are desirable. Intermittent gate-misfiring fault has not been extensively studied in the previous literature [41].

Two approaches that are used to detect this type of fault are Detection by Output Current Trajectory and Pattern Recognition Approach. The methodology behind Current Trajectory approach is as follows. When a gate-misfiring fault occurs in one of the inverter switching devices, the voltage disturbance will cause an increment to the stator current space vector. The incremental current happens toward a unique direction that is determined distinctively by the failed device. An incremental system model is used to provide compensation to the measured current response, so that the modified incremental current signal will decay in the opposite direction to the initial offset caused by the

voltage disturbance. The trajectory of the current response can be used to detect the fault and locate the defect switch [69] from [41].

The Pattern Recognition Approach is a fuzzy logic based technique and that inverter output current to detect the intermittent loss of firing pulses. A fuzzy fault detection and diagnosis block compares the stator Concordia current vector with that in the rule base. By this process, the method can detect a fault, locate the faulty device and can also find the fault severity. The method is discussed more in [70] and [71] from [40].

3.2 Algorithm Development

Although the algorithm described previously are desirable for enabling drives to operate in emergency situations, it is desirable to detect the underlying root causes much earlier so that performance degradation or potential root catastrophic danger could be avoided. This Section describes our solution to this problem, and it is based heavily on the failure modes and effects work done by several other researchers [[16]-[20]]. Figure 3.1 shows the proposed health-monitoring scheme. Inputs include various raw sensor signals or terminal variables. The inputs are the current (I_C), the voltages (V_{CE} $V_{CE,ON}$ V_{GE}), the ambient temperature (T_A), and the case temperature of the IGBT (T_C). The first step is to extract relevant key-performance indicators (KPIs) or features. Examples include the root mean squared (RMS) value collector current (I_C), the threshold voltage *VT* (gate voltage at which the IGBT turns on and collector current begins to flow), $V_{GE,ON}$ and other features which with are competed by the various manipulations of the terminal signals.

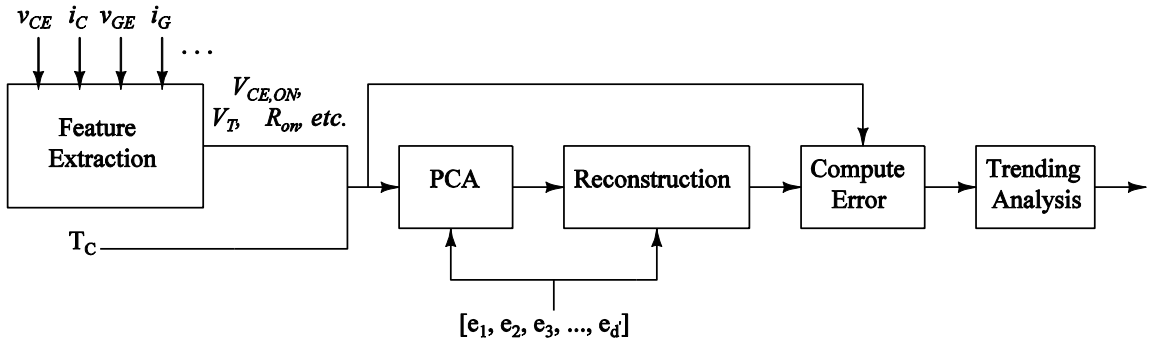


Figure 3.1: Generalized condition-monitoring algorithm for an operational IGBT based on principal-components analysis (PCA). The $\boldsymbol{e_i}$ are based on healthy conditions. New input signals and parameters can be added as needed. A similar implementation would be used for a MOSFET form [72].

This extraction is done in the feature extraction block which contains the raw signal manipulations needed in order to output the KPI(s). The process of each feature's extraction is detailed in section 4.3. It is to be noted that the terminal signals must be sampled at a high rate due to the fact that the extracted features are processed at a much slower rate. This is mostly due to the fact that multiple data samples are usually required for processing. For this experimentation the terminal signals are sampled at both 100 MHz and 50 KHz but are processed at much lower rate.

The crux of the fault-detection algorithm is the principal-component analysis (PCA) block. PCA casts high-dimensional data onto a lower dimensional space, which implements a transformation that best represents the information in the overall data set. In other terms, PCA aggregates a set of KPIs and provides information about the germane trends without having to develop a set of rules a priori. Experience has shown that that PCA leads to a robust detection mechanism [72]. The PCA block then relays information about relevant trends within the monitored data to the reconstruction block.

In order to provide such information, the features measured at each time step t_k are first grouped into a column vector x_k. The length of this vector is equivalent to the number of features; for generality, we say it has length *d*. The basic approach of the principal component- based algorithm is to compare each measurement x_k to an expectation. This expected vector is computed by projecting x_k onto a vector space created using "healthy" features. The healthy values are learned during a training phase in which the motor is assumed to be fault-free. During training, the healthy vectors are decomposed into a small set of characteristic vectors that best describe the distribution of the healthy parameters. During operation, each measured vector is projected onto this space. In the language of information theory, we are extracting the relevant information from the feature vector, encoding it efficiently, and then comparing the encoded result to a database of healthy features encoded in a similar manner. Any differences indicate that a fault may be developing. This approach is partially patterned after the facial-recognition scheme presented in [72].

The training space includes feature vectors recorded under normal operating conditions. In all, there are M such feature vectors and they are denoted as Γ 1, Γ 2, Γ3, … , Γ M. These training vectors are subject to a principal-component analysis in which one seeks a set of orthogonal vectors e_i that best describe the distribution of the data. The j−th training vector can thus be expressed as

$$\Gamma_j = \mathfrak{m} + \sum_{i=1}^{d'} a_{j,i}\boldsymbol{e}_i \qquad 3.1$$

where m is the sample mean, i.e.

$$\boldsymbol{m} = \frac{1}{M}\sum_{j=1}^{M}\boldsymbol{\Gamma}_j \qquad 3.2$$

Note that the distribution of the data is best described using $d' \leq d$ orthonormal vectors [[74], [75]]. During the training phase, one calculates these vectors by minimizing the squared-error criterion function

$$J = \sum_{j=1}^{M}\left\|\left(m + \sum_{i=1}^{d'} a_{j,i}\boldsymbol{e}_i\right) - \Gamma_j\right\|^2 \quad 3.3$$

[75] and [74] show that the $\boldsymbol{e_i}$ correspond to the eigenvectors of the sample covariance matrix which is

$$\boldsymbol{S} = \sum_{j=1}^{M}\left(\Gamma_j - \boldsymbol{m}\right)\left(\Gamma_j - \boldsymbol{m}\right)^T \quad 3.4$$

The actual e_i are the eigenvectors corresponding to the d′ largest eigenvalues of S [[74], [75]]. Ultimately, this process yields a compact basis that efficiently encodes the relevant features of a healthy motor during normal operating conditions. Further details on the PCA algorithm are included in various references, including [74] and [75].

When the motor is operational and the algorithm is applied to monitor its health, the first step is to project the features measured at time t_k onto the space spanned by the e_i. This projection is performed by the block labeled PCA, which computes the coefficients

$$a_{k,i} = \boldsymbol{e}_i^T(\boldsymbol{x}_k - m) \qquad 3.5$$

The next block uses these coefficients to reconstruct an approximation of x_k. The resulting estimate is thus denoted as

$$\hat{\boldsymbol{x}}_k = \boldsymbol{m} + \sum_{i=1}^{d'} a_{k,i}\boldsymbol{e}_i \quad 3.6$$

Following reconstruction, the algorithm calculates the two-norm of the residual vector r = $\hat{\boldsymbol{x}}_{\boldsymbol{k}} - \boldsymbol{x}_{\boldsymbol{k}}$. This quantity, which we term the health indicator (HI), is computed at each time tk. HI is thus

$$\mathrm{HI}_k = \mathbf{r}^T\mathbf{r} = (\hat{\boldsymbol{x}}_k - x_k)^T(\hat{\boldsymbol{x}}_k - x_k) \quad 3.7$$

This quantity represents the error between the measured features $\boldsymbol{x}_{\boldsymbol{k}}$ and their expected value for a healthy motor. In a very basic sense if the error is small, the motor is operating under normal conditions; if the error grows, a problem may be developing. The final block, labeled "Trending Analysis," monitors for such variations [72].

On its face, this mathematical derivation of the principal components seems unintuitive. Thus it would be best if a step back was taken in order to frame the PCA in terms of the problems at hand. Consider each vector $\boldsymbol{x}_{\boldsymbol{k}}$ as a set of 8 KPIs recorded at intervals of 3 seconds. In general, its unknown if the machine has a fault, and therefore what is indicated by $\boldsymbol{x}_{\boldsymbol{k}}$ is unknown. One could develop a set of rules that would monitor each of these 8 values. But in a rules-based context it would mostly likely be assumed that as long as each feature is individually important as a sort of fault indicator, it would be more prudent to simultaneously monitor all 8 KPIs. While it may seem natural to think that each feature possesses independence, this is not usually the case and as such classification accuracy does not improve with the monitoring of additional features. There can be a point diminishing returns due to the fact that the more features are added the more convoluted the information being acquired becomes and as such the more difficult it becomes to make meaningful sense in terms of creating a set of rules that can envelop all possible trends in the data .

The health monitoring algorithm combines the features/KPIs in such a manner that it compresses the size of the feature set and presents the trends that best highlights the difference in the features. This is demonstrated in chapter 4. It should be noted that the there is an effect on the Health Indicator due to loading conditions of the motor. As the load on the motor changes, there is an effect on other quantities such as The IGBTs collector current. This point will also be explored in chapter 5.

CHAPTER 4: SWITCH CONDITION MONITORING OF IGBT(s)

This chapter focuses on the experimental setup developed to test the algorithm discussed previously. Section 4.1 provides an overview of the complete experimental system. Section 4.2 describes the drive architecture and sensing circuits for raw signal extractions. Section 4.3 details the process used in software to extract the features needed by the algorithm. Section 4.4 describes the methods used to degrade several IGBT(s). These degraded IGBT(s) were placed in the drive during the testing described in Chapter 5.

4.1 System Setup Overview

As previously stated in the Section 1.2, this thesis aims to produce an offline IGBT incipient fault detection scheme based on PCA which can then be upgraded to an online method. The ultimate goal is to create an electronic motor drive using an advanced FPGA based controller (see Figure 1.3) with the diagnostic ability to sample key parametrics of the IGBT. In order to implement the proof of concept, for the offline incipient fault detection in IGBT switches in motor drives. A system comprised of the combination of custom designed hardware setup and custom software was developed. Figure 4.1 illustrates a high level block diagram of the systems structure. The system in its entirety is made up of custom built motor drive architecture with both sensing circuits and an oscilloscope for data measurements. The data measured by the sensing circuits are

sampled by the data acquisition device the DT9816. The sampling of data by both the DT9816 and the oscilloscope are controlled via coding developed in MATLAB software. The data is then passed to the computer with the MATLAB which then handles the data collection and processing by means of PCA in the MATLAB environment.

4.2 Drive Architecture

Figure 4.1 shows a three-phase full-bridge motor drive that has been constructed in a way that makes it is possible to sample all of the required terminal variables. The key feature is the advanced gate drive concept from [14]. The raw signals that are measured are as follows; the phase current (I_C), the collector-to-emitter voltage (V_{CE}), the collector-to-emitter on-state voltage ($V_{CE,ON}$), the gate-to-emitter voltage (V_{GE}), case temperature (T_C), and ambient temperature (T_A).

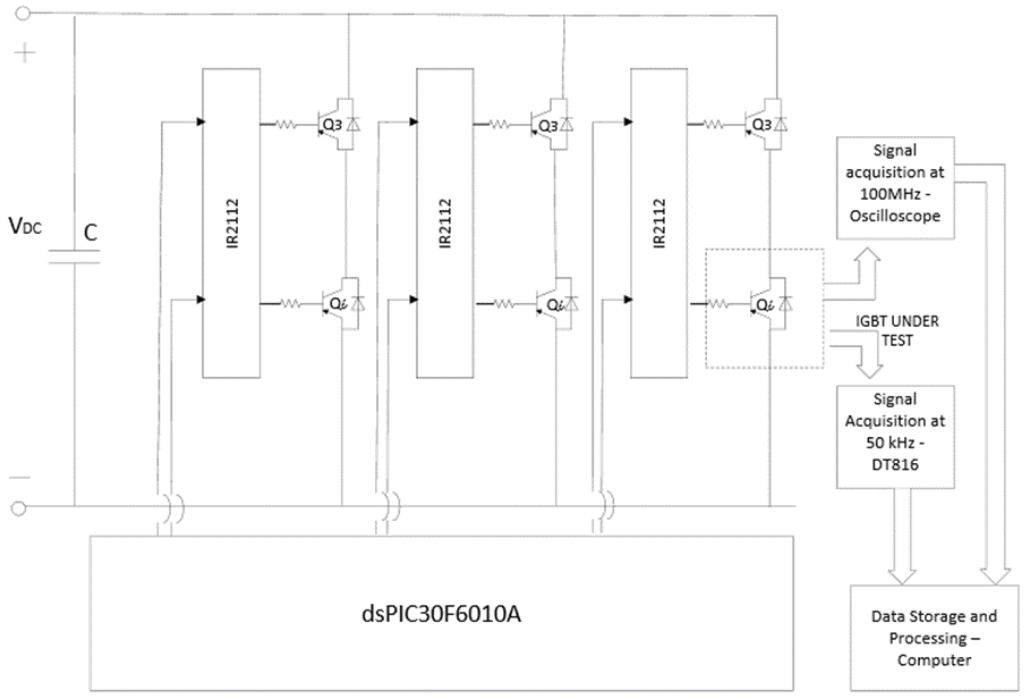


Figure 4.1: Schematic of custom-built IGBT based motor drive used for driving a 0.5hp induction motor. The control signals are supplied using a dsPIC30F6010A.

From a health monitoring point of view, it is paramount to sample raw signals such as $V_{CE,ON}$), and I_C fast enough to be able to extract features such as VJ and R_{ON}. Sampling rate is one issue that must be addressed regardless of the actual circuit architecture. Practically speaking when sampling, the sample rate must be sufficiently above the switching frequency. This in turn may pose a limitation in real world applications dealing with lower powered drives with switching frequencies at 10 kHz or higher. This would have to be overcome by using powerful microcontrollers or Field Programmable Gate Array - FPGA boards with a sampling speed in excess of 100 MHz with at least 8 bit resolution. In high power, medium voltage (MV) drives, which are more likely to require condition monitoring and power-dissipation limits, tend to cap switching frequencies at values on the order of 1 kHz [72] [75].

Given that the focus of this thesis is on demonstrating the algorithm, there were not sufficient resources available to address the basic issues associated with implementing an FPGA based drive. Instead, a reasonable hybrid technique was developed and applied to a traditional three phase drive architecture. (See literature [72] for complete catalog of drive development). Figure 4.1 shows the complete block diagram. Note that features are requiring low sampling rates are acquired using a low-cost 16-bit USB data acquisition board [76]. Features requiring high frequency sampling are acquired using a Tektronix 3014b. This device has a sampling rating of 1.25 GS/s with a bandwidth of 100MHz.

4.2.1 50 kHz Sampling for Switch Condition Monitoring

Measurement circuits, sensors, differential probes and current guns are used to measure the raw signals that are used to extract the relevant features. First to be discussed

will the sensing circuits and sensors used to get the raw signals that will be fed to the 50KS/s DT9816 data acquisition board. The phase current, I_C. is measured using the LTS 6-NP which is a Hall-effect transducer. The collector to emitter voltage V_{CE} is measured in the on-state only, due to difficulty in measuring said signal in its entirety. V_{CEON} is measured using a desaturation detection circuit included in gate drives to detect if an IGBT is no longer in the saturation region of operation. The reason for such an approach is due to the fact the V_{CE} is difficult to measure directly because the amplifier required would be exposed to common-mode swings on the magnitude of several hundred volts and thus analog-to-digital converter would have difficulty measuring the very low on-state voltage with adequate resolution.

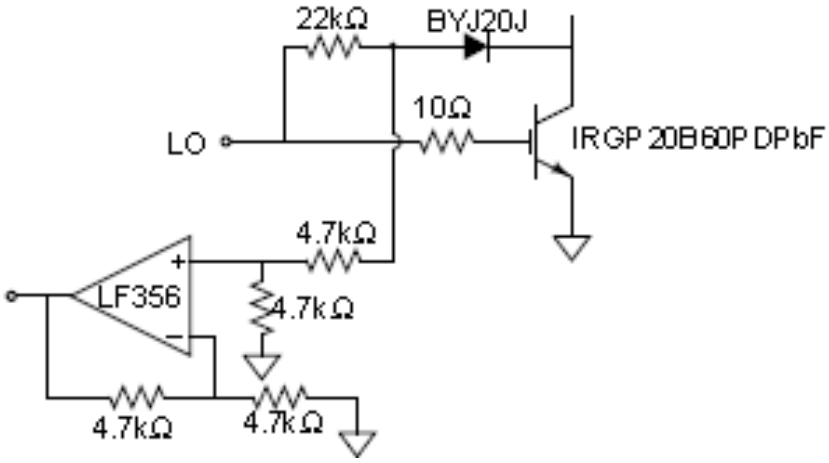


Figure 4.2: Circuit used for the measurement of on-state collector-to-emitter voltage, $V_{CE,ON}$ in experimental drive from [72].

$V_{CE,ON}$ is measured by the circuit shown in Fig. 4.2. The voltage *Vx* is referred to the high voltage side of the drive and is connected to a simple amplifier subtractor circuit. Once the IGBT is conducting, *Vx* has the definition

$$v_X = V_D + v_{CE}(i_c) \quad 4.1$$

where *VD* is the forward voltage drop across the diode in the desaturation circuit. Since the voltage at the output of the differential amplifier in Fig. 4.2 is referenced to the high voltage bus, the circuit shown in Fig. 4.4 is used to transmit its output to the data-acquisition system. Similar isolation would be required in an FPGA-based design, but its exact location in the signal path would depend upon the specifics of the design [72].

The optical transmission circuit from [14] maintains appropriate isolation when measuring $V_{CE,ON}$.. The main component of the optical transmission circuit is the HCNR201 which is a high-linearity wide-bandwidth analog opto-coupler consisting of a high-performance AlGaAs Light Emitting Diode, LED that illuminates two closely matched photodiodes, PD1 and PD2. The input photodiode, PD1, can be used to monitor, and therefore stabilize, the light output of the LED. The non-linearity and drift characteristics normally associated with LED can be virtually eliminated. The output photodiode, PD2, produces a photocurrent that is linearly related to the light output of the LED. This allows for a highly stable linear gain characteristic of the overall optical transmission circuit [72], [77].

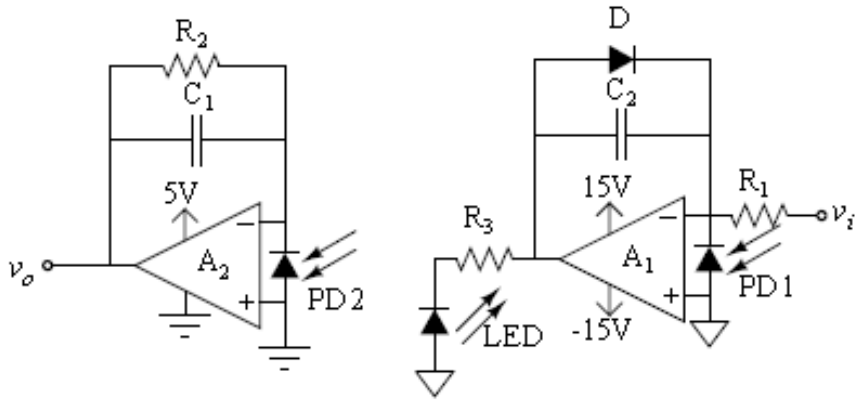


Figure 4.3: Analog circuit used to transmit signals from across isolation barrier.

The operational amplifier A1 adjusts the LED current, and therefore the current in PD1, IPD1, to maintain its + input terminal at 0 V. Since the + input of A1 is at 0 V, the current through R1, and therefore IPD1 is [77]

$$I_{PD1} = \frac{v_i}{R_1} \qquad 4.2$$

IPD1 depends only on the input voltage and the value of R1 and is independent of the light output of the LED.

The relationship between input optical power and the output current of the diode is linear. Thus by stabilizing and linearizing IPD1, the light output of the LED is also stabilized and linearized. Since light from the LED also falls on both of the photodiodes, IPD2 is stabilized as well. The physical package construction determines the relative amounts of light that fall on the two photodiodes and, therefore, the ratio of the photodiode currents. This results in very stable operation over time and temperature. The photodiode current ratio is expressed as a constant, K, where [72], [77]

$$K = \frac{I_{PD2}}{I_{PD1}} \qquad 4.3$$

The operational amplifier, A2 and resistor R2 form a trans-resistance amplifier that converts IPD2 back into a voltage, v_o, which is [72], [77]

$$v_0 = I_{PD2} R_2 \quad 4.4$$

Combining Eq. (4.2)-(4.4) yields a transfer function relating the output voltage to the input voltage,

$$v_o = K\left(\frac{R_1}{R_2}\right) v_i \qquad 4.5$$

Thus the relationship between $v_i$and v_o is constant, linear, and independent of the light output characteristics of the LED. The optical transmission circuit's gain can be adjusted

by simply changing the ratio of R2 to R1. The parameter *K* can be thought of as the gain of the opto-coupler and for this HCNR201 it is approximately 1 [72] [77].

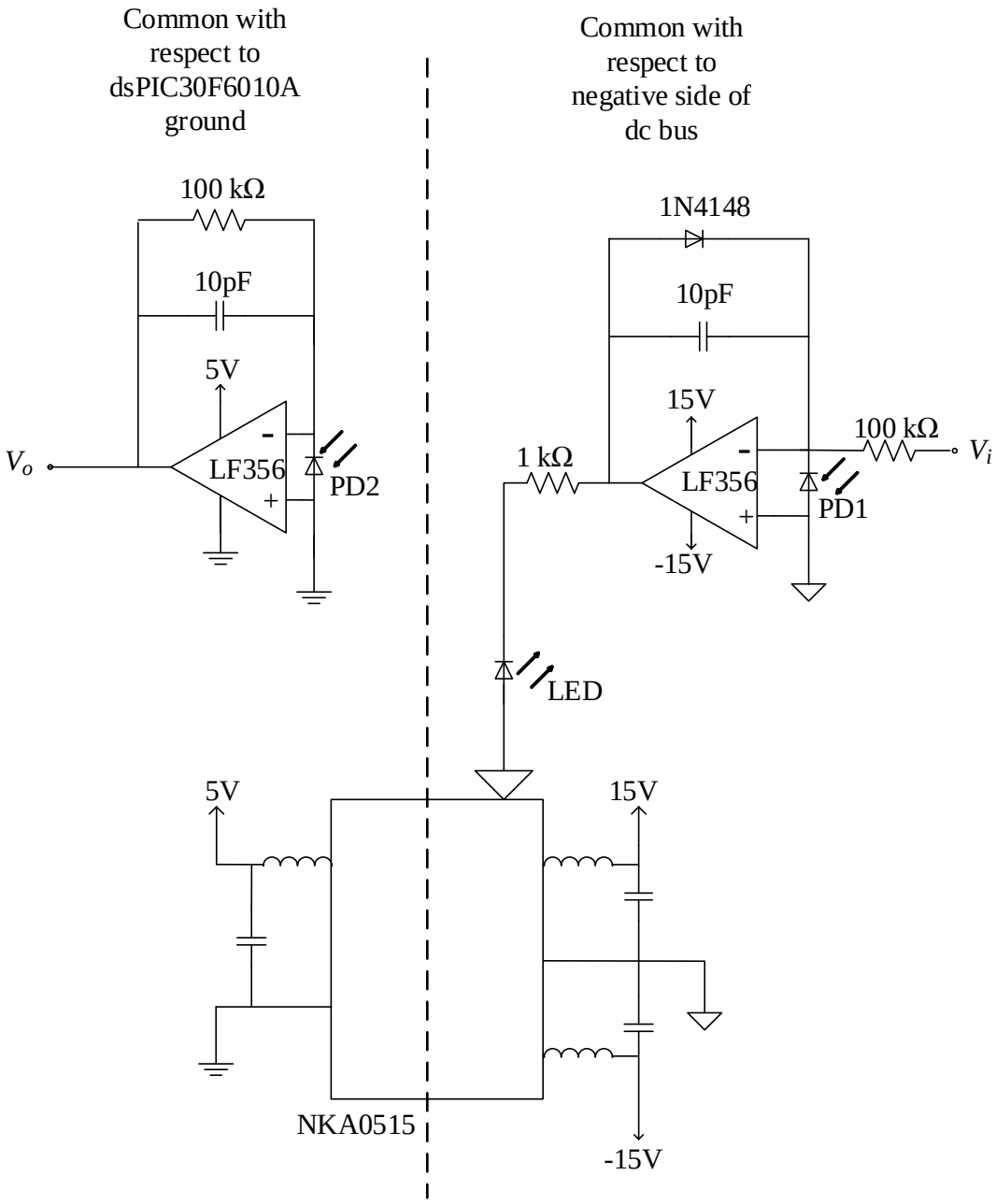


Figure 4.4: Optical transmission circuit used to provide isolation between the low voltage data-acquisition system and measurements recorded with respect to the high voltage bus. The NKA0515 provides isolated dc supplies for all of the measurement circuits included here.

Figure 4.4 shows the unity gain optical transmission circuit that was designed to measure $V_{CE,ON}$. Each of the measurement circuits that are referenced to the dc bus of the motor drive requires an optical transmission circuit to transmit the signal back to be acquired using the data acquisition board [72].

Figure 4.5 shows the circuit measuring the gate-to-emitter voltage, which is comprised of just a differential amplifier. A voltage divider was used to reduce the size of gate voltage. This is done, so that the output from the operational amplifier would be in a range so that a circuit similar to the one in Fig. 4.4 is able to transmit the output back to the data-acquisition system. For the computation of $V_{GE,ON}$, samples are averaged over each conduction interval for a given interval. A solid-state sensor measures the ambient temperature, and a thermocouple measures the case temperature as shown in Fig. 4.6. These measurements are also averaged over an interval of 1 second.

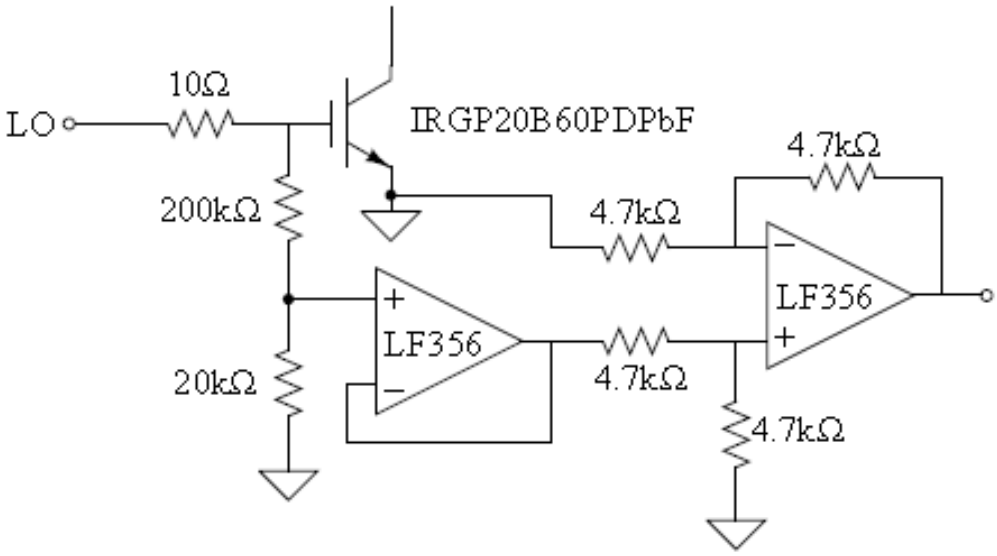


Figure 4.5: Circuit used for the measurement of gate-to-emitter voltage, V_{GE} in experimental drive.

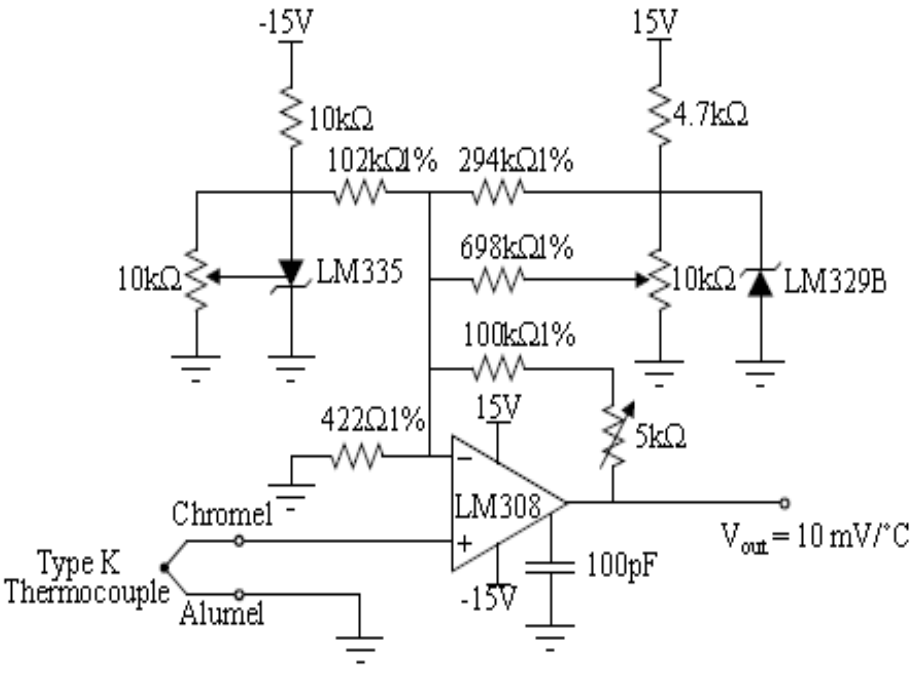
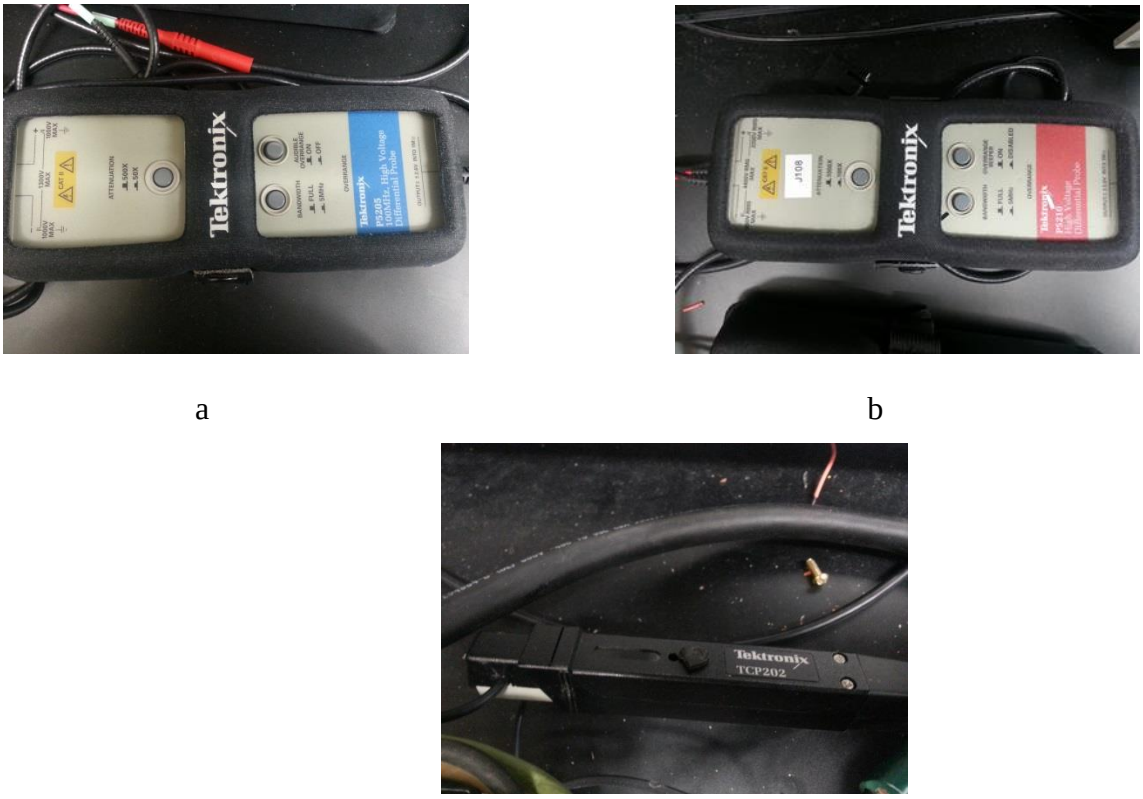


Figure 4.6: Circuit used for the measurement of IGBT case temperature in experimental drive. Form [72]

4.2.2 100MHz Sampling for Switch Condition Monitoring

As previously stated measurement circuits, sensors, differential probes and current guns are used to measure the raw signals that are used to extract the relevant features. For this section what is to be discussed is the use of oscilloscope differential probes and current guns used to get the raw signals that will be fed to the 100 MHz sampling Tektronix 3014b oscilloscope. This is acting as a data acquisition board. The phase current, I_C, is measured using the Tektronix TPS202 current gun which is a Hall-effect transducer, See Fig. 4.7 c. The gate to emitter voltage V_{GE} is measured and the voltage *Vx* across the resistor at the gate of the IGBT is measured by a low latency high voltage differential probes [Tektronix P5205] seen in Fig. 4.7 a. The gate to collector

voltage- V_{CE} is also measured by a low latency high voltage differential probe [Tektronix P5210] seen in Fig. 4.7 b.



a b

c

Figure 4.7: Measurement Device for 100 MHz Sampling. A – Tektronix P5205 High Voltage Differential probe. B – Tektronix P5210 High Voltage Differential probe. C – Tektronix TCP202 Current probe.

4.3 Feature Extraction

After the raw signals have been collected by the oscilloscope and data acquisition board, the relevant features must be extracted from them. The features that are extracted include $I_{C,rms}$, VJ, R_{ON}, Q_i, VT and $V_{GE,ON,AVG}$. Several features are procured through the processing measurements of I_C and V_{CE}. Section 4.1.1 illustrates the method for measuring V_{CE}. Although the literature suggests that V_{CE} can be directly applied in fault detection, this is only true under carefully controlled conditions that do not necessarily

apply in the field. In general, the circuit model as shown in in Fig 4.8 b for a conducting IGBT is a diode in series with a power MOSFET, meaning that

$$V_{CE,ON} = V_J + i_C R_{ON} \qquad 4.6$$

where R_{ON} is the resistance of the MOSFET channel and VJ represents the combination of the voltage drops across the p-n junction and the drift regions shown in in Fig 4.8 'a' [78].

It may be difficult to distinguish between changes in I_C and true fault conditions, due to the fact that most incipient faults have only a small impact on $V_{CE,ON}$. This problem is further compounded by the fact that both VJ and R_{ON} are affected by temperature [78]. In variable-speed ac drives feature extraction is notably difficult in part due to the fact that the current and $V_{CE,ON}$ are subjected to continuous variation as seen in Fig. 4.9. Figure 4.9 illustrates the effect of Eq. 4.6 in the operational drive. Typical behavior of $V_{CE,ON}$ is that it follows the current when the IGBT is conducting. The effect of the current thus has to be isolated, such that the two parameters can be estimated separately in Eq. 4.6. This is done using a least squares approach [72].

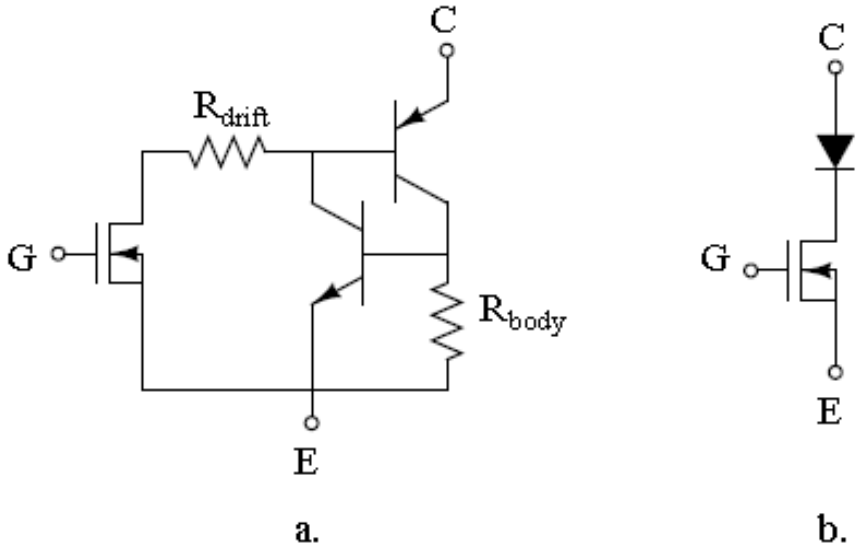


Figure 4.8: Measurement of phase current I_C and V_{CE} for the low side IGBT under test. When V_{CE} is negative, the anti-parallel diode is conducting.

In Fig 4.9 there are 4 plots; from top to bottom showing I_C Phase, V_{CE}, I_C and the voltage $V_{CE,ON}$ samples during transistor conduction that are taken for approximately 1 second. [But only an interval of 0.04 is shown in Fig. 4.9] After which a linear fit on the data is performed. Figure 4.10 shows the second of relevant measured points and the best-fit line that corresponds to the data shown in Fig. 4.9. The estimated y-intercept (*VJ*) and the slope (R_{ON}) are from the I-v curve developed from the conduction data of I_C transistor and $V_{CE,ON}$. These data point vectors *VJ* and Ron are ultimately extracted as features. The value RON for this particular second of data is 170.6 m and *VJ* is 1.053 V.

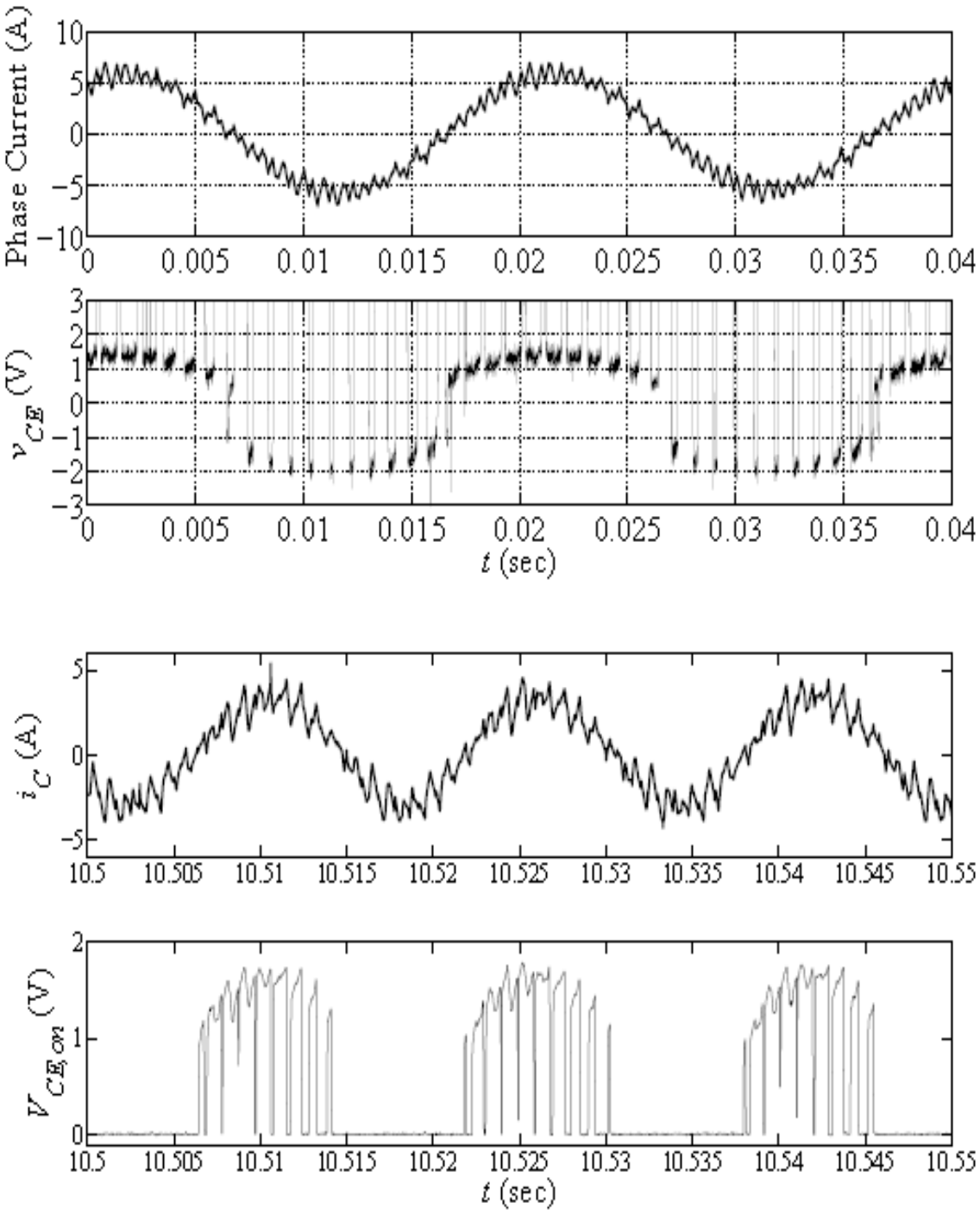


Figure 4.9: Oscilloscope measurement of phase current IC and V_{CE} and measurements of the phase current IC and $V_{CE,ON}$ with the sensing circuits for the low side IGBT under test. When V_{CE} is negative, the anti-parallel diode is conducting. From [72]

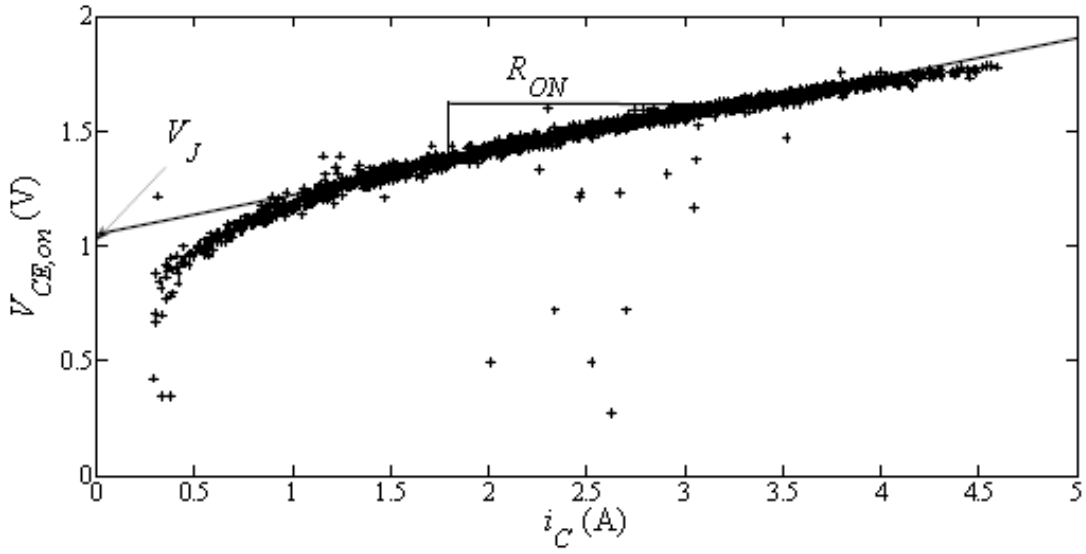


Figure 4.10: Plot showing $V_{CE,ON}$ versus I_C over one second. Note that a fit is performed to extract *VJ* and R_{ON}. From [72].

These parameters and the corresponding *rms* current are extracted. The root mean square of the phase current is calculated over the each individual period of a 60 Hz sinusoid and is defined as

$$I_{C,rms} = \sqrt{\frac{1}{N}\sum_1^N \|i_C\|^2} \qquad 4.7$$

where N is the number of points equaling 833, in a 60 Hz period which for a sampling frequency of 50 kHz. The mean of the *rms* current is then computed over one second intervals. Ultimately, estimates of T_A, T_C, I_C, VJ, and R_{ON} are processed over one second interval and are used to create a feature vectors at time $\boldsymbol{t_k}$. Threshold voltage and possibly other features could be included in the future.

The threshold voltage is derived from sampling the raw V_{GE} signal at 100 MHz with a P5205 differential probe connected to the Tektronix 3014b oscilloscope. In order to obtain *VT*, the collector to emitter voltage V_{CE} was as also measured in the same fashion with a high voltage differential probe. The current flowing through the IGBT- I_C was measured with current gun TCP202 also at 100 MHz frequency. A MATLAB script was written in order to detect the moment at which the threshold voltage is reached. This was done by looking at V_{GE} voltage corresponding to when the collector current began to flow. This signified the point where the transistor is turned on. (See Fig. 4.11)

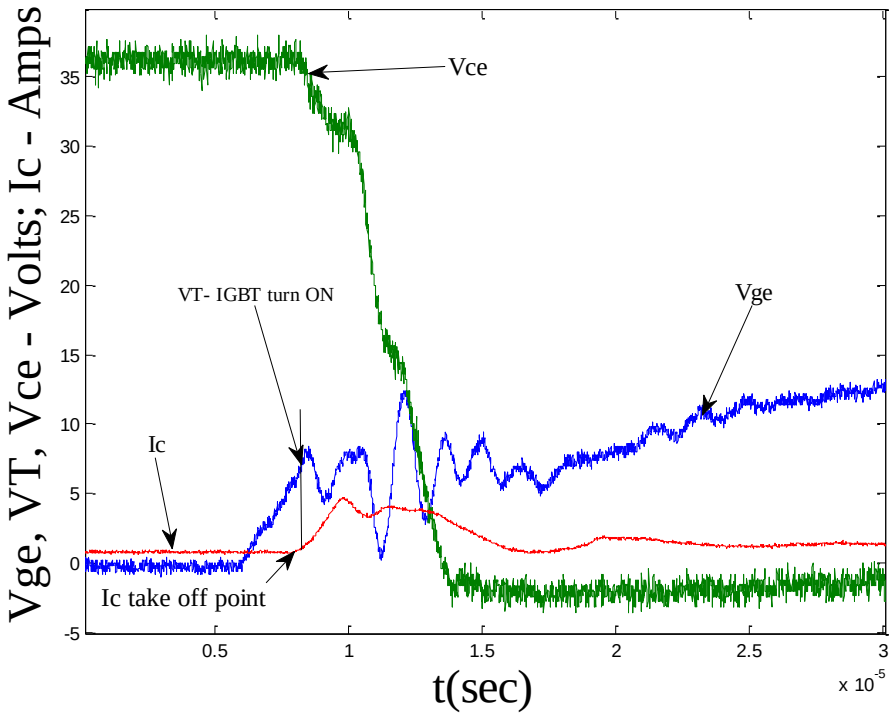


Figure 4.11: Switching signal for IGBT Turn on. Figure show the signals **VT**, V_{CE}, V_{GE}, and I_C.

Another feature derived from the V_{GE} sampled at 100 MHZ is $V_{GE,ON,AVG}$. $V_{GE,ON,AVG}$ is attained by averaging the last 1000 discrete points of the raw signal V_{GE} that

was sampled. (See Fig 4.12) It is important to know that due to the fact that some of the raw signals are being sampled at two different rates, there are two different capture rates. The data being sampled by the DT9816 data acquisition board is captured at intervals of 3 seconds and the data being sampled by the Tektronix 3014b oscilloscope is captured in 0.2 micro-second intervals. The oscilloscope is the first device to trigger based on digital logic with respect to the voltage levels of V_{GE} and the slope of V_{CE}. After the oscilloscope is triggered a control signal sent to start the sampling and data capture of the DT9816 (See Appendix A.3). In order to match the processing rate of 1 second interval for features T_C, T_A, I_C, VJ, and R_{ON}). The features VT and $V_{GE,ON,AVG}$ that are obtained from data captured at 0.2 micro-second are averaged for 1 second intervals.

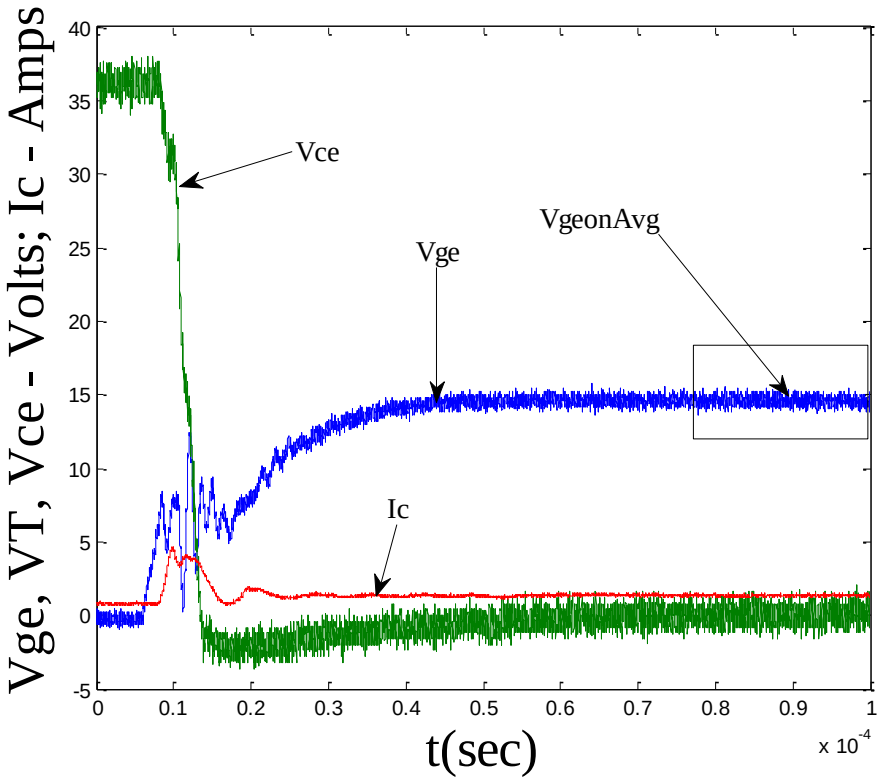


Figure 4.12: Show the section of V_{GE} signal that is average to obtain the KPI$V_{GE,ON,AVG}$.

Generally speaking, multiple samples of the raw signals or input data are needed to calculate any feature. Therefore, indicators are extracted at a rate well below the sampling frequency. For this experiment it's done at an interval of one second. The various feature estimates are ultimately combined with measurements of the case temperature and passed to the remainder of the algorithm. Feature vectors computed at the time $\boldsymbol{t_k}$ are grouped into a column vector $\boldsymbol{x_k}$. In our current implementation in the IGBT based drive, this vector includes the following features, each measured once per line cycle:

• Voltage drops across the p-n junction and drift regions, *VJ*
• On-state resistance of the MOSFET channel, R_{ON}
• Threshold Voltage, *VT*
• Root-mean-squared value of the collector current, I_C
• Average on-state gate-to-emitter voltage, $V_{GE,ON,AVG}$
• Ambient temperature, T_A,
• Case temperature, T_C

These features are then placed in the health monitoring algorithm detailed in section 3.3 to determine if there is a fault developing. (See Appendix A.1 and A.2 for all code relating to feature extraction)

4.4 IGBT Degradation for Testing

There is the need for both healthy and degraded/faulted IGBT data collection at different operating conditions, in order to show the algorithm's ability to recognize incipient faults. As such, the motor drive was operated with the IGBT in both healthy and degraded/faulted states, in various combinations of following loading levels depending on the health of the IGBT.

- 0% loading
- 20% Loading
- 40% Loading
- 50% Loading
- 60% Loading
- 80% Loading
- 100% Loading

This was done controlling the power output of the motor by means of torque control. The device used for torque control was a Magtrol Model DSP6001 dynamometer as seen in Fig 4.13.

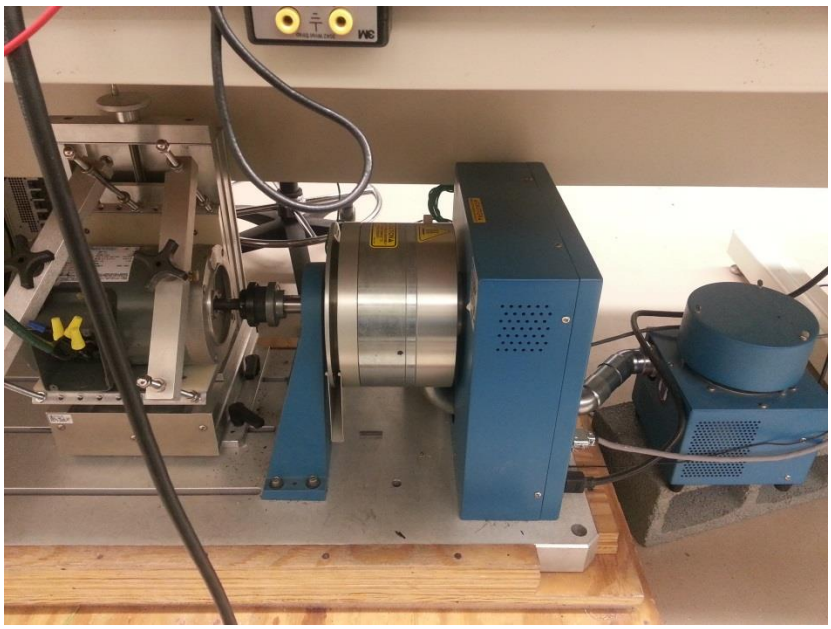


Figure 4.13: for Magtrol Dynamometer used to load Motor.

It was seen in section 2.2-2.3, that there are multiple fault mechanisms for the IGBT transistor. In this experimentation the fault mechanisms used were limited to two. A fault mechanism of the intrinsic nature, Dielectric Breakdown, and of the extrinsic nature, Latch-UP failure was chosen.

Several methods have been presented in the literature to degrade or artificially age power semiconductors [[79], [80]] to achieve these fault mechanisms. And of these Electrical and thermal stress are the two most common means by which power semiconductors are degraded. Therefore in order to simulate each of the two fault mechanisms are were selected, the methods of chronic temperature over stress and thermal cycling to induce latch-up will be used to cause Latch-Up and adding a Gate Resistor to the operating Drive will simulate Die electric Break Down due to gate leakage current. In the following sections each method will be explained in detail.

4.4.1 IGBT Aging due to Chronic Temperature over stress and Thermal Cycling

Multiple accelerated thermal ageing schemes are shown in the literatures [[27], [80], [81], and [82]]. Chronic temperature overstress and thermal cycling are the most prevalent thermal stress methods. Thermal overstress subjects the transistor to high temperatures for extended periods of time and can be artificially created by switching the transistor without proper heat sinking. The transistor is operated in extreme temperatures well beyond the safe operating area of the device. The transistor is aged by placing it into a Class-A amplifier and the case temperature is monitored with a temperature sensor. A controller is used to keep the temperature of the case at a constant value for an extended period of time [72].

Figure 4.14: Shows a schematic of the testing setup that is used to age the IGBTs by creating latch-up. The IGBTs after degradation are then used in the motor drive for fault detection.

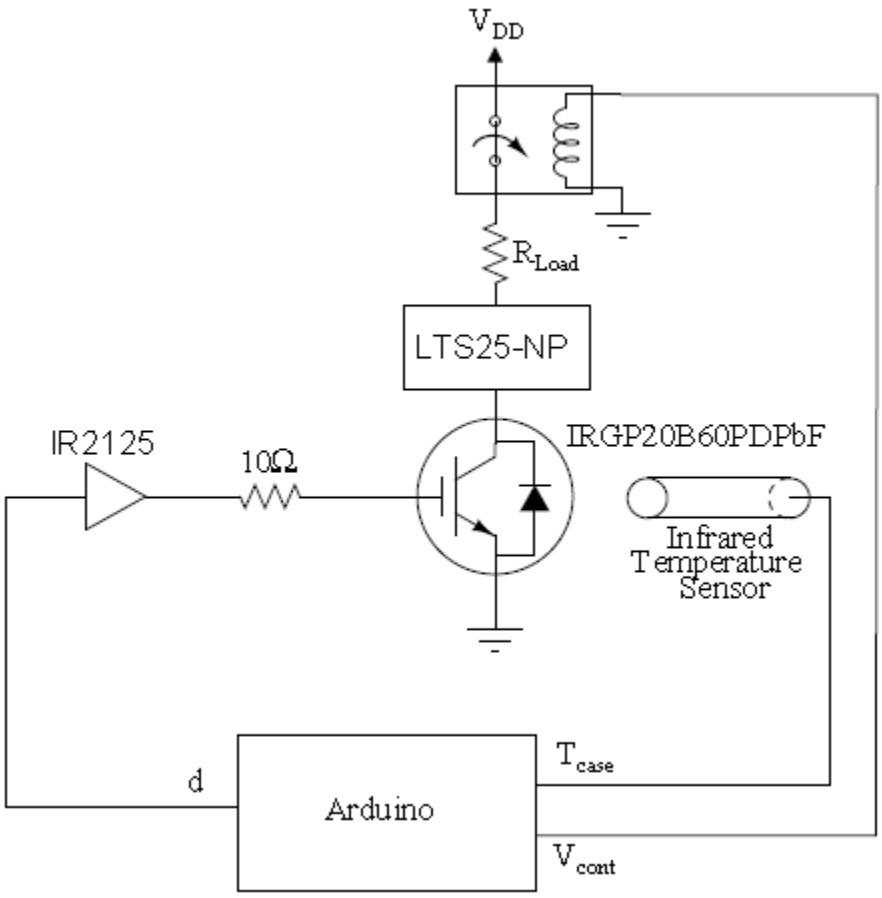


Figure 4.14: Circuit used to perform accelerated aging of the IGBT.

The gate signal in this circuit is a 1 kHz square waveform with a 40% duty ratio, *D* and a 15 Volt amplitude. The collector current, I_C, is controlled using the load resistor, R_{LOAD}, such that current flowing through the IGBT creates the wanted power dissipation in the device. The power dissipated by the IGBT is defined as

$$P_{diss} = V_{CE,ON} I_C D + P_{sw} \qquad 4.8$$

Where $V_{CE,ON}$ is the on-state collector-to-emitter voltage and *Psw* the power loss due to the low switching frequency.

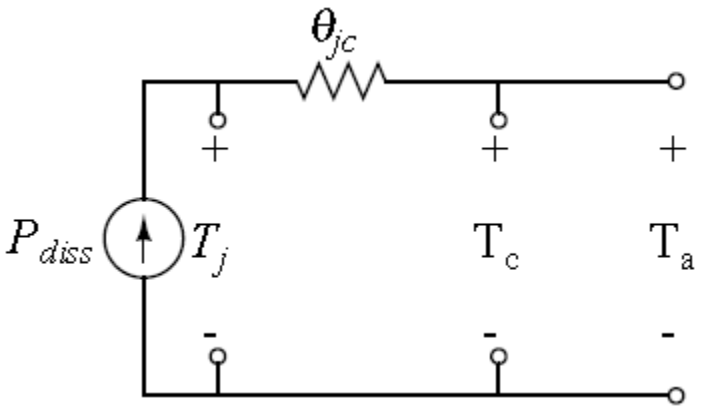


Figure 4.14: Thermal equivalent circuit for IGBT when placed in thermal degradation circuit.

The thermal circuit in Fig. 4.15 models the thermal conduction of the IGBT in the degradation circuit. Note that without the power dissipated in the IGBT, *Pdiss* due to thermal conduction is

$$P_{diss} = \frac{T_j - T_C}{\theta_{jc}} \qquad 4.9$$

where T_j is the junction temperature, T_c is the case temperature, and θ_{jc} is the junction-to-case thermal impedance in ◦C/W specified by the manufacturer. $T_{j,set}$ is the desired junction temperature during the first degradation test and it is set to 125% of the maximum operational junction temperature defined by the manufacturer. For the IRGP20B60PDPbF the maximum operational junction temperature is 150◦C so therefore $T_{j,set}$ is 187.5◦C. Since the junction temperature is not easy to measure the case temperature, $T_{c,set}$ is actually set to desired value using Eq.(4.8) and Eq.(4.9) and is defined as

$$T_{c,set} = T_{j,set} - P_{diss}\theta_{jc} \qquad 4.10$$

During degradation, the Arduino micro-controller, controls the case temperature at Tc,set by raising from room temperature and also monitors IC to insure the IGBT does not latch up. If the IGBT latches up during degradation VDD is disconnected using the relay which is also controlled by the Arduino. This allows the junction temperature to start falling. The IGBT is immediately tested for a short between the collector and the emitter which indicates a latch up event has occurred. After the IGBT has cooled to room temperature the IGBT is again tested for a short. If there is still a short, then the device has been permanently failed and is thus not useable for further testing. If there is no short present the degradation is restarted again after the junction temperature has been reduced but with Tj,set reduced to 50◦C. This is done to ensure the latch up event has cleared and to check if gate control is still present, thus ensuring a successful degradation. The failure mechanisms that can be caused by a thermal over- stress include die-attach failures [82], gate latch-up [27], and time-dependent dielectric breakdown [72].

Another accelerated aging methodology that is also popular is thermal cycling. For this technique power transistors are made to go through rapid changes in temperatures therefore causing thermal expansion and contraction. This procedure uses the same circuit as shown in Fig. 4.14 but the device operates in the safe area of operation only. The temperature of the transistor is cycled using the degradation circuit in the following steps:

- Hold case temperature at 50◦C for 5 minutes.
- Increase case temperature to 80◦C for 5 minutes.
- Increase case temperature to 110◦C for 10 minutes.
- Increase case temperature to 140◦C for 20 minutes.
- Decrease case temperature to 110◦C for 10 minutes.

- Decrease case temperature to 80◦C for 5 minutes.
- Decrease case temperature to 50◦C for 5 minutes.

The failure mode associated with thermal cycling is the failure of the transistor to switch as a result of the increased resistance of the device. As a result of the increased resistance, the drain current dropped and the transistor fails to switch. Die solder degradation and wire lift are associated with thermal cycling [72].

4.4.1 IGBT Aging due to Power/Thermal Cycling Setup

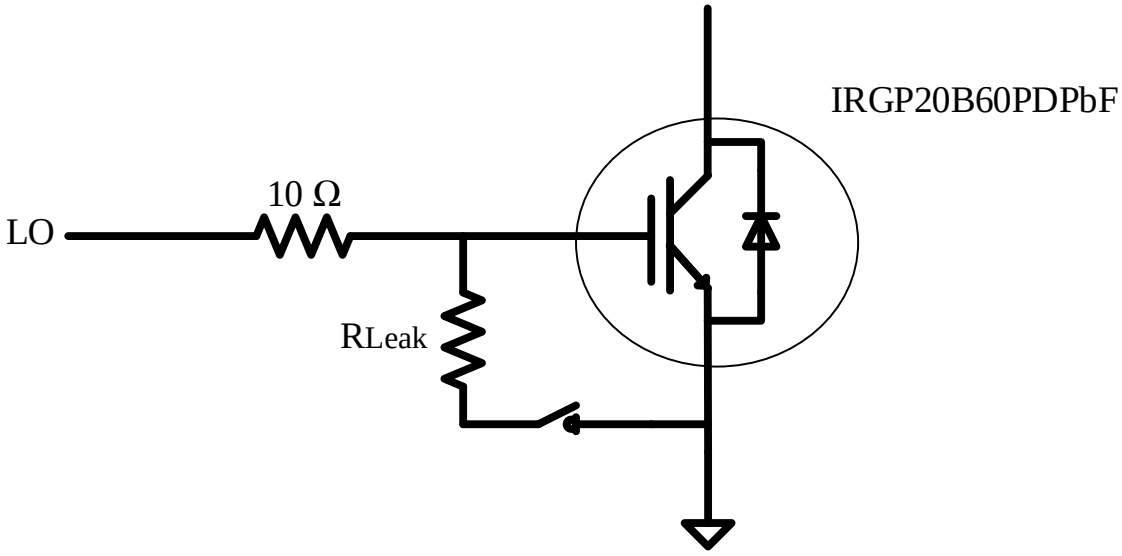


Figure 4.16: $V_{CE,ON}$ versus temperature for the IGBT, both before and after aging.

An electrical failure was developed by using a method to mimic the degradation of the gate dielectric. Figure 4.16 shows the test setup for the gate dielectric wear out detection. When a gate dielectric breakdown occurs a large leakage current is seen during the on and off state of an IGBT. This condition is artificially created by placing in parallel a resistor R_{Leak} between the gate and emitter during operation. A 47k, 4.7k, and 470 resistors were placed between the gate and emitter to simulate and increase in the leakage gate current. An increase in the gate leakage current which occurs after the gate

dielectric wears out was observed when looking at the voltage across the resistor connected between the gate driver and the gate of the IGBT [72].

CHAPTER 5: RESULTS

In order to validate the OIFD scheme being proposed, there are two test cases that were developed and executed. These test cases were based on the two failure mechanisms that were induced; Latch UP failure and Gate Die attach Breakdown. These were then individually compared to healthy data from the IGBT in the motor drive operating at different loading levels. The first test case that is shown is Thermal Cycling - Latch up Failure and the second test case is called Gate Resistance Induced Die Attach Breakdown. See Table 5.1 gate test case breakdown.

Table 5.1: Test cases used in experimentation.

IGBT	**Test Cases**	**Stage 1**	**Loading**	**Stage 2**	**Gate Resistance**	**Loading**	**Failure Method**
1	Latch-Up	Healthy	0-100%	Degraded	0	0 & 100%	Chronic Temperature over stress
2	Gate Dielectric Breakdown	Healthy	0-100%	Degraded	47000 ohms	0%, 50%, 100%	Simulated gate current leakage by gate-emitter resistance
2	Gate Dielectric Breakdown	Healthy	0-100%	Degraded	4700 ohms	0%, 50%, 100%	Simulated gate current leakage by gate-emitter resistance
2	Gate Dielectric Breakdown	Healthy	0-100%	Degraded	470 ohms	0%, 50%, 100%	Simulated gate current leakage by gate-emitter resistance

5.1 Latch up Failure due Chronic Temperature Overstress

The monitored IGBT was first removed and then subjected to an accelerated aging test to produce a Latch UP failure. After creating a Latch up an early stage die-attach fault was observed. This was consistent with results from the literatures on accelerated thermal ageing schemes are shown in the literatures [[27], [80], [81], and [82]]. Figure 5.1 shows results recorded during the aging process. A positive temperature coefficient was seen; this is expected for IGBTs [82]. Also, the component had a lower average on-state V_{CE} at all temperatures once it has degraded. This result was also obtained in [82]. It is believed that the drop occurs because the degraded die-attach increases θjc and thus creates a higher internal temperature at any given T_C. Since the pn junction has a negative temperature coefficient of resistance [25], an increase in the temperature of this junction lowers average on state V_{CE}[72]. Figure 5.2 shows the results obtained using the proposed condition-monitoring algorithm once the degraded part was returned to the drive.

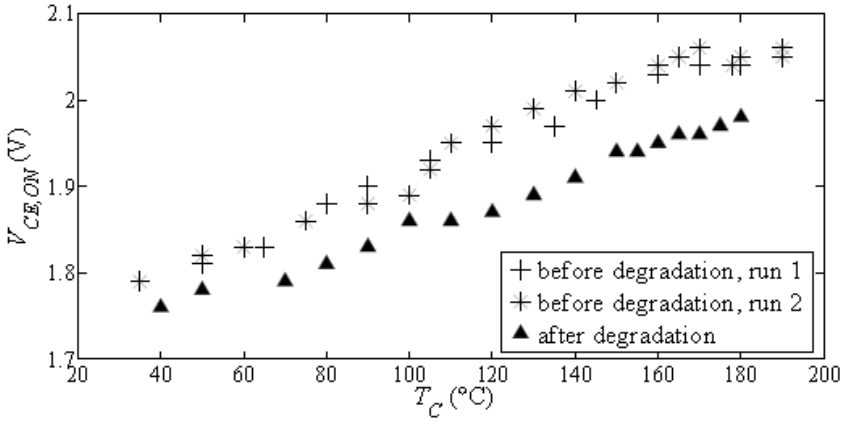


Figure 5.1: $V_{CE,ON}$ versus temperature for the IGBT, both before and after aging. From [72].

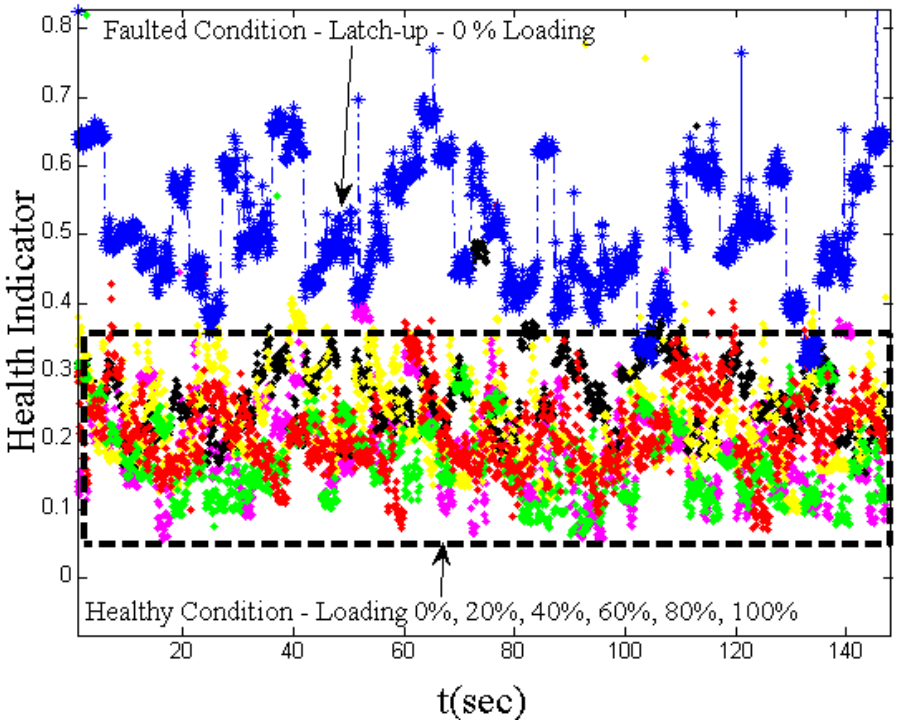


Figure 5.2: The reconstruction error recorded by the health-monitoring algorithm before and after the die-attach fault was induced. Note that there are a number of samples recorded at each loading condition and that the higher error at each load step corresponds to the degraded transistor. The training data is chosen from the first 5 minutes on all healthy data loading levels. This takes into consideration startup conditions as well as steady state operation.

For this case, as was stated before, this part was tested in both the healthy and degraded states. The healthy state IGBT was tested with a varied load on the machine in order to simulate changes in operating conditions. For every loading level the data collected for the healthy data was over a time period of 3 hours, with a 3 second interval capture rate. The data for the degraded data was collected for over a time period of 20 minutes with a 3 second interval capture rate. The reason behind the shorter collection time for the degraded part was due its volatility. When latch-up failure occurs the lifespan of the IGBT is shortened and its failure becomes random [this is based on observation of the operation on many latch parts]. When these sporadic failures occurred the custom

drive would be destroyed, thus as a mitigating condition, the testing time of these parts were significantly reduced to ensure successful testing. The Figure 5.2 shows the reconstruction errors in both the healthy and degraded conditions. The degraded IGBT caused a significantly higher reconstruction error at each load step. Note that changes in the loading condition had minimal impact in comparison. It can also be seen that the drive's operation appeared normal despite the fact that the device was nearing failure. This ultimately, provides a solid argument for the incipient fault monitoring capabilities of the offline detection scheme.

5.2 Gate Resistance Induced Die Attach Breakdown.

For the other test case, the effect of a breakdown in the gate oxide was simulated, which results in an increased leakage current [12]. As in [20], this fault was simulated by placing resistance between the gate and the emitter of the monitored switch. Different resistor values were used to demonstrate the ability to detect early stage faults. Figure 3.26 shows the reconstruction error versus sampled points for both healthy and faulted operation, with 20% load instrumentations form no-load to full load. The $V_{GE,ON,AVG}$ indicator changes by lowering in magnitude, with an increase in leakage current from hundreds of micro-amps to tens of milliamps. Thus with a decrease in the resistance across the gate there is a larger change in the $V_{GE,ON,AVG}$ magnitude.

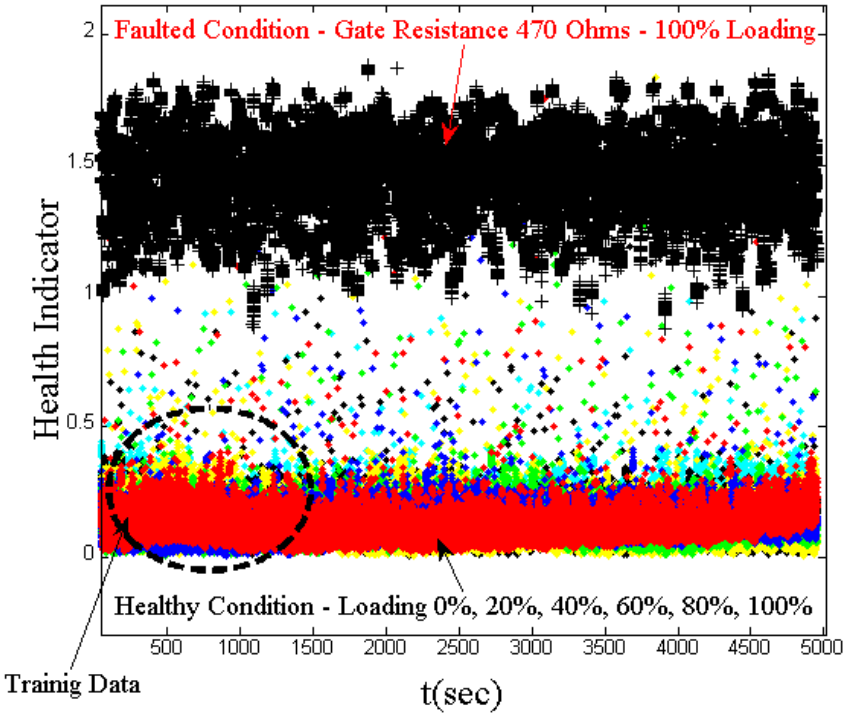


Figure 5.3: The reconstruction error recorded by the health-monitoring algorithm in 20% load steps for a healthy part. Note that they are all reasonably similar. The high errors beginning occurs at gate resistance of 470 ohms which correspond to increased gate current ~ 32 mA, respectively. The Faulted data is at 100% loading. The training data is chosen from the first 15 minutes on all healthy data loading levels. This takes into consideration startup conditions as well as steady state operation.

To simulate a faulted part, a 47k resistor was added to the drive. The drive was subsequently run from no load to full load in 20 percent increments. The same process was then repeated for the resistors with the values of 4.7k ohms and 470 ohms. It can be observed that the corresponding leakage currents are small (~ 0.32 mA, ~ 3.2 mA and ~ 32 mA, respectively), and that all conditions correspond to scenarios in which the drive maintains complete functionality despite the presence of the soon-to-fail switch. It is important to note that for a resistor value of 47k ohms no discernable difference between that simulated faulted part and a part test. This is due in large part to the very small

leakage current that actually occurs. Because there were no discernable changes, no plots on this test scenario is presented. Figure 5.3 shows the reconstruction error by the health-monitoring algorithm for the test case with simulated fault due to 470 ohm resistance. Other tests are completed at other loading levels with said gate resistances but all followed the exact same pattern as the 100 % loading. Therefore for clarity in terms of number of plots, only 100% loading conditions are shown. Upon observing all resistance valued simulated faults, a clear trend of, with a increase in resistance value there is a decrease in health indicator/reconstruction error. Figure 5.4 gives a clear demonstration of this concept, showing a lower health indicator value for a resistance value of 4700 ohms than for that of 470 ohms.

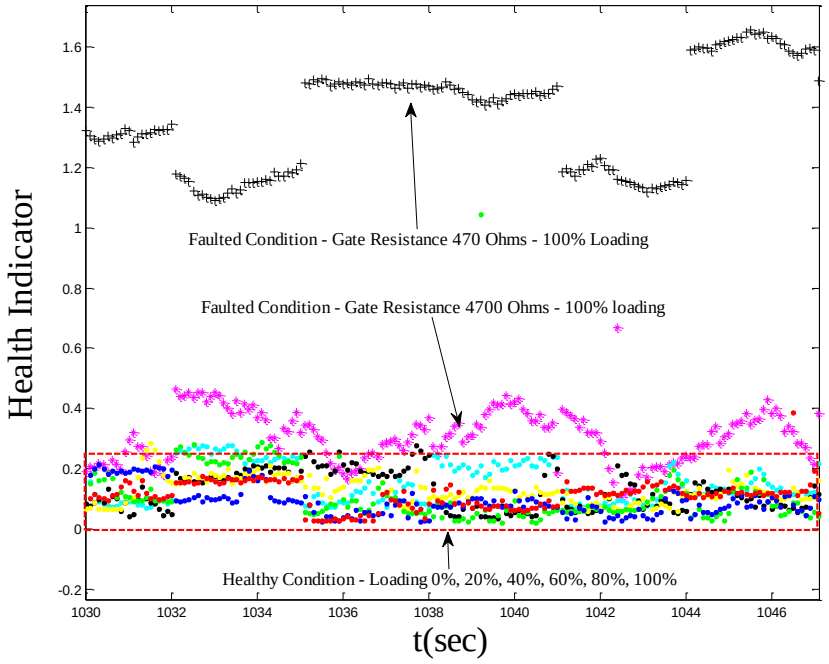


Figure 5.4: The reconstruction error recorded by the health-monitoring algorithm in 20 % load steps for a healthy part. Note that they are all reasonably similar. The high errors starts at a gate resistance of 4700 ohms, and continues to rise at gate resistance of 470 ohms which correspond to increased gate current ~ 32 mA, respectively. The Faulted datas are at 100% loading.

CHAPTER 6: SUMMARY AND FUTURE WORKS

6.1 Summary

This thesis proposed an incipient fault detection scheme for robust condition monitoring of IGBT(s) in power conversion devices that robustly detects faults locally allowing for remote fault diagnosis. The fault detection algorithm was developed based on a concept from facial recognition. The thesis developed several methods for the offline measurement of key health indicators in the IGBT(s). The algorithm was then implemented using the health indicators for incipient fault detection of IGBT(s) in a custom 3-Ø motor drive. Seeing that online methods for early stage fault detection will become essential as systems become more heavily dependent on power electronics, the proof–of-concept offline fault detection scheme was developed and tested. This scheme, that is able to be converted to an online approach was able to distinguish between true faults and changes in operating conditions and did not require the development of rules to set thresholds for fault detection. This scheme showed sensitivity to small changes in load behavior which provides the ability to correlate those changes with the underlying KPI data, as well as maintenance records and other process-variable measurements.

6.2 Future Works

One potential improvement for fault detection in power semiconductors is using the advanced gate-drive concept that has been described in several works [14]-[17].

Figure 1.2 shows a three-phase, full-bridge inverter capable of sampling the required terminal variables. FPGA-based gate drives have been used to acquire switch terminal variables at rates as high as 100 MHZ with emphasis on their use in optimizing turn-on and turn-off performance [15], [16]. The signals measured by such devices can also be used to extract meaningful health-related features, such as threshold voltage, on-state voltage and on-state resistance. Feature extraction could be performed locally at the gate-drive unit, and features could be transmitted back to a central controller over fiber-optic cables for better noise immunity at a much lower data rate [72]. A potential smart drive architecture including FPGA-based gate drives. These devices are labeled here as advanced gate drivers (AGDs). The AGDs provide the controller with appropriate measurements. These connections would ideally be fiber optic [83], [72].

The advanced gate-drive concept provides a viable method for monitoring key indicators. For this thesis these signals were obtained without the onboard solution of the high-speed FPGAs. Though, it would not be difficult to implement the condition monitoring algorithm using an FPGA-based gate drive, so the off-line processing should not be viewed as a limitation of the approach but as poof viability. The algorithm could be implemented by either the controller of the inverter or at each individual gate drive [72]. The method of using sampling devices off the drive board was used here simply for ease of demonstration. The system's architecture for this thesis was constructed to focus on demonstrating the concept rather than engineering design. Though, it should be noted that the limited resolution of the high-speed data converters on the FPGA would likely require some level of analog preprocessing regardless of the format of the implementation. A major advantage of using the advanced gate-drive concept is that it

allows additional features that are not currently being monitored to be, such as the gate charge, after settling time is achieved.

REFERENCES

[1] "Power Electronics Market worth $36.86 Billion by 2020." Yahoo Finance. N.p., 01 May 2014. Web. 24 Nov. 2014. <http://finance.yahoo.com/news/power-electronics-market-worth-36-160000751.html>.

[2] J. M. Anderson, R. W. Cox, and P. O'Connor. "Online algorithm for early stage fault detection in IGBT switches." Diagnostics for Electric Machines, Power Electronics and Drives, Valencia, 2013.

[3] N. D. Benavides, T. J. McCoy, and M. A. Chrin, “Reliability improvements in integrated power systems with pressure-contact semiconductors,” in Proc. ASNE Day, Apr 2009.

[4] C. Huifeng, J. Bing, V. Pickert, and C. Wenping, "Real-Time Temperature Estimation for Power MOSFETs Considering Thermal Aging Effects," Device and Materials Reliability, IEEE Transactions on , vol.14, no.1, pp.220,228, March 2014

[5] W. Lixiang, R. J. Kerkman, R. A. Lukaszewski, B. P. Brown, N. Gollhardt, and B. W. Weiss, “Junction temperature prediction of a multiple-chip IGBT module under DC condition,” in Conf. Rec. IEEE 41st IAS Annual Meeting, 2006, pp. 754–762.

[6] Z. Khatir, S. Carubelli, and F. Lecoq, “Real-time computation of thermal constraints in multichip power electronic devices,” IEEE Trans. Compon. Packag. Technol., vol. 27, no. 2, pp. 337–344, Jun. 2004.

[7] M. Musallam, P. P. Musallam, C. M. Johnson, L. Pritchard, and V. Pickert, “Open loop real-time power electronic device junction temperature estimation,” in Proc. IEEE Int. Symp. Ind. Electron. 2004, vol. 2, pp. 1041–1046.

[8] T. Reimann, R. Krummer, U. Franke, J. Petzoldt, and L. Lorenz, “Real time calculation of the chip temperature of power modules in PWM inverters using a 16 bit microcontroller,” in Proc. 12th Int. Symp. Power Semicond. Dev. ICs, 2000, pp. 127–130.

[9] K. Fink and S. Bernet, “Advanced Gate Drive Unit with closed-looped di/dt-control,” IEEE Transactions on Power Electronics, vol. PP, no. 99, pp. 1–9, Aug. 2012.

[10] L. Michel, X. Boucher, A. Cheriti, P. Sicard, and F. Sirois, “FPGA implementation of an optimal IGBT gate driver based on Posicast control,” IEEE Transactions on Power Electronics, vol. PP, no. 99, pp. 1-7, Jul. 2012.

[11] Z. Wang, X. Shi, L. M. Tolbert, and B. J. Blalock, “Switching performance improvement of IGBT modules using an active gate driver,” in IEEE Applied Power Electronics Conference and Exposition, 2013, 8 pages.

[12] S. Lombardo, J. Stathis, B. Linder, K. Pey, F. Palumbo, and C. Tung, “Dielectric breakdown mechanisms in gate oxides,” Journal of Applied Physics 98, 2005.

[13] G. Buh, H. Chung, and Y. Kuk, “Real-time evolution of trapped charge in a SiO2 layer: An electrostatic force,” Applied Physics Letters, vol. 79, no. 13, pp. 2010–2012, 2001.

[14] P. Xiao, G. K. Venayagamoorthy, K. A. Corzine, and R. Woodley, “Selfhealing control with multifunctional gate drive circuits for power converters,” in Conference Record of the 2007 IEEE Industry Applications Conference, New Orleans, LA, Sept. 2007, pp. 1852 – 1858.

[15] Y. Lobsiger and J. W. Kolar, “Voltage, current and temperature measurement concepts enabling intelligent gate drives,” in European Center for Power Electronics (ECPE) Workshop – Electronics Around the Power Switch: Gate Drives, Sensors, and Control, 2011.

[16] H. Kuhn, T. Koneke, and A. Mertens, “Potential of digital gate units in high power applications,” in Proc. 13th Power Electronics and Motion Conference (EPE-PEMC), 2008, pp. 1458–1464.

[17] ——, “Considerations for a digital gate unit in high power applications,” in Proc. 2008 Power Electronics Specialists Conference (PESC), 2008, pp. 2784–2790.

[18] A. M. S. Mendes and A. J. M. Cardoso, “Fault-tolerant operating strategies applied to three-phase induction-motor drives,” IEEE Trans. Ind. Electron., vol. 53, no. 6, pp. 1807–1817, Dec. 2006.

[19] J. R. Fu and T. A, Lipo, “A strategy to isolate the switching device fault of a current regulated motor drive,” Proc. IEEE IAS Annual Meeting, vol. 2, pp. 1015-l020, 1993.

[20] L. Chen, F. Z. Peng, and D. Cao, “A smart gate drive with self-diagnosis for power MOSFETs and IGBTs,” in Proc. of IEEE Applied Power Electronics Conference and Exposition, Austin, TX, Feb. 2008, pp. 1602 – 1607.

[21] A. Sattar. "Insulated Gate Bipolar Transistor (IGBT) Basics." Santa Clara: IXYS Corporation (2003): 11-14.

[22] Semiconductor, Fairchild. "Application Note 9016." IGBT Basics 1.

[23] "An Industry-Based Survey of Reliability in Power Electronic Converters", IEEE IAS, 2011.

[24] V. K. Khanna, The Insulated Gate Bipolar Transistor (IGBT): Theory and Design. Piscataway, NJ: IEEE Press, 2003. ISBN 0-471-23845-7.

[25] B. Jayant Baliga, "Power Semiconductor Devices" PWS Publishing Company, ISBN: 0-534-94098-6, 1996.

[26] Ralph Locher, "Introduction to Power MOSFETs and their Applications" Fairchild Semiconductor, Application Note 558, October 1998.

[27] D. Brown, M. Abbas, A. Ginart, I. Ali, P. Kalgren, and G. Vachtsevanos, "Turn-off time as a precursor of gate bipolar transistor latch-up faults in electric motor drives," in Proc of Annual Conference of the Prognostics and Health Management Society, Portland, Oregon, Oct. 2010.

[28] R. S. Chokhawala, J. Catt, and L. Kiraly, "A discussion on IGBT shortcircuit behavior and fault protection schemes," IEEE Trans. Ind. Appl., vol. 31, no. 2, pp. 256–263, Mar./Apr. 1995.

[29] T. Aoki, "A discussion on the temperature dependence of latch-up trigger current in CMOS/BICMOS structure," IEEE Trans. Electron. Devices, vol. 40, no. 11, pp. 2023–2028, Nov. 1993.

[30] J. D. Kraus, Electromagnetics. McGraw-Hill Book Company, Inc., 1953.

[31] J. R. Black, "Electromigration—A brief survey and some recent results," IEEE Trans. Electron Devices, vol. ED-16, pp. 338–347, 1969.

[32] D. L. Goodman, "Prognostic methodology for deep submicron semiconductor failure modes," IEEE Trans. on Components and Packaging Technologies, vol. 24, no. 1, pp. 109 – 111, March 2001.

[33] L. Somos, "Power semiconductors empirical diagrams expressing life as a function of temperature excursion," IEEE Trans. Magn., vol. 29, no. 1, pp. 517–522, 1993.

[34] M. Held, P. Jacob, G. Nicoletti, P. Scacco, M.H. Poech: Fast Power Cycling Test for IGBT Modules in Traction Application, Proceedings Power Electronics and Drive Systems, IEEE 1997

[35] M. Ciappa, "Selected failure mechanisms of modern power modules," Microelectronics Reliability, no. 42, pp. 653–667, Jan 2002.

[36] D. Katsis and J. van Wyk, "Void-induced thermal impedance in power semiconductor modules: Some transient temperature effects," IEEE Trans. Ind. Appl., vol. 39, no. 5, pp. 1239 – 1246, Sept. / Oct. 2003.

[37] OPNAVINST 4790.16A, "Condition Based Maintenance (CBM) Policy," 17 Dec 2007.

[38] R. Caccese and D. Lewis, "Implementing true CBM+ in the fleet," in Proc. ASNE Intelligent Ships Symposium VIII, Philadelphia, PA, May 2009.

[39] M. G. Walker, R. Kapadia, B. Sammuli, and M. Venkatesh, "A model-based reasoning framework for condition based maintenance and distance support," in Proc. ASNE Automation and Controls Symposium, Biloxi, MS, Dec. 2007.

[40] F. W. Fuchs, "Some diagnosis methods for voltage source inverters in variable speed drives with induction machines—A survey," in Proc. IEEE Ind. Electron. Conf., 2003, pp. 1378–1385.

[41] B. Lu and S. Sharma, "A literature review of IGBT fault diagnostic and protection methods for power inverters," in Conf. Rec. IEEE IAS Annu. Meeting, Oct. 2008, pp. 1–8.

[42] D. Kastha and B. K. Bose, "Investigation of fault modes of voltage-mode inverter system for induction motor," IEEE Trans. Ind. Appl., vol. 30, no. 4, pp. 1028–1038, Jul./Aug. 1994.

[43] A M. S. Mendes and A. J. Marques Cardoso, "Voltage source inverter fault diagnosis in variable speed ac drives, by the average current Park's vector approach," in Proc. IEMDC, 1999, pp. 704–706.

[44] K. Rothenhagen and F. W. Fuchs, "Performance of diagnosis methods for IGBT open circuit faults in three phase voltage source inverters for ac variable speed drives," in Proc. Eur. Power Electron. Appl. Conf., 2005, pp. 1–10.

[45] K. Rothenhagen and F. W. Fuchs, "Performance of diagnosis methods for IGBT open circuit faults in voltage source active rectifiers," in Proc. IEEE Power Electron. Spec. Conf., 2004, pp. 4348–4354.

[46] R. Peuget, S. Courtine, and J. P. Rognon, "Fault detection and isolation on a PWM inverter by knowledge-based model," IEEE Trans. Ind. Appl., vol. 34, no. 6, pp. 1318–1326, Nov./Dec. 1998.

[47] R. L. A. Ribeiro, C. B. Jacobina, E. R. C. Silva, and A. M. N. Lima, "Fault detection of open-switch damage in voltage-fed PWM motor drive systems," IEEE Trans. Power Electron., vol. 18, no. 2, pp. 587–593, Mar. 2003.

[48] O. S. Yu, N. J. Park, and D. S. Hyun, "A novel fault detection scheme for voltage fed PWM inverter," in Proc. IEEE Ind. Electron. Conf., 2006, pp. 2654–2659.

[49] P. Gilreath and B. N. Singh, "A new centroid based fault detection method for 3-phase inverter-fed induction motors," in Proc. IEEE Power Electron. Spec. Conf., 2005, pp. 2663–2669.

[50] L. Zhang, I. B. Aris, and L. N. Hulley, "A knowledge-based system for online fault diagnosis of power inverter circuits for AC machines," in Proc. Eur. Power Electron. Conf., 1995, pp. 334–339.

[51] R. Szczesny, H. Piquet, and P. Kurzynski, "Fault detection and diagnosis in the electric drives," in Proc. Eur. Power Electron. Appl. Conf., 1997, pp. 995–1000.

[52] B. Raison, G. Rostaing, and J. P. Rognon, "Towards a global monitoring scheme for induction motor drives," in Proc. Int. Power Electron. Conf., 2000, pp. 1183–1188.

[53] C. Kral and K. Kafka, "Power electronics monitoring for a controlled voltage source inverter drive with induction machines," in Proc. IEEE Power Electron. Spec. Conf., 2000, pp. 213–217.

[54] M. R. Mamat, M. Rizon, and M. S. Khanniche, "Fault detection of 3-phase VSI using wavelet-fuzzy algorithm," Amer. J. Appl. Sci., vol. 3, no. 1, pp. 1642–1648, 2006.

[55] F. Charfi, F. Sellami, and K. Al-Haddad, "Fault diagnosis in power system using wavelet transforms and neural networks," in Proc. IEEE Int. Symp. Ind. Electron., 2006, pp. 1143–1148.

[56] M. A. Awadallah and M. M. Morcos, "Diagnosis of switch open-circuit fault in PM brushless dc motor drives," in Proc. Large Eng. Syst. Conf. Power Eng., May 2003, pp. 69–73.

[57] J. H. Park, D. H. Kim, S. S. Kim, D. J. Lee, and M. G. Chun, "C-ANFIS based fault diagnosis for voltage-fed PWM motor drive systems," in Proc. Annu. Meeting North Amer. Fuzzy Inform. Soc., 2004, pp. 379–383.

[58] Y. Guan, D. Sun, and Y. He, "Mean current vector based online real-time diagnosis for voltage source inverter fed induction motor drives," in Proc. IEMDC, 2007, pp. 1114–1118.

[59] M. A. Masrur, Z. Chen, B. Zhang, and Y. L. Murphey, "Model-based fault diagnosis in electric drive inverters using artificial neural network," in Proc. IEEE Gen. Meeting Power Eng. Soc., 2007, pp. 1–7.

[60] K. Debebe, V. Rajagopalan, and T. S. Sankar, "Expert systems for fault diagnosis of VSI fed ac drives," in Conf. Rec. IEEE IAS Annu. Meeting, 1991, pp. 368–373.

[61] K. Debebe, V. Rajagopalan, and T. S. Sankar, "Diagnosis and monitoringfor ac drives," in Conf. Rec. IEEE IAS Annu. Meeting, 1992, pp. 370–377.

[62] B. M. Gonzalez-Contreras, J. L. Rullan-Lara, L. G. Vela-Valdes, and A. Claudio, "Modelling, simulation and fault diagnosis of the three-phase inverter using bond graph," in Proc. IEEE Int. Symp. Ind. Electron., 2007, pp. 130–135.

[63] M. A. Rodriguez, A. Claudio, D. Theilliol, and L. G. Vela, "A new fault detection technique for IGBT based on gate voltage monitoring," in Proc. IEEE Power Electron. Spec. Conf., 2007, pp. 1001–1005.

[64] F. Huang and F. Flett, "IGBT fault protection based on di/dt feedback control," in Proc. IEEE Power Electron. Spec. Conf., 2007, pp. 1478–1484.

[65] S. Musumeci, R. Pagano, A. Raciti, G. Belverde, C. Guastella, and M. Melito, "A novel protection technique devoted to the improvement of the short circuit ruggedness of IGBTs," in Proc. IEEE IECON, 2003, pp. 1733–1738.

[66] R. S. Chokhawala and S. Sobhani, "Switching voltage transient protection schemes for high-current IGBT modules," IEEE Trans. Ind. Appl., vol. 33, no. 6, pp. 1601–1610, Nov./Dec. 1997.

[67] A Bhalla, S. Shekhawat, J. Gladish, J. Yedinak, and G. Dolny, "IGBT behavior during desat detection and short circuit fault protection," in Proc. Int. Symp. Power Semicond. Devices ICs, 1998, pp. 245–248.

[68] F. Richardeau, P. Baudesson, and T. A. Meynard, "Failures-tolerance and remedial strategies of a PWM multicell inverter," IEEE Trans. Power Electron., vol. 17, no. 6, pp. 905–912, Nov. 2002.

[69] K. S. Smith, L. Ran, and J. Penman, "Real-time detection of intermittent misfiring in a voltage-fed PWM inverter induction-motor drive," IEEE Trans. Ind. Electron., vol. 44, no. 4, pp. 468–476, Aug. 1997.

[70] D. Diallo, "Fault detection and diagnosis in an induction motor drive: A pattern recognition approach based on Concordia stator mean current vector," IEEE Trans. Energy Convers., vol. 20, no. 3, pp. 512–519, Sep. 2005.

[71] F. Zidani, D. Diallo, M. E. H. Benbouzid, and R. Naït-Saïd, “A fuzzybased approach for the diagnosis of fault modes in a voltage-fed PWM inverter induction motor drive,” IEEE Trans. Ind. Electron., vol. 55, no. 2, pp. 586–593, Feb. 2008.

[72] J. M. Anderson. "Robust condition monitoring for modern power conversion," Ph.D. dissertation, Dept. of Electr. & Comput. Eng., UNC Charlotte, Univ., Charlotte, N.C., 2013.

[73] M. Turk and A. Pentland, “Eigenfaces for recognition,” Journal of Cognitive Neuro-science, vol. 3, no. 1, pp. 71 – 86, 1991.

[74] R. Duda, P. Hart, and D. Stork, Pattern Classification. Wiley, 2006

[75] J. Morroni, A. Dolgov, R. Zane, and D. Maksimovi, “Online health monitoring in digitally controlled power converters,” in Proc. IEEE Power Electron. Specialists Conf., Orlando, FL, Jun 2007.

[76] Data Translation, UM-21336-T DT9816 Series Users Manual, Sept. 2012.

[77] HCNR200 and HCNR201: High-Linearity Analog Optocouplers, Avago Technologies, 2011.

[78] N. Mohan, T. Undeland, and W. Robbins, Power Electronics Converters, Applications, and Design, Third Edition. John Wiley & Sons, 2003.

[79] G. Sonnenfeld, K. Goebel, and J. R. Celaya, “An agile accelerated aging, characterization and scenario simulation system for gate controlled power transistors,” in Proc of IEEE AUTOTESTCON, Salt Lake City, UT, Sept. 2008, pp. 208 –215.

[80] J. Celaya, N. Patil, S. Saha, P. Wysocki3, and K. Goebel, “Towards acceler- ated aging methodologies and health management of power MOSFETs (technical brief),” in Proc. of Annual Conference of the Prognostics and
Health Management Society, San Diego, CA, Sept./Oct. 2009.

[81] J. Celaya, A. Saxena, P. Wysocki, S. Saha, and K. Goebel, “Towards prognostics of power MOSFETs: Accelerated aging and precursors of failure,” in Proc of Annual Conference of the Prognostics and Health Management Society, Portland, Oregon, Oct. 2010.

[82] N. Patil, D. Das, K. Goebel, and M. Pecht, “Identification of failure precursor parameters for insulated gate bipolar transistors (igbts),” in Proc of International Conference of Prognostics and Health Management, Denver, CO, Oct. 2008, pp. 1–5.

[83] O. V. Thorsen and M. Dalva, “A survey of faults on induction motors in offshore oil industry, petrochemical industry, gas terminals, and oil refineries,” IEEE Trans. Ind. Appl., vol. 31, no. 5, pp. 1186–1196, Sep./Oct. 1995.

[84] Rectifier, International. "Application Note AN-990." IGBT Characteristics.

[85] “Examples for failures in power electronics systems”, ecpe, 20072

APPENDIX A: MATLAB CODE

A.1 Health Monitoring Algorithm Code

KPI Classifier Code

```
% 'filename' is used as generic filename in this example of the KPI classifier

data = filename(: ,:);
[ Ureduced ,psi , lambda ] = defineEigenSpace2(data ');
[ E_filename , E2_filename] = KPI2E( data28 , Ureduced , psi , lambda );
```

Function Used to Calculate Error from KPIs

```
function[E,E2] = KPI2E (data , Ureduced , psi , lambda )


        omegaMatrix = eigenParams2(data ',0,Ureduced ,psi , lambda );
        data_hat = bsxfun (@plus , omegaMatrix* Ureduced ',psi ');
        e = data - data_hat;
        E = zeros( size(e ,1) ,1);
        E2 = zeros ( size(e ,1) ,1);


        for ii = 1: size(e ,1)
                vec = e(ii ,:);
                E(ii) = norm( vec);
                E2(ii) = norm(vec)/ norm( data(ii ,:));
        end
end
```

Function Loads or Calculates Eigenspace Basis Vectors, Ureduced, and the Mean Vector

```
% This fuction loads or calculates the eigenspace basis
% vectors , Ureduced , and also the mean vector , psi. Psi
% must be subtracted from evt transients before being
% projected into the eigenspace defined by Ureduced.
%
% If no inputs are given the function will lookfor and
% load Ureduced and psi from a EIGEN_SPACE. mat file in
% the current directory.
%
% [ Ureduced ,psi , lambda ] = defineEigenSpace ();
%
% If inputs are given , Ureduced and psi will be
% calculated and then saved to EIGEN_SPACE. mat in the
% current directory. This will write over any existing
% EIGEN_SPACE_2. mat file.
%
%* USE 'eigenParams.m' for projecting evts into eigenspace.*
%
% function [ Ureduced ,psi , lambda ] = defineEigenSpace2( gammaMatrix)
%
% Inputs :
% gammaMarrix , a the complete set of test sample features
% in a DxM matrix . Where D is the dimensions of the
% feature vectors , and M is the number of sample vectors.
%
% D = number of pts in vector
% M = number of training Vectors
% Ufull , Dx(D or M) matrix of eigenvectors , min(D,M)
% lambda , (D or M)x1 vector of corresponding eigenvalues , min(D,M)
% psi , a Dx1 image mean vector
%
% Nt = number of test vectors
% K = reduced dimensions from D (K<D)
% PhiMatrix , DxM zero - mean traing vectors
% Ureduced , DxK matrix of eigenvectors
% omegaMatrix is MxK matrix , K reduced dimensions ,
% projected training vectors => the features for
% each training vector


function [ Ureduced ,psi , lambda ] = defineEigenSpace2( gammaMatrix)


        if ~ exist ('gammaMatrix','var')
        disp('Loaded current EIGEN_SPACE_2.mat ')
        load('EIGEN_SPACE_2. mat'); % this loads Ureduced and psi
        return
end

% Setting Full Space
```

```
[ Ufull , lambda , psi] = computeFullEigenSpace ( gammaMatrix);
PhiMatrix = bsxfun (@minus , gammaMatrix ,psi); % zero mean image g
→vectors
[ omegaMatrix , Ureduced] = reduceEigenSpace(Ufull ,lambda , PhiMatrix);

%%% note: omegaMatrix = PhiMatrix1 '* Ureduced;

% Saving

save(' EIGEN_SPACE_2. mat',' Ureduced','psi','lambda ');
disp('Saved new EIGEN_SPACE_2. mat')
end

function [Ufull , lambda , psi] = computeFullEigenSpace( gammaMatrix)
% Computes the Eigenspace vectors/ values , and the mean
% input vector for a set of M training vectors with D
% dimensions.
%
% [Ufull , lambda , psi] = computeFullEigenSpace( gammaMatrix)
%
% Input ,
% gammaMarrix , a the complete set of test images in a
% DxM matrix .
% Where D is the dimensions of the image vectors , and
% M is the number of image vectors.
%
% Output ,
% Ufull , a DxM matrix of eigenvectors of the full eigenspace ( ||g
→Ufull (:,i)|| = 1 ).
% lambda , a Mx1 vector of eigenvalues (in descending order ).
% psi , a Dx1 image mean vector .

[D,M]= size( gammaMatrix);
% average the image vectors (2nd dim)
psi = (1/M)*sum( gammaMatrix ,2);

% zero mean image vectors
PhiMatrix = bsxfun (@minus , gammaMatrix ,psi);

if D>M
        C = (1/M)* PhiMatrix '* PhiMatrix; % if D>M
else
        C = (1/M)* PhiMatrix* PhiMatrix '; % do it this way for M>D
end

[vec ,d]= eig(C);
[ lambda , index] = sort( diag(d),'descend');
% sorting eigenvectors to correspond with sorted eigenvalues
U= vec(:, index );
% if eigenvalue is <= 0 set corresponding eignevector to zero(g
→numerical instability)
U(:, lambda <=0) =0;
```

```
if D>M
        % if D>M also makes || Ufull (:,i) || = 1
        Ufull = bsxfun (@times ,((1./(M* lambda )) .^0.5) ',PhiMatrix*U);
else
        % do it this way for M>D
        Ufull = U;
end
end

function [ omegaMatrix , Ureduced] = reduceEigenSpace(Ufull ,lambda ,
→PhiMatrix)
% Reduces the Eigenspace dimensions from M to K (K<M<D);
% eigenvectors are D dimensions.
%
% [ omegaMatrix , Ureduced] = reduceEigenSpace( Ufull , lambda , PhiMatrix
→)
% Input ,
% Ufull , DxM matrix of eigenvectors for the full eigenspace.
% lambda , Mx1 vector of corresponding eigenvalues.
% PhiMatrix , DxM zero - mean traing vectors , M vectors of D dimension
→.
%
% Output ,
% omegaMatrix , MxK matrix of reduced eigenspace training
% vectors (K coordinates for each training vector M).
% Ureduced , DxK matrix of eigenvectors associated with
% the K dimensional eigenspace.

[D,M] = size( Ufull );
avgVal = (1./M).* sum( lambda );
Ureduced = Ufull (:, lambda > avgVal )

% must have at least 2 dimensions.
if size( Ureduced ,2) <2
        Ureduced = Ufull (: ,1:2);
end

% projecting each training vector into reduced eigenspace.
omegaMatrix = PhiMatrix '* Ureduced;
end
```

Function Projects Measurements Into Eigenspace Defined by Ureduced and the Mean Vector

```
% This function projects measurements into an eigenspace
% defined by Ureduced and the mean vector psi. Psi is
% subtracted from evt transients to make them zero - mean
% before being projected into Ureduced.
%
%* USE 'defineEigenSpace.m' to define Ureduced and psi *
%
% If Ureduced and psi not inputted , will load them from
% EIGENSPACE. mat
%
% function omegaMatrix = eigenParams( evtlist , white , Ureduced ,psi ,g
→lambda )
%
% white = 1/0 , 1 if you want the eigenvector to undergo
% a whitening transform , this makes each of the
% cordinates to have similar values . Otherwise the
% omegaMatrix would be heavily weighted to the first
% few eigenvectors. If white is not given will be set
% to 0.
%
% D = number of pts in vector
% M = number of training Vectors
% Ufull ,Dx(D or M) matrix of eigenvectors , min(D,M)
% lambda ,(D or M)x1 vector of corresponding eigenvalues , min(D,M)
% psi , a Dx1 image mean vector
%
% Nt = number of test vectors
% K = reduced dimensions from D (K<D)
% PhiMatrix , DxM zero - mean traing vectors
% Ureduced , DxK matrix of eigenvectors
% omegaMatrix is MxK matrix , K reduced dimensions ,
% projected training vectors => the features
% for each training vector

function omegaMatrix = eigenParams2( gammaMatrix ,white , Ureduced ,psi ,lambda )
        if ~ exist ('Ureduced','var')
                load('EIGEN_SPACE_2. mat');
                % this loads Ureduced , psi , and lambda
        end
        if ~ exist ('white ','var')
                white = 0;
        end
                % zero mean image vectors
                PhiMatrix = bsxfun (@minus , gammaMatrix ,psi);
                omegaMatrix = PhiMatrix '* Ureduced;
        if white == 1
                unweight_lambda = lambda (1: size( omegaMatrix ,2)).^ -0.5;
                omegaMatrix = bsxfun (@times , omegaMatrix , unweight_lambda ');
        end
end
```

A.2 Drive KPI Extraction

Drive KPI Extraction Function Call

```
fileLoca = ''; % File location of all collected data files



% Kiserord returns a filter order n and beta parameter to specify a
Kaiser window for use with the fir1 function. Given a set of
specifications in the frequency domain, kaiserord estimates the minimum
FIR filter order that will approximately meet the specifications.
kaiserord converts the given filter specifications into passband and
stopband ripples and converts cutoff frequencies into the form needed
for windowed FIR filter design.



[ni,Wn,beta,ftype] = kaiserord( [1100 6000], [1 0], [0.01 0.1], 1000000
);

b = fir1(ni,Wn,ftype,kaiser(ni+1,beta));

a = length(b);

% Variable declarations
avhh = [];
avdd = [];
nn = 0:9999;
errmat = [];
Icavy = [];
data = [];
dat = [];
y7 = [];
y8 = [];
y9 = [];
y71 = [];
y81 = [];
y91 = [];
Qgck = 1e-7;
vthck = 6.5;
vgeck = 14;

runNum = 1; % Number of files saved

n = 1;



for runNum = 1:1650

% Called saved files from DT9816
filename=[fileLoca,'000_GR0_T045_R02_H_D_FS',num2str(runNum),'.raw'];
```

```
% Called saved files from Tektronix 3014b
filename2 =[fileLoca,'000_GR0_T045_R02_H_O_FS',num2str(runNum),'.raw'];
    if(exist(filename,'file'))
        x = load(filename);
    else
        break;
    end

    if(exist(filename2,'file'))
        h = load(filename2);
    else
        break;
    end

    clear filename
    clear filename2



    % Call for Drive KPI Extraction for DT 9816 Function
    for n = 1:30 %n = 1
    K = ((5000*n-5000)+1:5000*n);
    y= KPIConv3_2(x,K);

    data = [data; y];
    n = n+1;
    end

    n = n-1;
%   h = medfilt1(h,100);
    Vgeh = h(:,1);
    Vce = h(:,2);
    Ic = h(:,3);
    Vgx = h(:,4);


%% The remainder of this handles the data from the faster sampling   3
% Tektronix and the combination of the two sampling devices
% scope
% Section for finding Qg Gate Charge Not discussed in Thesis, but
needed for operation of code



    Vgxx = Vgx/10;

%    Vgxx = smooth(Vgxx);

    t = 0.0:0.0000000002:0.0000019999;

    VSI= find(Vgxx > 0.065);

    ck = length(VSI)
```

```
if (ck < 1)
Qgx = Qgck;

end

if (ck >= 1) % Checks for Errors in data collection
VSI1 = VSI(1);
t2 = t(VSI1):0.0000000002:0.0000019999;
VSii = VSI1:10000;
t2 = transpose(t2);

Vgxxsl = Vgxx(VSii);

Qgx = trapz(t2,Vgxxsl)
Qgck = Qgx;
end

t = transpose(t);
Qg = trapz(t,Vgxx) % Qg from scope

m = ones(n,1);

y881 = (Qgx * m)*1.0e7;

y81 = [y81; y881];


y88 = (Qg * m)*1.0e7;

y8 = [y8; y88];

clear Qgx VSI VSI1 t2 VSii Vgxxsl

%% Section for finding VT
% This section lines deals with the filtering and finding 75% of
% the max kiserord filtered Ic. This is then used to find
% the Vge value at which the current takes off an the IGBT turns
% on.
hh3 = filter(b,1,filter(100e6*[1 -1],1,Ic)); % filtered Ic

len = length(hh3);

len1 = 600:len;

hh3 = hh3(len1); length of Kiserord filtered Ic

Vgeh = Vgeh(len1); % Vge vector matching length of Kiserord
% filtered Ic
```

```
hmm  = max(hh3); % Max Ic

hmm = hmm * 0.75; % 75% of Max Ic


p = find(hh3 > hmm & Vgeh > 3); % 75% of Max Ic

mh = (p - (a/2)) - 30; % Indices for approximate take of point of
% Ic


lmh = length(mh);

vthh = []

if (ck < 1) % Checks for Errors in data collection
VTH6 = vthck; % Faulty data VT used from previous run
errmat = [errmat; runNum]; % Faulty data catalog
end

if (ck >= 1) % Uses Vge vector to find all possible turn on values
for i = 1:3

     ii = mh(i);
     ii = round(ii);
     if Vgeh(ii) < 9.5
     vthh = [vthh; Vgeh(ii)];
     end
end

mhrev = ii(1) : ii(1) + 850;

Vgehrev  = Vgeh(mhrev);

VTH6 = mean(Vgehrev) % VT

vthck = VTH6; % If data is healthy VT is recorded from current run

end


y77 = (avthh * m);  %     Vector for 3 second intervals of VT

y7 = [y7; y77];     %     Vector for VT for all runs

clear i vgeh Ic p mh lmh ii vthh  avthh mhrev Vgehrev VTH6

%% Section for finding Vge,on,Avg

if (ck < 1) % Checks for Errors in data collection
VgeAvg = vgeck;
```

```
    end

    if (ck > 1)
    VgehL = length(Vgeh);
    VgeL = VgehL - 200: VgehL;
    VgeN = Vgeh(VgeL);

    VgeAvg = mean(VgeN)

    vgeck = VgeAvg;      % Vge,on,avg

    end

    m = ones(n,1);

    y991 = (VgeAvg * m); % Vector for 3 second intervals of Vge,on,avg


    y91 = [y91; y991];  % Vector for Vge,on,avg for all runs

    clear VgeAvg

%
    clear y x K y77 y88 y 99 Qg Qgm Vgxx Vth Vthm h Vge Vce Ic Icx m
hmm y771 y881

end
    data0 = [data, y71, y91, y81 , y7, y8] % Vector with all KPI for
% current 3 seconds of data
    errmat
    clear y7 y8 Qgck;

save '-v7' BD_6_000_GR0_T045_R01_H3.mat data0
```

save out.kpi data -ascii -tabs -double ;

Drive KPI Extraction for DT 9816 Function

```
function [y z] = KPIConv3_2(x,K)

%Scale raw data
mn = mean(x(K,1)); % mn is the raw Ic peak to peak value

w1 = -2.4*(x(K,1)-mn);
w2 = medfilt1(x(K,2),3);
w3 = medfilt1(x(K,3),3);
w4 = medfilt1(x(K,4),5);
w5 = medfilt1(x(K,5)*100,3);
w6 = medfilt1((x(K,6)-.5)*100,3);


%Edge Detector
x1 = medfilt1(x(K,3),5);
dx1 = filter([1 -1],1,x1);
dx2 = zeros(1,length(dx1));
dx2(dx1 < -1) = -1;
dx2(dx1 > 1) = 1;
dx3 = filter([1 -1], 1, dx2);
locs = find(dx3 ~= 0);
locs = locs(1:2:end);
slopes = dx3(locs);
start_ind = find(slopes == 1);
end_ind = start_ind+1;


OUTPUT = [];



Vc = [];
Vc2 = [];
Icc = [];
Icc2 = [];

for m = 1:length(start_ind)-1
N = locs(start_ind(m))+2:locs(end_ind(m))-2;
Ic = w1(N);
Icc = [Icc; Ic];
VCEon = w2(N);
Vc = [Vc;VCEon];
R = find(VCEon > 0.6 & Ic > 0);
Ic2 = Ic(R);
Icc2 = [Icc2;Ic2];
VCEon2 = VCEon(R);
Vc2 = [Vc2;VCEon2];
OUTPUT(end+1:end+length(R),[1 2]) = [Ic2(:) VCEon2(:)];

end
```

```
OUTPUT2 = sortrows(OUTPUT,1);


M = 1:size(OUTPUT2,1);
py = OUTPUT2(M,1);
px = OUTPUT2(M,2);
px1 = medfilt1(px,10);
py1 = medfilt1(py,10);

pxx = find(px1 > .6);

px1 = px1(pxx);
py1 = py1(pxx);

P = polyfit(py1,px1,1);


Irms = sqrt(filtfilt(1/833*ones(833,1), 1, w1.^2));

y1 = (mean(Irms(1:length(K)-1500)))*10; %Irms
y2 = (P(:,1))*10; % Ron
y3 = P(:,2); % Vj
y4 = (mean(w4)); %Qi
y5 = (mean(w5))/10; %TC
y6 = (mean(w6))/10; %TA


y = [y1,y2,y3,y4,y5,y6];


end
```

A.3 Data Collection Code

Control and Recording code for DT9816 and Tektronix 3014b

```
%% Create a VISA-TCPIP object.
interfaceObj = instrfind('Type', 'visa-tcpip', 'RsrcName',
'TCPIP0::10.16.18.15::inst0::INSTR', 'Tag', '');

% Create the VISA-TCPIP object if it does not exist
% otherwise use the object that was found.
if isempty(interfaceObj)
    interfaceObj = visa('TEK', 'TCPIP0::10.16.18.15::inst0::INSTR');
else
    fclose(interfaceObj);
    interfaceObj = interfaceObj(1);
end

% Create a device object.
deviceObj = icdevice('tektronix_tds2024.mdd', interfaceObj);

% Connect device object to hardware.
connect(deviceObj);

% Auto Set Fucntion
%invoke(deviceObja, 'autoset');

%% Set Control Mode

% Set TimeBase
set(deviceObj.Acquisition(1), 'Timebase', 1e-08); % 1e-08 2.5e-05


% Configure property value(s).
set(deviceObj.Acquisition(1), 'Control', 'single');

%%  Set Up Trigger Action

d = 0;
b = 50;
g = 0;
ggf = 0;
T = 0;
f = 0;
u = 0;
i = 1; % starting run number
TF = 0;
TFa = 0;
TF2 = 0;
TF2a = 0;
% Set comparartor variables
s2 = 'save';
s3 = 'trigger';
```

```
%% Set up DT9816

ai0 = analoginput ('dtol', 0);
addchannel(ai0, 0:5);
fsample = 50000; %sampling frequency in Hz
N = 3; %number of seconds to sample
set(ai0,'SampleRate',fsample);
set(ai0,'SamplesPerTrigger',fsample*N);
set(ai0, 'BufferingMode', 'Auto');


%%
while (T < 1)
    % tell device to start running again
    set(deviceObj.Acquisition(1), 'State', 'run');

    while (TF2 < 1) % Interupt for 'Trigger' state
        check = get(deviceObj.Trigger(1), 'State') % Retrieves OSCope
Trigger state
        TF2 = strcmp(check,s3) % comparator for 'triiger' flag

        if TF2 == 1
            % DT9816 Start Recording
            start(ai0);
            [data, time] = getdata(ai0);
            stop(ai0);
            u = u + 1 % Trigger State Counter
        end

        if g >= 20 && TF2 ~= 1
          set(deviceObj.Acquisition(1), 'State', 'run');
          clear data time;
          g = 0;
        end
        g = g + 1
    end

    while (TF < 1) % interupt for 'Save' state
        check1 = get(deviceObj.Trigger(1), 'State') % Retrieves OSCope
Trigger state
        TF = strcmp(check1,s2) % comparator for 'save' flag

        g = 0;
    end


 %%
    if TF == 1 && u > 0 % Checks for state of completion sampling for
Oscilloscope and DT9816
        u = 0; % Resets Trigger State Counter
        ggf = 0;
        TF2 = 0;
```

```
%         TF2a = 0;
        TF = 0;
        TFa = 0;
        Tempdata = data;

        % Reading Waveform Channel 1 from Oscope
        groupObj = get(deviceObj, 'Waveform');
        groupObj = groupObj(1);
        [Y,X,YUNIT,XUNIT] = invoke(groupObj, 'readwaveform',
'channel1');

        % Reading Waveform Channel 2 from Oscope
        groupObj = get(deviceObj, 'Waveform');
        groupObj = groupObj(1);
        [Y1,X1,YUNIT,XUNIT] = invoke(groupObj, 'readwaveform',
'channel2');

        % Reading Waveform Channel 3 from Oscope
        groupObj = get(deviceObj, 'Waveform');
        groupObj = groupObj(1);
        [Y2,X2,YUNIT,XUNIT] = invoke(groupObj, 'readwaveform',
'channel3');

        % Reading Waveform Channel 4 from Oscope
        groupObj = get(deviceObj, 'Waveform');
        groupObj = groupObj(1);
        [Y3,X3,YUNIT,XUNIT] = invoke(groupObj, 'readwaveform',
'channel4');


        ch1 = transpose(Y);  % Vge Amplitude
        ch2 = transpose(Y1); % Vce Amplitude
        ch3 = transpose(Y2); % Colllector Current
        ch4 = transpose(Y3); % Gate Current

        channels = [ch1,ch2,ch3,ch4];

        % saved file for DT9816 for 3 second intervals
        str=sprintf('save 000_GR0_T045_R01_H_D_Data%d.raw Tempdata -
ascii;',i)
        eval(str);
        % saved file for Tektronix 3014b for 3 second intervals
        str2 =sprintf('save 000_GR0_T045_R01_H_O_Data%d.raw channels -
ascii;',i)
        eval(str2);

        clear data check check1 checka time start() stop();
        clear str;
        clear Tempdata
        clear Y Y1 Y2 Y3 Ya Yb Yc Yd;
        clear X X1 X2 X3 Xa Xb Xc Xd;

        i = i + 1;
    end
```

```
    if i >= 11  % Total number of runs wanted
           T = 2; % Trigger State Counter
    end

     f = f + 1


end
```