3-12-2015

A Generalized Logic-Based Approach for Intelligent Fault Detection and Recovery in Power Electronic Systems

Weiqiang Chen
University of Connecticut - Storrs, weiqiang.chen@uconn.edu

Recommended Citation
https://opencommons.uconn.edu/gs_theses/726
A Generalized Logic-Based Approach for Intelligent Fault Detection and Recovery in Power Electronic Systems

Weiqiang Chen

B.E., Nanjing University of Information Science & Technology, 2012

A Thesis
Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science
At the
University of Connecticut
2015
APPROVAL PAGE

Masters of Science Thesis


Presented by

Weiqiang Chen, B.E.

Major Advisor_______________________________________________________________
Dr. Ali Bazzi

Associate Advisor___________________________________________________________
Dr. Sung-Yeul Park

Associate Advisor___________________________________________________________
Dr. Shalabh Gupta

University of Connecticut

2015
TABLE OF CONTENTS

I. Introduction ........................................................................................................................................... 1
   1.1 Motivation ........................................................................................................................................... 1
   1.2 Challenges ........................................................................................................................................... 1
   1.3 Contribution ....................................................................................................................................... 3

II. Literature Review ..................................................................................................................................... 5
   2.1 Review of Major Power Electronic Components’ Fault Modes ....................................................... 5
   2.2 Review of Fault Diagnosis Methods Applied to Power Electronic Converters ......................... 9
      2.2.1 Wavelet Transformation .............................................................................................................. 9
      2.2.2 Neural Networks and Pattern Recognition .................................................................................. 11
      2.2.3 Fuzzy logic ............................................................................................................................... 14
      2.2.4 Dependency Matrixes ............................................................................................................... 15
   2.3 Literature Review Summary .............................................................................................................. 18

III. Proposed Methodology .......................................................................................................................... 20
   3.1 Proposed Approach Using Combinational Logic ............................................................................ 20
      3.1.1 Mathematical Description ......................................................................................................... 20
      3.1.2 System Recovery and Approach Flowchart ........................................................................... 24
   3.2 Proposed Approach Using Fuzzy Logic ............................................................................................ 26
      3.2.1 Difference between Combinational and Fuzzy Logic Approaches ........................................ 26
      3.2.2 Fuzzy Logic-Based Approach and Its Flowchart .................................................................. 26
      3.2.3 Illustrative Example ................................................................................................................ 28

IV. Micro-inverter Platform: Nominal Operation & Fault Injection .............................................................. 32
   4.1 Basic Topology .................................................................................................................................. 32
      4.1.1 Platform Description ................................................................................................................. 32
      4.1.2 Fault Injection ............................................................................................................................ 36
4.2 Simulation Model .................................................................................................................. 36
4.3 Experimental Setup.................................................................................................................. 37
4.4 MODEL VERIFICATION ......................................................................................................... 40
   4.4.1 Nominal/Healthy Operation .......................................................................................... 40
   4.4.2 Operation under faults ................................................................................................. 41
V. Implementation of the Proposed Approach ......................................................................... 43
   5.1 Combinational Logic Implementation ............................................................................. 43
      5.1.1 Logic Gates and Tables ......................................................................................... 43
      5.1.2 Example Results ................................................................................................. 46
   5.2 Fuzzy Logic Implementation ............................................................................................ 50
      5.2.1 Implementation in Simulink .................................................................................. 50
      5.2.2 Example Results ................................................................................................. 53
VI. Implementation Challenges ................................................................................................. 57
VII. Conclusions and Future Work ............................................................................................ 59
Reference .................................................................................................................................. 61
Appendix A: Publications .......................................................................................................... 64
Appendix B: Simulation Models .................................................................................................. 65
   Micro-inverter: ...................................................................................................................... 65
   Combinational logic quantity calculation: .......................................................................... 66
   Fuzzy logic quantity calculation subsystem and fuzzy controllers: .................................. 67
Appendix C: Real-time Control .................................................................................................. 70
   PI current and voltage closed-loop control: ....................................................................... 70
   Digital signal bank: .............................................................................................................. 72
   Analogy signal bank: ............................................................................................................ 72
Combinational logic measurements \( V_{dc} \), \( V_{S2} \), and \( V_{D2} \) and injected faults \( L\ S\ C \), \( S\ 2\ \text{OC} \), and \( SB'\ \text{OC} \): ................................................................. 73

Fuzzy logic with partial MFs: ........................................................................................................ 74
LIST OF FIGURES

Fig. 1. Boost converter example
Fig. 2. Example of failed component a) Failed MOSFET b) Failed capacitor
Fig. 3. Boost converter – Proposed methodology-based control
Fig. 4. Wavelet-theory-based fault detection
Fig. 5. Layer structure of neural network
Fig. 6. Fuzzy controller with two inputs and one output
Fig. 7. Measurements and related quantities
Fig. 8. Components’ fault conditions
Fig. 9. Fault recovery using redundancy
Fig. 10. Flowchart of the proposed approach for combinational logic
Fig. 11. Fuzzy logic based methodology correlation system
Fig. 12. Flowchart of the proposed approach for fuzzy logic
Fig. 13. Example of fuzzy controller
Fig. 14. Example of fuzzy controller with different fuzzy rules
Fig. 15. Example correlation system
Fig. 16. Platform utilized to demonstrate the proposed approach
Fig. 17. Proposed approach for fault diagnosis and recovery
Fig. 18. Micro-inverter platform used to test the proposed approach
Fig. 19. Fault locations highlighted on the power side of the platform
Fig. 20. Component fault injection strategy
Fig. 21. Experimental platform
Fig. 22. High power supply
Fig. 23. Micro-inverter power board
Fig. 24. FPGA board
Fig. 25. Interfacing panel
Fig. 26. Transient of Micro-inverter DC voltage and AC voltage a) Simulation result b) Experimental result
Fig. 27. Steady-state of Micro-inverter DC voltage and AC voltage a) Simulation result b) Experimental result
Fig. 28. Micro-inverter DC voltage and AC voltage under MOSFET $S_2$ OC condition
Fig. 29. Micro-inverter DC voltage and AC voltage under Diode $S_B$’ OC condition
Fig. 30. Micro-inverter DC voltage and AC voltage under $D_2$ OC condition
Fig. 31. Design for signal maintenance
Fig. 32. An example design for a unique combination using combinational logic fault diagnosis
Fig. 33. Simulation: System survival by detecting $C_{out}$ SC and engaging a spare capacitor
Fig. 34. Simulation: System survival by detecting the $S_2$ OC and engaging a spare MOSFET
Fig. 35. Experimental: System survival by detecting the $L$ SC and engaging another Inductor in place
Fig. 36. Experimental: System survival by detecting the $S_2$ OC and engaging another MOSFET in place
Fig. 37. Experimental: System survival by detecting the $S_B$’ OC and engaging another MOSFET in place
Fig. 38. Simulation validation of $L$ SC experimental results
Fig. 39. Simulation validation of $S_2$ OC experimental results
Fig. 40. Simulation validation of $S_B \cdot OC$ experimental results
Fig. 41. Fuzzy toolbox interface
Fig. 42. Fuzzy logic controller (fault diagnosis) in Simulink
Fig. 43. Fuzzy-logic-based fault diagnosis implementation in simulations
Fig. 44. Output Membership function
Fig. 45. Input membership function
Fig. 46. Simulation: System survival by detecting the $C_{out} \cdot SC$ and engaging another capacitor in place with fuzzy logic control
Fig. 47. Simulation: System survival by detecting the $S_2 \cdot OC$ and engaging another capacitor in place with fuzzy logic control
Fig. 48. Experiment: System survival by detecting the $S_2 \cdot OC$ and engaging another capacitor in place with fuzzy logic control
Fig. 49. Experiment: System survival by detecting the $S_B \cdot OC$ and engaging another capacitor in place with fuzzy logic control
Fig. 50. Simulation validation of $S_2 \cdot OC$ experimental results with fuzzy logic control
Fig. 51. Simulation validation of $S_B \cdot OC$ experimental results with fuzzy logic control
LIST OF TABLES

Table 1. Example of combinational logic combination for $M$ measured quantities and $Y$ faults
Table 2. System parameters
Table 3. Value of Components
Table 4. Non-ideal parameters and values
Table 5. Correlation between faults and change of measured signals from nominal operation (O is for OC and S is for SC)
Table 6. Combinational Logic fault diagnosis and system recovery time
Table 7. Fuzzy rules
Table 8. Fuzzy Logic fault diagnosis and system recovery time
I. INTRODUCTION

1.1 Motivation

Power electronics is the application of electronic components and devices for the control and conversion of electric power. Power electronics applications have extended from renewable energy interfacing with the utility grid to electric traction. With energy conversion in the power electronic stage, several problems may occur due to internal and external reasons. Power electronic systems thus need to be sustainable for these problems by ensuring power availability during the power conversion process and utilizing proper topologies to satisfy application requirements.

Renewable energy is usually defined as energy that comes from resources which are naturally supplemented such as sunlight, wind, rain, tides, waves and geothermal heat. In many renewable energy applications, the source is typically of direct current (DC) type while the load or grid side is commonly of an alternating current (AC) type. Faults can occur within the power electronic system, on the source side, load side, or grid side, as applicable and depending on the application. If essential components fail in the power converter, unavoidable fault conditions may cause fault propagation across the system and lead to cascaded series of faults. Thus, fault detection and fault diagnosis should be included in each power electronic system to prevent such failure, in other words, to increase the system’s stability and reliability.

1.2 Challenges

Each component in a power electronic converter has its irreplaceable function and importance. Conventional power electronic circuits do not typically have the capability to survive from any failure except with added components or added control and diagnosis functionality. A motivational example is a typical DC/DC boost converter operated in continuous
conduction mode (CCM) as shown in Fig. 1 where $L$ is the inductor’s inductance, $S$ represents a MOSFET (with built-in reverse diode), $D$ represents a diode, and $C_{out}$ represents capacitance of the output capacitor. The small green squares represent sensors for voltage and current measurements. This topology can boost the input DC voltage to higher output DC voltage applied to the load. The voltage and current controls form a closed-loop control system to track a desired output voltage reference value. For instance, if the output capacitor fails as an open circuit shown in a red spark in Fig. 1, high voltage ripple will be seen on the output side (load) due to inadequate voltage ripple filtering. Another example is when $S$ fails leading to the control loop losing its ability to control the output voltage value, and if $S$ fails as an open circuit, the DC source will be short circuited. Examples of a failed MOSFET and capacitor are shown in Fig. 2. Note that the diode and inductor can also fail and cause the circuit to fail. Thus, if the circuit in Fig. 1 or any other power electronic converter is implemented in a safety-critical or reliability-critical application, the failure of a component may cause the converter failure, and in turn propagate to cause a larger system failure.

![Fig. 1. Boost converter example](image-url)
Conventional power electronic converters face high risks of component failure which may be irretrievable or irreversible. This thesis thus proposes a hybrid approach of recovering a converter from component failure through engaging redundant components using logic-based fault diagnosis. Using the conventional circuit and closed-loop control, more functional controls are introduced as summarized in Fig. 3 which illustrates at a high level the research done as part of this thesis. As shown in Fig. 3, the use of sensor feedback is no longer limited to closed-loop control, but also to provide necessary information to execute the proposed control, in addition to more sensor feedback that might be needed. The shadow of each component illustrates a redundant component that is engaged once a main component fails, to ensure the health of the circuit. The proposed control has three main stages: 1) Pre-processing is applied to the measured signals, such as calculating useful quantities, setting proper tolerances, etc; 2) Different logic-based methods are used for fault diagnosis by correlating the quantities available from the pre-processing stage; and 3) Recovery signals are generated and sent to eliminate the failed component and engage the redundant component.

The proposed methodology is expected to improve the stability and reliability of the power electronic system integrated fault diagnosis. Its fast fault detection, diagnosis, and recovery can avoid the extended unhealthy operational condition of the system. By correlating different quantities available from sensor feedback and comparing them to healthy conditions, robust diagnosis is achieved and avoids making false diagnosis or wrong decisions even with
some unavoidable fluctuation of one or more measured signals. Some tolerances of all the quantities are selected to decrease the impact of inherent harmonic components, noise and unpredictable variations, and the training or design period of the fault diagnosis algorithms is critical but not complex. The proposed methodology is straightforward for implementation on existing digital control platforms with existing controllers, and is forecasted to have significant advantages in hybrid and electric transportation systems, renewable energy systems, and other power electronics applications which cannot tolerate long down time caused by faulty conditions in essential components. Although the redundancy strategy induces the increase of the cost, redundancy is standard practice in safety critical applications and costs less than a complete replacement of a converter when a single component fails. For example, hardened epoxy that fills converter enclosures to tolerate ambient temperature fluctuation and avoid humidity penetration, such as in solar micro-inverters, prevents replacement of specific components and requires a complete replacement of the micro-inverter, but the proposed strategy would include redundant components and intelligent fault diagnosis to engage such components as part of the micro-inverter to increase its lifetime without invasive maintenance.
II. LITERATURE REVIEW

2.1 Review of Major Power Electronic Components’ Fault Modes

When current flows through a component, the current input side of the component is termed an input node (IPN), and the current output side of the component is named as output node (OPN). The abnormal relationship between the IPN and OPN forms two main fault modes, open-circuit (OC) and short-circuit (SC) faults, of power electronic components. An OC fault is when no connection exists between the IPN and OPN, while an SC fault is when the IPN and OPN are tied together directly or through some extremely low-resistance path. Major components which may encounter faulty conditions include power semiconductors, e.g., MOSFETs, insulated-gate bipolar transistors (IGBTs), and diodes, in addition to inductors and capacitors. In a survey of power electronic converters [1, 2], semiconductors were shown to be the most common to fail for numerous reasons, including the surrounding environment of a
power converter being the major root cause. For example, humidity can wet a component causing its failure. Electrical transients were also a main reason for component failure. For example, inrush current commonly accompanies the starting transient of a power electronic converter and may exceed semiconductor device’s rated current causing the device failure. This is similar to heavy loading conditions that may cause similar semiconductor device failure. Therefore, it is important that when a power electronic converter is designed, nominal or steady-state, transient, and faulty conditions should be considered.

Significant work has been done to diagnose switch OC of switching semiconductor devices in multiple systems, such as the matrix converter drive system and the single power switch system. In [3], the fault diagnosis methods relied on measurements and comparisons to healthy conditions. Based on comparisons of nine pairs of the measured input and output voltage, nine error values were created. The voltage errors are assigned to nine bi-directional switches of the matrix converter. If any dedicated voltage error signals exceed the threshold, it is possible to detect and locate the faulty switch. In [4], error is also induced to diagnose the fault where the express is:

\[ e_n = i_n^* - i_n \]

where \( i_n^* \) is the reference current value in a matrix converter phase, \( i_n \) is the measured value, and \( e_n \) is the error. All the errors were compared with a threshold to make a decision of the fault diagnosis. In the single switch system, only one current measurement was needed and fault diagnosis was simpler. The post-processing of the measured current included the absolute and average values of the measured current, in addition to the average error value. The final diagnostic variables \( d_n \) were calculated as,
where $\langle |i_n| \rangle$ is the average value of the absolute value of the phase current and $\langle e_n \rangle$ is the average error value. In [5], diagnostic variables were fed into a fault detection and localization system where wavelet theory was used to decompose the current component to obtain some necessary coefficients. By investigating the root mean square (RMS) value of these coefficients, the feature of the single-switch or double-switch OC fault could be distinguished and located. Another intelligent method to avoid switch SC fault damaging the power electronic system was through bypassing the faulty switch. The approach in [6] was applied to a multi-level converter that inherently survives OC faults, thus SC faults were masked in [6] to look like OC faults which are survivable. Fuses and relays were used to isolate the SC switch and converter, the SC fault to an OC fault leading to survival of the converter. Another method, remedial action, was also presented in the literature [7]. This method is heavily reliable on the converter topology and was applied to a phase-shifted full-bridge converter. Basically, the first step of its fault diagnosis is by measuring and combining real-time criteria and gate driver signals for semiconductor switches. Then proper actions were taken to reconfigure the converter and adjust the output voltage using various steps including a spare switch. Diode failures have also been shown to include OC and SC conditions. Diode failures could cause the distortion of the current waveform since current can flow in both directions in the location where a diode SC occurs, or will stop flowing if an OC occurs. The Fast Fourier Transformation (FFT) is a typical transformation used with current sensor feedback for diode failure detection. The distorted current was transformed in [20, 21], by FFT to extract useful components that are fed into certain logic to realize fault diagnosis. By
using AC ripple current as a source of information, harmonic analysis was able to detect diode failures.

In addition to semiconductors, the OC fault and SC faults may occur in other components. A common cause for inductor SC fault is temperature rise due to overload or when induced from other faults [8]. High temperatures could cause winding insulation to melt and thus short circuits between all or some inductor turns, leading to a decrease in inductance. Impacts of an inductance decrease include increase in a converter’s (or filter’s) resonant frequency and undesired voltage tracking in a closed-loop control. In [8], a control strategy to detect inductor SC faults is proposed by using the critical and resonant frequencies of a converter. Once a SC fault occurs, the resonant frequency would suddenly increase, and the converter would lose its regulation leading to flagging a fault. Induction motor drives have been very useful in providing insight into inductor and capacitor faults due to their common presence across both industrial and research literature. Induction motor winding faults are very useful in understanding inductor faults, and faults in DC link capacitors in induction motor drives are useful in understanding capacitor faults [9]. Common causes of DC-link capacitor failures include dry soldering, mistakes during system integration and manufacturing, material stress due to temperature cycling and environmental conditions, in addition to inadequate sealing that can cause dryness of dielectric material or humidity leaks. An OC fault in a DC-link or filter capacitor could increase current harmonics after the capacitor in addition to voltage ripple across the capacitor terminals. Spikes caused by current ripple could damage other components in a power electronic system such as semiconductors [10]. The SC fault may lead abnormal currents passing through other components including damage to the power source with abnormal currents mainly having overshoots [11]. Threshold-based fault diagnosis for capacitor OC was used in the literature
where only capacitor faults were considered, which made fault diagnosis easy [12]-[14], with a major measurement being the ripple voltage across the capacitor. Temperature rise in capacitors is another measurement useful for fault diagnosis as shown in [15]-[18]. A combination of measurements, sensor fusion, and other sensor feedback provided better decision making, such as by combining voltage ripple, load condition, and power source condition [19]. Compared to traditional systems, many emerging power electronic applications have more emphasis and need for fault tolerance, e.g., more-electric aircraft, submarines, satellites, hybrid and electric vehicles, and others, where faults could cause catastrophic failures. Intelligent fault detection and recovery thus has essential priorities in present and future power electronic systems.

2.2 Review of Fault Diagnosis Methods Applied to Power Electronic Converters

2.2.1 Wavelet Transformation

Significant work in the area of fault detection and diagnosis has already been established. The work includes the utilization of wavelet theory where sensor feedback waveforms are transformed into more useful waveforms with more obvious features that help with fault diagnosis. Wavelet theory has been essential in power grid applications where by analysing fast transient measurements under faulty conditions, and with the help of some monitoring devices, the fault could be located and the distance of the fault from the reference point could be measured. The discrete wavelet transform has also been commonly used where measured signals are decomposed at several different levels. If a fault event occurs, the decomposed signals would be compared to healthy ones to detect and locate the fault, otherwise, the transformation and comparison would keep tracking online [22]. The most common application of wavelet theory has been in the enhancement of the power quality due to its ability to analyse harmonics of various measurement of a power system. In power electronic circuits, such as DC/DC converters
and DC/AC inverters, wavelet theory is useful in the presence of inherent harmonics due to PWM or other switching schemes [23]. The wavelet-theory-based fault detection in power electronic systems shows excellent performance with OC and SC faults of power electronic components due to drastic changes in many measurements, and the correlation with time information could reduce the confusion when different faults happened at similar situations. With voltage and current measurements being available in most power electronic converters, wavelet theory can be widely adopted for various power electronic converters. A very typical wavelet fractal method was presented in [24] where major steps are:

1. Before using wavelet theory, measured voltages and currents under fault condition were obtained as time series;
2. After being decomposed using the wavelet theory, different signal sequences of different frequency bands were found;
3. The correlation among signal sequences of different frequency bands was carried out to obtain results with the help of fractal dimension information.

The process is shown in Fig. 4. In [24], the platform was a thyristor-based three-phase inverter with four different fault modes: 1) Only one thyristor failed, 2) Same phase thyristor failed, 3) Same half-bridge thyristor failed, and 4) Cross pair thyristor failed. If the number of failed thyristors is more than one, they may locate at the same phase/bridge leg or different phase/bridge leg where cross failure occurs.
2.2.2 Neural Networks and Pattern Recognition

A pattern recognition algorithm assigns observations into different patterns based on the knowledge of system features. For example, it is effective to assign rectifier switch faults into different fault patterns to diagnose different types and locations of the faults [25]. This method utilizes principal components analysis (PCA) to transform a set of observations of possible correlated variables into the uncorrelated variables by reducing the dimensionality of the observed variables, which are obtained from input three-phase line currents. Meanwhile, all necessary information is not reduced with the reduction of the dimensionality of the variables. Major steps of pattern recognition algorithm in [25] are as follows for a rectifier example:

1. The input rectifier three-phase line currents were measured and PCA is utilized to transform variables;
2. The eigenvalues and eigenvectors of uncorrelated variables were computed;
3. Fault condition judgement by comparing eigenvalues: If eigenvalues were the same, no fault condition happened; if eigenvalues were different, a faulty switch existed;
4. Fault diagnosis by using eigenvectors: The eigenvector principal direction would indicate which phase of the rectifier failed.
A neural network mimics an animal’s central nervous system that it integrates observations it receives, and processes the observations to achieve the object of the system such as classification and regression. Pre-processing is commonly applied to the observations to extract useful information before the observations enter neural network processing system. The process is usually separated into several steps, and each single step is named a “neuron”. The process in the neuron follows the designer’s choice of suitable classifiers or regression functions. To investigate the flow of a working neural network, three layers should be introduced: 1) Input layer, 2) Hidden layer, and 3) Output layer. The proceeding of a neural network starts from system activation by the input layer where the input data is weighted, and then neurons in the hidden layer perform a user chosen computation method and continue to activate all neurons to the end of this layer. Finally, the output layer determines which characteristics should be read. The neural network has not been given a formal definition. Only the features mentioned above can be identified as a neural network. Fig. 5 shows a typical layer construction of a neural network.

Fig. 5. Layer structure of neural network
The neural network is commonly used to solve classification or regression problems, which can be power electronics fault detection and diagnosis, or other complex systems. For example, neural networks were used in [26] to detect and diagnose the rectifier circuit faults. As mentioned before, system features are the basis of a neural network, so the error back propagation (BP) algorithm is used to extract useful features of the rectifier circuit faults. Then the extracted features are feed into the neurons to generate waveforms which relate to different faults. From these waveforms, forward propagation method is used to generate outputs. The generated outputs are compared with the desired output patterns, and then the error of each output is obtained. Afterwards, the errors are transmitted back to intermediate neurons to contribute final outputs for fault detection and diagnosis. This can be summarized as follows:

1. The output voltage and one phase voltage were measured;
2. The phase delay was applied to the first $N_s$ samples of the output voltage;
3. The $N_s$ samples of the output voltage were normalized with respect to the peak value of line voltage;
4. The normalized values were feed into the network for calculation;
5. The output error was generated to feed back to contribute to the final output;
6. The fault results were obtained to form a code table. The fault types could be extracted from the code table to diagnosis faults.

However, if the complexity of a power electronic system increases and even though the number of basic fault types is the same, the increase in number of power electronics devices causes significant growth in the fault universe. In this situation, the numbers of generated error and fault patterns increase simultaneously, thus the effective organization and reconstruction for this multiplicity are dramatically increased. Therefore, a more effective way termed recurrent
neural network (RNN) is provided in [27], to resolve this issue using the BP algorithm but in a different application. In RNN, multiple faults are grouped into small dimension fault pattern vectors by a logical processing. Based on the fault pattern’s value, zero or one state from each element in the vector is given, zero meaning no fault occurrence and one meaning fault occurrence. By utilizing non-linear function mapping, the relationship between fault vectors and fault codes is generated and stored in a network to form a code table. The advantages of this method include: 1) The approach could solve normal problems effectively, 2) The extraction of voltage waveform and trigger angle did not need pre-processing, and 3) It has very high precision and fast detection speed, and is capable of real-time fault detection and diagnosis.

2.2.3 Fuzzy logic

The fuzzy control theory is another approach for fault detection and diagnosis. As stated in [28], basic measurement and comparison are necessary, then the decision of fault conditions is made by using pre-established membership functions. The logic inside the fuzzy controller is called fuzzy logic, which is different from combinational logic with binary decisions for faults. Fuzzy logic has several states across some input ranges, and for a single state, the different membership levels are set to represent a certain input value’s proportion of contribution to the output. Fuzzy rules are set by “if…then” logic which uses the states of inputs as conditions and states of outputs as results. The proportion of contribution from each input is mapped to the output. Finally, the mapped values are combined to generate the output value. Because limiting the number of inputs to a fuzzy controller helps reduce complexity, the fuzzy approach is not very applicable to complex power electronic systems. Three elements form a complete fuzzy controller: 1) Input membership functions (MFs), 2) Fuzzy rules, and 3) Output membership function (MF). The input process is called “fuzzification” and output process is called
“defuzzification”. Fig. 6 shows the structure and the flow of a two-input fuzzy controller. More details will be described in Chapter III.

![Fuzzy controller with two inputs and one output](image)

1. If \( L \) and \( L \), then \( L \).
2. If \( L \) and \( M \), then \( L \).
3. If \( H \) and \( H \), then \( L \).

Different strategies which are related to fuzzy logic were carried out to solve the problem of detecting component failure, and were proved to have good performance for fault detection and diagnosis in systems containing multiple possible faulty conditions. Combining other methods with fuzzy logic is an effective way such as the combination with wavelet theory [29]. In the combined method, the wavelet theory is utilized to transform the output signals under no fault condition and all faulty conditions to obtain different sequence components. Based on these components, a fault dictionary is generated to reflect the circuit status, and these coefficients are fed into the fuzzy system as inputs to generate output for fault detection and diagnosis.

### 2.2.4 Dependency Matrixes

The dependency matrix is similar to the combinational logic based approach which will be introduced in later chapters. It has been utilized in fault diagnosis [41]. Before applying dependency matrix, test points are placed to monitor the system under study. The central part in the dependency matrix is node selection. These nodes are carried out by the combination of the values obtained from test points, and they have binary states as zero (no fault happens) or one (fault occurs). Dependency matrix had some definitions to optimize its performance:
1. Some invalid test points are eliminated if these nodes have no contribution to fault diagnosis results;
2. The conditions called undetectable faults which don’t exist in a dependency matrix should be deleted;
3. Some faults are very confusing if they have same states combinations, these ambiguity faults should be combined.

However, three differences should be noticed between the proposed approach in Chapter III and the dependency matrix approach:

1. All existing dependency matrix applications in the power and energy area are in power transmission systems and no applications are found in power electronic systems.
2. In the proposed methodology, if all kinds of faults can be detected after elimination of some measured quantities, these quantities are eliminated as redundant quantities and the dependency matrix approach does not have this characteristic.
3. After measuring or calculating each quantity, a threshold based on nominal condition will be set in the proposed approach and is used to define correlation between various faults and measured quantities, which is missing in the dependency matrix approach.

2.2.5 Other Intelligent Methods

The random forests and hidden Markov model (HMM) have been utilized for fault diagnosis in [30, 31]. HMM has the basic layers as shown in Fig. 5, the only difference is the intermediate states are not visible, each state has a probability distribution to the output with a certain sequence which gives the information of the hidden states, and in the states, a number of distinct observation symbols are defined. Optimization to some system parameters is performed to fit the way the sequence of observation symbols comes about. Two algorithms are used for
optimization: One is the Baum-Welch algorithm which starts with initial HMM parameter, and the parameter is updated by maximum likelihood calculation and then the probability of the observation sequence is obtained; the second one is forward-backward algorithm where the features of a circuit condition are extracted, and the waveform samples are calculated for all HMMs. The corresponding output which shows the top likelihood is recognized as the state which relates to a certain circuit fault.

The random forest method utilizes auto-regressive (AR) model to extract features of a power electronic circuit, before processing, it optimizes the faulty data by simplifying the structure of the data and improving the speed and precision of the classification. After fault extraction and optimization, data is structured as a tree which is a classifier. The system examines the input vector on each tree in the forest. A unit vote is casted by each tree at the input vector and the vote is processed to the end. At last, highest votes are selected as the fault decision.

Parallel systems in [32, 33] are also a good way to survive a system from failure. Parallel systems require two fully functional systems to work simultaneously in parallel such that when one system fails, the parallel one can continue working with the same functionality. This method ensures a longer system lifetime and increases system reliability. Any fault diagnosis methodologies can be applied to protect the parallel systems to further enhance their reliability. Several other fault detection and diagnosis methods for power electronic systems are available, e.g. observer-base methods [39] and can be implemented on microprocessors, e.g. [34]-[37], and a review of several fault diagnosis methods in power electronics is available in [40]. Fault isolation is also a way to avoid having fault conditions that influence system operation [38].
2.3 Literature Review Summary

Though all the reviewed fault detection and diagnosis approaches have their advantages, they are only useful in their certain applications or under limited conditions. The largest distinction between the proposed approach and the existing methods are the coverage of fault types. Great contributions have been done to detect and diagnose semiconductor faults, which can be seen from previous literature review. Almost all of these approaches focus on implementation in the switch OC and SC faults, such as the wavelet theory in [24] and the neural network in [27]. In fact, there are four types of power electronic devices in power electronic circuits. They are inductor, semiconductor switch, diode and capacitor, and all of these devices have been proven to fail as stated in section 2.1. It is essential to establish a comprehensive, simple, accurate, fast, and robust fault diagnosis approach to detect and diagnosis all possible faults in a power electronic system. This approach should be tailored to specific topologies for best fault diagnosis results.

This thesis proposes a generalized approach which has the capability to be applied to any power electronic system. This approach/methodology relies on existing or basic measurements for fault diagnosis, and utilizes this diagnosis to engage redundancy for system recovery. Two strategies are introduced in this thesis: combinational logic and fuzzy logic. Comparing the combinational logic with the wavelet theory in reference [23], the combinational logic does not need data transformation, and it requires fewer complex concepts and knowledge than wavelet theory, which gives a simple and intuitional way to understand and implement it. And the multiplicity of states combination of the combinational logic shows the potential to distinguish all kinds of fault types. It is also not necessary to have complex organization and reconstruction to deal with multiple faults as in neural networks [27]. Another proposed fuzzy-logic-based
methodology only uses fuzzy logic, by regulating the fuzzy rules and membership functions, different fault conditions can be recognized and the system can be recovered. Comparing all the methods used before, the proposed methodology provides the ability to diagnose all components’ fault in real time and achieves system recovery after fault occurrence.
III. PROPOSED METHODOLOGY

The generalized methodology in this thesis includes the combinational logic and the fuzzy logic, however, the implementation is not limited to these two methods. Even though the differences between these two logic-based methods are significant, three common characteristics in the stages of the input, processing and output are obvious. In the input stage, the demands of various measurements are necessary to acquire the information of the system. On the processing stage which is the main part for each method, even if the processing ways are different, the correlation is a core requirement to achieve the implementation of these methods and obtain desired results. In the output stage, regardless of the implemented method, the final outputs of different methodologies all come to a binary case. The binary output decides the condition of a system, and can be transformed to a switching signal which should be sent to the hardware circuit for the system recovery. Two parts in this chapter are shown to explore the respective characteristics for combinational and fuzzy logic.

3.1 Proposed Approach Using Combinational Logic

When a fault occurs in a power electronic circuit, it affects several if not all voltages and currents on nodes and in branches, respectively. Based on the correlation between the occurred faults and their effect on the measured signals, combinational logic can be used for fault diagnosis and to engage redundant components for intelligent recovery as described later.

3.1.1 Mathematical Description

In general, assume that $M$ measurements exit for essential voltages and/or currents. For each measurement, $P$ quantities are evaluated, as shown in Fig. 7, with an example quantity being the mean of a measured signal. This yields $Q$ measured quantities where $Q=M \times P$. Also,
assume that $N$ components are susceptible to faults with each component having $K$ fault conditions, as shown in Fig. 8. Thus, $Y$ different faults could occur in the system, where $Y=N\times K$.

\[
\begin{align*}
\text{Measurement 1} & \quad \text{Component 1} \\
\text{Measurement 2} & \quad \text{Component 2} \\
\text{Measurement M} & \quad \text{Component N}
\end{align*}
\]

Fig. 7. Measurements and related quantities

Fig. 8. Components’ fault conditions

Component faults are defined as

\[
f_{nk}; n = 1,2,3...N; k = 1,2,3...K,
\]

and their set is

\[
F = \{f_{nk} \mid n = 1,2,3...N; k = 1,2,3...K\}.
\]

Measured quantities are defined as

\[
q_{mp}; m = 1,2,3...M; p = 1,2,3...P,
\]

and their set is

\[
Q = \{q_{mp} \mid n = 1,2,3...M; p = 1,2,3...P\}.
\]

In (1)-(4), $n$ is failed component index, $k$ is component fault type index, $m$ is measured value index, and $p$ is quantity type index. Correlation is defined as

\[
c_{ij} = \begin{cases} 
1, & q_{mp} \geq q_{mp}^* \\
0, & q_{mp} < q_{mp}^* 
\end{cases},
\]
where $c_{ij}; i = mp, j = nk$, $c$ is the correlation results which represents the state of the fault, $i$ is quantity index, $j$ is fault index. The resulting correlation matrix $C$ combines a set of states for each fault occurrence as correlated with each measured quantity:

$$C = \begin{pmatrix}
c_{11} & c_{12} & \ldots & c_{1j} & \ldots & c_{1NK} \\
c_{21} & c_{22} & \ldots & c_{2j} & \ldots & c_{2NK} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
c_{i1} & c_{i2} & \ldots & c_{ij} & \ldots & c_{iNK} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
c_{MP1} & c_{MP2} & \ldots & c_{MPj} & \ldots & c_{MPNK}
\end{pmatrix}.$$  \hspace{1cm} (6)

Each measured quantity in $Q$ varies with each fault. A measured quantity is assessed online or in real-time and compared to a pre-determined threshold $q_{mp}^*$. A decision is made by comparing each of the $Q$ quantities to its respective threshold. This is reflected as a 1 or 0 decision where 1 represents a change in the measured quantity by more than the acceptable threshold, and 0 represents a change in the measured quantity within the acceptable threshold.

Two unexpected conditions may occur:

1. Ambiguous fault: There exist faults that they have the same correlation combination:

$$\{\forall f_{11}, f_{22}, \ldots \in F, q_{mp} \in Q, \exists c_{1j} = c_{2j} = \ldots\} \hspace{1cm} (7)$$

2. Redundant quantity: Some quantities have the same reaction to all faults:

$$\{\forall q_{11}, q_{22}, \ldots \in Q, f_{nk} \in F, \exists c_{i1} = c_{i2} = \ldots\} \hspace{1cm} (8)$$

Then, the decision for $Q$ inputs takes the form of a $Q$-bit number leading to $Z = 2^Q - 1$ combinations per fault.

Selecting the threshold is based on design specifications, requirements, and constraints when available. Otherwise, common-sense engineering judgment is followed. For each of the $Y$
possible faults, a combinational logic table can then be generated with an example shown in Table 1 where \( c \) is either 0 or 1. Note that each column identifies a unique fault.

It is desired to reduce the number of \( M \) measured signals, thus a method is established to eliminate some redundant inputs:

1) Some rows in the table might be the same, which implies that these quantities have the same reaction to all faults and are thus redundant. For example, assume two identical rows exist in Table 1; these quantities are thus redundant and only one is enough;

2) The difficulty of measuring related voltage or current values determines which of the redundant quantities to eliminate. It is generally easier, more accurate, and more robust to implement voltage measurements and evaluate their quantities compared to currents. Thus for example, if two identical rows are one for voltage and another for current, current measurement is eliminated;

3) Grounded measurements are generally easier than differential measurements;

4) At the end of the elimination process, \( Y \) faults should still be distinguished using the remaining measured quantities being \( Q' < Q \).

Table 1. Example of combinational logic combination for \( M \) measured quantities and \( Y \) faults

<table>
<thead>
<tr>
<th>Measurement</th>
<th>Component 1</th>
<th>Component 2</th>
<th>Component N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 2 … K</td>
<td>1 2 … K</td>
<td>… 1 2 … K</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td>M</td>
<td>1</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>⋅ ⋅ ⋅</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>c c c c c</td>
<td>c c c c c</td>
</tr>
</tbody>
</table>
3.1.2 System Recovery and Approach Flowchart

In order to achieve fault recovery, the following sequence is followed: Once a circuit or component fails, a fault is detected using combinational logic using a matrix or table as shown in Table 1, and a redundant or spare component provides support to the system to enhance its reliability. While this strategy could cause cost increase, it guarantees recovery when used with proper fault diagnosis, and cost increase can be justified through increased system lifetime, zero downtime, and postponed maintenance. When an OC fault occurs, a parallel redundant component is engaged to support the circuit; on the other hand, when a SC occurs, the faulty component is switched out of the circuit and the parallel component is engaged. This is illustrated in Fig. 9. Also, sensors installed on each parallel component can be added to maintain online current or voltage information. Measured signals are sent to the fault diagnosis and intelligent where signal pre-processing is also achieved in order to get desired quantities.

![Flowchart of System Recovery](image)

**Fig. 9.** Fault recovery using redundancy

The correlation among these quantities is carried out by using combinational logic as explained above, or fuzzy logic as explained in Section 3.2. The final decision is made and the binary signal is sent to the switch $S_n$ or the switch $B_n$ to recover from a fault. If the upside component is connected in the circuit, the initial condition of $S_n$ is set to closed and the initial condition of $B$ is set to open to make downside component as standby or redundant component.
Once the open circuit fault of upside component is detected, the output binary signal from the fault diagnosis and intelligent recovery system gives a logic 1 to the switch $B$ to engage the downside redundant component into the circuit. If the short circuit fault occurs instead, an action of sending a logic 0 signal to switch $S_n$ should be added to the recovery steps of the open circuit fault recovery. Fig. 10 summarizes the proposed approach using a flowchart. Note that $S_n$ is component $n$’s series switch, and $B_n$ is its parallel redundant component switch as shown in Fig. 9. $I$-bit numbers are the combinations used as columns in Table 1. The fault diagnosis and intelligent recovery system will keep monitoring the circuit until a user ends operation.

Fig. 10. Flowchart of the proposed approach for combinational logic
3.2 Proposed Approach Using Fuzzy Logic

3.2.1 Difference between Combinational and Fuzzy Logic Approaches

Fuzzy logic presents a more intelligent control method compared to the combinational logic by utilizing intermediate values between 0 and 1 when deciding on the quality of a measured quantity. While the membership functions (MFs) for the inputs and outputs need to be defined, the problem of identifying a suitable threshold is eliminated.

3.2.2 Fuzzy Logic-Based Approach and Its Flowchart

The proposed fuzzy-logic-based fault diagnosis method is based on two important concepts: 1) Fault occurrence in any component of a power electronic system will impact various voltages and currents across the system; and 2) the impact on these voltages and currents will vary in severity depending on the fault mode and location. For the proposed approach, the voltages and currents across or in each circuit component are measured for each possible SC or OC fault in major components. A fuzzy controller is then designed for each component and the number of measured signals \( M \) is reduced as needed with voltages taking priority over currents as they are less expensive and more practical to measure. Each of the \( M \) signals would then have \( P \) quantities calculated, e.g. mean or RMS value. These quantities are then fed to fuzzy controllers that are each dedicated for a specific fault mode. Fuzzy controllers utilize the \( P \) quantities per fault mode to output a value that is compared among all controllers. The correlation block shown in Fig. 11 selects the maximum value from all fuzzy controller outputs and declares a fault diagnosed. More details of correlation system are introduced in subsection Redundant components are engaged to replace faulty ones as needed when a fault is diagnosed.

In implementation, each component has its unique fuzzy rules. A critical problem here is that when one component fails, all the system will be influenced and all fuzzy controllers’
outputs are influenced. Note that fuzzy controllers’ outputs range from 0 to 1 such that the higher the number for a specific output related to a specific fault mode, the more probability the related component has failed in that fault mode. To engage redundancy shown in Fig. 9, the failed component is removed from the circuit by switching off $S_n$ and switching on $B$. Due to some signal variations during nominal healthy operation, a safe margin (SM) is set in the correlation system of Fig. 11 where only fuzzy controller outputs that exceed SM are sent to the correlation block. The flowchart of the proposed method implementation is shown in Fig. 12.

![Flowchart of the proposed method implementation](image)

**Fig. 11. Fuzzy logic based methodology correlation system**

Similar to combinational logic, the measured quantities are defined as

$$q_{mp}; m = 1,2,3...M; p = 1,2,3...P \ (9),$$

and their set is

$$Q = \{q_{mp} | m = 1,2,3...M; p = 1,2,3...P\} \ (10).$$

The output values of fuzzy controllers are defined as

$$OV_i; i = 1,2,3...M \ (11),$$

and the over SM output values are defined as
where \( j \) is the index of over SM output values. The final output of the correlation system if component \( k \) failed is defined as

\[
FO_k = \max\{OSMC_j\} (13)
\]

Fig. 12: Flowchart of the proposed approach for fuzzy logic

3.2.3 Illustrative Example

As mentioned in section 2.2.3, there are three stages in each fuzzy controller, the input MFs, fuzzy rules, and output MFs. The construction of a fuzzy controller should be suitable to the application requested by users. An example is shown in Fig. 13. In this case, measured
quantities are RMS and mean of a voltage signal, and MF blocks show a horizontal axis being the real values of these quantities while the vertical axis is the MF proportion between 0 and 1. Each of the quantity has three MFs: low (L), median (M) and high (H). In general, the higher the value on the vertical axis, the more contribution the MF makes to the output. There are three MFs for the RMS and mean inputs. The output MF is similar with the main difference being the names of the MFs: Low (L), normal (N) and high (H). The fuzzy rule uses “if…then” logic to assign the combination of input MFs to an output MF where input MF contributions determine the corresponding output MF’s contribution to the final decision.

RMS and the mean values of the voltage across a component are calculated in real time and fed to the input MFs. Assume that the mapped values in vertical axis with respect to the triangular MFs are points “1” in L, “2” in M, “3” in H in the MF of “RMS”, and “4” in M, “5” in H, “6” in L in the MF of “Mean”. These points are all highlighted with yellow dots in Fig. 13. It should be noticed that the value of 2, 3 and 6 point is zero, which means they do not have contributions in this case. After obtaining these values, the specified fuzzy rules are run to assign output MFs. Here, maximum is used to extract output MFs. The vertical value of each output MF is determined by the maximum value of corresponding input memberships. For instance, the first and third rules are related to output L MF, and the values of the corresponding input memberships in these two rules are compared to get the maximum value which is point “1”. The same procedure is followed, “4” is determined for N, and “5” is determined for H. Mapping the values of the three points from vertical axis to horizontal axis, two points are obtained for each output MF. However, because of overlapping, there are some interactions between these MFs and some of the points are useless. Only the yellow dotted points in output MF which form the envelope are taken into consideration. If the horizontal value is represented as $x$, corresponding
vertical value is represented as $u(x)$ and the number of points is $n$, the final decision/output of the fuzzy controller is determined by

$$g = \frac{\sum_{i=1}^{n} x_i * u(x_i)}{\sum_{i=1}^{n} u(x_i)} \quad (14)$$

The output value is between 0 and 1 and significantly relies on the setting of fuzzy rules, so the fuzzy rules should be set properly in a training phase to obtain the desired output. An example of a different fuzzy rule setting is shown in Fig. 14 which gives distinct points in the output MF.

Fig. 13. Example of fuzzy controller

Fig. 14. Example of fuzzy controller with different fuzzy rules
In an online implementation, each component has its own fuzzy-logic-based diagnosis to monitor its condition. All output values are compared through a maximum function which indicates the failure of the component related to the maximum output. Because of some variations in measured signals and quantities, even when all components are healthy, false diagnosis may occur. Thus, 0.6 is selected as the SM to eliminate such nuisance faults. The closer to the values of faulty condition the input values are, the higher level of output MF will be mapped to. Fig. 15 shows the procedure which is inside the correlation system block of Fig. 11. Four outputs of individual components’ fuzzy diagnosis are generated and sent to compare with SM. Assuming that these outputs take the values shown on the left of Fig. 15, three of four values exceed SM and are passed to the maximum function to flag a fault in component 4 and generate a recovery signal for that component.

![Comparison: > SM=0.6?](image)

**Fig. 15. Example correlation system**
IV. MICRO-INVERTER PLATFORM: NOMINAL OPERATION & FAULT INJECTION

4.1 Basic Topology

4.1.1 Platform Description

A photovoltaic (PV) micro-inverter is used as an example platform to test the proposed approach for combinational and fuzz logic. The platform is shown in Fig. 16 at a high level. Three stages should be considered in a renewable energy subsystem [42]:

1. The initial stage is the renewable energy source being PV, wind turbine, fuel cell etc. It is common to have the output of the renewable energy source in the form of DC voltage.

2. A problem of renewable energy output is that the voltage is needs to be synchronized with grid voltage or regulated for supporting a load. A micro-inverter is used in PV systems to provide both DC/DC regulation and DC/AC inversion.

3. With a well-regulated output voltage, power is feed into the grid or an AC load.

Closed-loop control provides voltage regulation of the DC link voltage between the DC/DC and DC/AC converters, and fault diagnosis and recovery control are also built on the same control platform. The choice of a micro-inverter as an illustrative platform is due to many reasons including its reasonable complexity, need for closed-loop control for voltage regulation, rising interest in its adoption as a PV system integration solution, and need to improve its reliability given that its warranty is usually lasts less than half of a PV panel’s warranty so improving its reliability is essential to avoid replacing the whole micro-inverter twice or more during a panel’s lifetime. Details about the implementation of the proposed approach for this platform are summarized in Fig. 17, but can be applied to other power electronic systems.
In order to apply fault diagnosis and recovery, the platform should experience fault conditions, so the second main step is fault injection and evaluation. Three central sub-steps are quite significant:

1. Modeling and injecting faults into the platform: Multiple fault conditions could occur, and OC and SC faults are considered and injected into each component to analyze their effects.

2. Fault detection and diagnosis relies on the measured signal (voltage & current), so the effect of each fault condition to each measured signal should be monitored. Moreover, certain quantities are calculated from each measurement where fault effects can be observed.
3. Following observation, analysis is applied to find out the relationship between quantities and faults; this step can give a better overview among several fault conditions and is a preparation for later intelligent fault detection and diagnosis system.

The proposed fault diagnosis approach is designed to express the influence from fault conditions to quantities; also, these quantities are utilized as inputs of the logic-based approach to generate fault condition flags and recovery signals. The platform is simulated to prove its feasibility and validity, after that hardware setup is built to validate the simulation and implement proposed approach. Last but not least, evaluating system superior performance with fault recovery using the proposed approach is achieved through simulation and hardware testing of the proposed approach.

The two-stage micro-inverter is shown in Fig. 18 [43], where the first stage is a buck-or-boost DC/DC converter. Two modes are available in this topology where it can be modified to achieve either:

1. Boost converter:
   1) Remove MOSFET $S_1$ and short its drain-to-source connection.
   2) Remove diode $D_1$ and keep its position open.

2. Buck converter:
   1) Remove MOSFET $S_2$ and keep its position open.
   2) Remove diode $D_2$ and short its anode-to-cathode connection.

The second stage is an H-bridge DC/AC inverter. The output side of this micro-inverter is connected to a resistor acting as an AC load through an LCL filter. The desired output voltage of the micro-inverter is 115Vrms at a fundamental frequency of 60Hz. In the platform used here, the DC/DC converter is setup as a boost converter to boost the voltage source up to 200V DC
across $C_{out}$ to meet the final output requirements. Table 2 shows the micro-inverter specifications. Note that harmonics and power quality were not part of the design specifications but can be integrating into the component selection process. Also, other micro-inverter topologies or power electronic converters can be used instead of that shown in Fig. 18, and the choice of this topology is just for illustrative purposes.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input voltage</td>
<td>20~40V</td>
</tr>
<tr>
<td>DC bus voltage</td>
<td>200V</td>
</tr>
<tr>
<td>Output voltage RMS</td>
<td>115V</td>
</tr>
<tr>
<td>Output voltage fundamental frequency</td>
<td>60Hz</td>
</tr>
<tr>
<td>Full power level</td>
<td>200W</td>
</tr>
</tbody>
</table>

Even though only boost mode is employed on the DC/DC side, components were chosen to work in both boost and buck modes for other research purposes. Table 3 shows the chosen inductor and capacitor values for the DC/DC converter and DC/AC LCL filter.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Component</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC/DC</td>
<td>Inductor $(L)$</td>
<td>470 $\mu$H</td>
</tr>
<tr>
<td></td>
<td>Capacitor $(C_{out})$</td>
<td>220 $\mu$F</td>
</tr>
<tr>
<td>DC/AC (LCL Filter)</td>
<td>Inductor $(L_{f1}, L_{f2})$</td>
<td>155 $\mu$H</td>
</tr>
<tr>
<td></td>
<td>Capacitor $(C_f)$</td>
<td>1 $\mu$F</td>
</tr>
</tbody>
</table>

Fig. 18. Micro-inverter platform used to test the proposed approach
4.1.2 Fault Injection

Fig. 18 is implemented in Simulink but with each component modified for controlled SC and OC fault injection. This is reflected in Fig. 19, where the highlighted components are internally built as shown in Fig. 20. \( S_{SC} \) is an ideal switch used to cause a SC fault across a component, while \( S_{OC} \) is an ideal switch used to make an OC fault in series with the component.

![Fig. 19. Fault locations highlighted on the power side of the platform](image)

![Fig. 20. Component fault injection strategy](image)

4.2 Simulation Model

MATLAB Simulink is the tool used to build the simulation model of Fig. 18 and fault injection of Fig. 19. To capture major hardware transients and effects, non-ideal components are used in the simulation where parasitic resistances were measured using and LCR meter and included in the simulation model. These resistances include wire resistance and component series resistance. Table 4 gives the non-ideal parameters and values. The power supply used in place of the DC source has capacitance \( C_{in} \) shown in Fig. 18, which was also included in the simulation. More details about simulation model can be found in appendix B.
Table 4. Non-ideal parameters and values

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Wire Resistance (Ω)</td>
<td>0.3</td>
</tr>
<tr>
<td>Inductor L Series Resistance (Ω)</td>
<td>0.0024</td>
</tr>
<tr>
<td>MOSFET (S_2, S_A, S_A', S_B, S_B') Series Resistance (Ω)</td>
<td>0.042</td>
</tr>
<tr>
<td>Diode (D_2) Series Resistance (Ω)</td>
<td>0.001</td>
</tr>
<tr>
<td>Capacitor (C_{in}, C_{out}) Series Resistance (Ω)</td>
<td>0.3</td>
</tr>
</tbody>
</table>

### 4.3 Experimental Setup

Fig. 21 shows the experimental setup. Micro-inverter has exactly the same topology as Fig. 18. Many MOSFETs standing along the board edges are used as switches to inject OC and SC, and each component unit has a parallel spare component to provide redundancy that is actively engaged using the fault diagnosis algorithm. High power tube resistive load is used as the AC load. Low power supplies are used to provide supply to the FPGA board, sbRIO-9612, hosting the real-time control and fault diagnosis platform, current sensors, and MOSFET gate drivers. LabView is used for all the control and interfacing between software and hardware. The oscilloscope in Fig. 21 is used to capture the voltage and current waveforms. Fig. 22 is the 200W high power supply, and it is controlled to have same range as a PV panel and is put to use as input to the platform instead of a real PV panel.

As shown in Fig. 23, the DC/DC converter and DC/AC inverter are designed separately. Each central component has a redundant component in parallel with the original one. The original component and its redundant one compose a pair. Six pairs of components can be found on the board, which is the same as Fig. 19. Voltage dividers are used to measure voltages in the circuit and current sensors are connected in series with components where needed. The voltage and current measured form the voltage dividers or the current sensors are scaled to -10–10V to fit the range of the analog FPGA ports. These scaled values are sent to analog output ports on the
micro-inverter board. Afterwards, the signal connectors transmit the analog signals to the FPGA analog input ports. The digital input ports on the micro-inverter are connected to the FPGA digital output ports to obtain switching signals for both PWM switching on the DC/DC and DC/AC sides, and switching in and out components with emulated failures. Op-amps are used to fit digital signals to the switching required voltage levels. For example, if a MOSFET requires up to 12V to be switched on, but the rating of the digital signal from the FPGA is only 3.3 V with limited current, the op-amps will adjust the 3.3V digital signal to 12V to feed into the MOSFET gate driver and switch on the MOSFET. Fig. 24 shows the FPGA board.

![Fig. 21. Experimental platform](image1)

![Fig. 22 High power supply](image2)
Fig. 23. Micro-inverter power board

Fig. 24. FPGA board
4.4 Model Verification

4.4.1 Nominal/Healthy Operation

Fig. 25 shows the experimental interfacing panel which has the same typology as the platform. Experimental results are obtained from the control of the interfacing system. Nominal transient and steady-state operation are validated as shown in Figures 26 and 27 for the micro-inverter DC voltage ($V_{dc}$) and AC voltage ($V_{ac}$). For transient validation, it shows the rise of $V_{dc}$ from 50V to 150V; for steady-state validation, the system runs almost at full power of 200W with 200V $V_{dc}$. The measurement of channel 1 is at 100V/div, 1x scaling magnification; the measurement of the channel 2 is also 100V/div because of 20x scaling. From the simulation and experimental result comparison, it can be seen that both transient and steady-state have similar waveforms. In Figure 26, the transients appear a spike before it enters the steady-state of 150V, even though the settling time has 50% error, the settling time is relatively small. In Figure 27, the steady-state DC bus voltages have close values with 8% error approximately, and the steady-state frequency of simulation and experimental results are almost the same. The Experimental setup induces some harmonics which does not appear in simulation result, but the peak value and frequency of $V_{ac}$ in experimental results are similar in simulation results.
4.4.2 Operation under faults

To validate the modelling accuracy under faults, OC fault of the MOSFET $S_2$, $S_{B'}$ and the diode $D_2$ are shown here as examples. To emulate fault conditions, components are actively removed from the converter circuit by either disconnecting an auxiliary series switch to mimic an open circuit, or a parallel auxiliary switch to mimic a short circuit. Figures 28, 29 and 30 show the effect of fault conditions, respectively, on $V_{dc}$ and $V_{ac}$. All experimental and simulation results show similar steady-state values and transient settling times. To be more specific, in Figure 28, under faulty condition, the DC voltage drops to the input voltage value with the dropping of AC voltage peak value accordingly; in Figure 29, the peak value of AC voltage in experimental result is slightly different from expected but it shows the similarity with the disappearance of upside AC waveform and the double frequency after fault happens; in Figure 30, the DC voltage and AC voltage in simulation and experimental results all drop to zero under
faulty condition. With the simulation model validated for the micro-inverter, the simulation can then be used to develop the proposed approach in a flexible environment for later implementation in hardware experiments.

**Fig. 28.** Micro-inverter DC voltage and AC voltage under MOSFET $S_2$ OC condition

**Fig. 29.** Micro-inverter DC voltage and AC voltage under Diode $S_B$ OC condition

**Fig. 30.** Micro-inverter DC voltage and AC voltage under $D_2$ OC condition
V. IMPLEMENTATION OF THE PROPOSED APPROACH

5.1 Combinational Logic Implementation

5.1.1 Logic Gates and Tables

Calculated quantities from measured signals are fed into comparison blocks to compare with their thresholds. Based on whether the quantity is smaller or larger than its threshold value, logical 0 or 1 is generated. If a quantity is larger than its threshold value, the generated logic 1 value should be maintained even after the quantity returns to value below the threshold. Therefore, the output of the combinational logic fault diagnosis design is followed by the structure in Fig. 31 utilizing the MinMax (Maximum) and unit delay blocks in Simulink. When the logic signal is 0, the delay signal is also 0, which gives 0 at the maintained signal side; if the logic signal suddenly becomes 1, the delay signal takes previous 0 value, the maintained signal becomes 1 and stays at 1 due to the maximum function, even if the logic signal returns to 0. This helps maintain the fault flag that a fault occurred.

![Fig. 31. Design for signal maintenance](image)

Under nominal conditions, the logic signal of each quantity is zero since all quantities are below their fault thresholds; under faulty conditions, different combinations of logic 1s and 0s indicate different faults. The unique representation ensures the 100% fault diagnosis and system recovery given that the nominal operating point of the system does not change and sensor feedback is reliable. These limitations have been addressed in a recent publication [44]. Suppose that the RMS and mean values of $V_{dc}$ and $I_{dc}$ are the quantities used in combinational logic fault diagnosis for certain faults. Under nominal conditions, the logic values follow the vector \{0, 0, 0,
the elements of the vector are in the order of the logic value for comparing $V_{dc}$ RMS, $V_{dc}$ mean, $I_{dc}$ RMS, and $I_{dc}$ mean with their thresholds. When an OC fault of MOSFET $S_2$ happens, the vector becomes $\{1, 1, 1, 1\}$, which means all quantities are over the threshold; when an SC fault of an inductor $L$ happens, the vector becomes $\{0, 1, 0, 1\}$ which means that $V_{dc}$ mean and $I_{dc}$ mean are over their thresholds. The way to obtain the final output logic is thus by inverting 0 to “not 1” using “not gates”, e.g. $\{\bar{1},1,\bar{1},1\} = \{0, 1, 0, 1\}$. The procedure is shown in Fig. 32. The signal of the OC fault of MOSFET is sent to the upside “AND” block, and the output “1” of the upside “AND” indicates the failure of MOSFET for OC fault. The downside output “1” indicates the failure of Inductor for SC fault.

![Diagram](image)

Fig. 32. An example design for a unique combination using combinational logic fault diagnosis

In order to study the correlation between each fault and the measured voltages and currents, nominal operation is first simulated then faults are injected one at a time with no sequential faults so that correlation is studied independent of fault sequences. Three main quantities are checked for each signal or measurement as shown in Fig. 16—the mean or average
value, RMS, and THD. Table 5 shows the effect of each fault on the measured voltages and currents. Note that these tables originally had 12 rows corresponding to one voltage and one current measurement in each of the six faulty components, but the number of measurements was reduced due to redundant combinations as described in Section III. All values are compared with a nominal condition, where 1 means over 90% difference and 0 means less than 90%. Note that if a fault cannot be distinguished from these quantities, other quantities such as the mean value sign can also be utilized. Faults in $S_A$ and $S_A'$ behave similarly to those in $S_B$ and $S_B'$ and are thus not shown to simplify the analysis.

Table 5. Correlation between faults and change of measured signals from nominal operation (O is for OC and S is for SC)

<table>
<thead>
<tr>
<th></th>
<th>$L$</th>
<th>$D_2$</th>
<th>$S_2$</th>
<th>$C_{OUT}$</th>
<th>$S_B$</th>
<th>$S_B'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_L$</td>
<td>RMS</td>
<td>1 0</td>
<td>0 1</td>
<td>1 1</td>
<td>1 1</td>
<td>0 1</td>
</tr>
<tr>
<td></td>
<td>MEAN</td>
<td>1 0</td>
<td>0 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
</tr>
<tr>
<td></td>
<td>THD</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 0</td>
</tr>
<tr>
<td>$V_{S2}$</td>
<td>RMS</td>
<td>1 0</td>
<td>0 1</td>
<td>0 1</td>
<td>1 0</td>
<td>0 0</td>
</tr>
<tr>
<td></td>
<td>MEAN</td>
<td>1 0</td>
<td>0 0</td>
<td>0 1</td>
<td>0 0</td>
<td>0 0</td>
</tr>
<tr>
<td></td>
<td>THD</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 0</td>
</tr>
<tr>
<td>$V_C$</td>
<td>RMS</td>
<td>1 0</td>
<td>0 1</td>
<td>0 1</td>
<td>1 1</td>
<td>0 1</td>
</tr>
<tr>
<td></td>
<td>MEAN</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
</tr>
<tr>
<td></td>
<td>THD</td>
<td>0 0</td>
<td>0 1</td>
<td>0 1</td>
<td>1 0</td>
<td>1 1</td>
</tr>
<tr>
<td>$V_B$</td>
<td>RMS</td>
<td>1 0</td>
<td>0 1</td>
<td>0 1</td>
<td>1 1</td>
<td>1 1</td>
</tr>
<tr>
<td></td>
<td>MEAN</td>
<td>1 0</td>
<td>1 1</td>
<td>1 1</td>
<td>0 1</td>
<td>0 1</td>
</tr>
<tr>
<td></td>
<td>THD</td>
<td>0 0</td>
<td>0 1</td>
<td>0 0</td>
<td>1 1</td>
<td>1 1</td>
</tr>
<tr>
<td>$V_B'$</td>
<td>RMS</td>
<td>1 0</td>
<td>0 1</td>
<td>0 1</td>
<td>1 1</td>
<td>1 1</td>
</tr>
<tr>
<td></td>
<td>MEAN</td>
<td>1 0</td>
<td>1 1</td>
<td>1 1</td>
<td>0 1</td>
<td>0 0</td>
</tr>
<tr>
<td></td>
<td>THD</td>
<td>0 0</td>
<td>0 1</td>
<td>0 0</td>
<td>1 1</td>
<td>1 0</td>
</tr>
</tbody>
</table>

In this case, the five remaining measurements after elimination are: inductor current ($I_L$), MOSFET $S_2$ voltage ($V_{S2}$), capacitor $C_{out}$ voltage ($V_C$), switch $B$ voltage ($V_B$), and switch $B'$ voltage ($V_{B'}$). Note that $V_C$ is $C_{out}$’s series sensing resistor voltage which reflects $I_{Cout}$. For each measurement, three quantities are taken and a 15-bit number is generated (columns of Table 5
represent unique 15-bit numbers). By relating to Section III, $M = 5$ measurements, $P=3$ quantities, $Q=15$, $N=6$ components, $K=2$ fault modes (OC and SC), $Y=12$, and $Z=215-1$ of which 12 combinations are used (columns in Table 1) for fault diagnosis.

5.1.2 Example Results

An example application of the fault injection and intelligent recovery by controlling redundancy is applied to $C_{out}$. Results shown in Table 5 are utilized to remove $C_{out}$ and engage a spare capacitor. $V_{dc}$ is maintained as desired as shown in Fig. 33 and the DC/DC $V_{dc}$ recovers. It is important to note that the fault diagnosis speed and system recovery are critical. In simulations, a fault is injected at $t_0$, detected at $t_1$, and the system recovers at $t_2$ as demonstrated in Fig. 33. Another example is shown in Fig. 34 that the system survives from OC fault of MOSFET ($S_2$).

Fig. 33. Simulation: System survival by detecting $C_{out}$ SC and engaging a spare capacitor

Fig. 34. Simulation: System survival by detecting the $S_2$ OC and engaging a spare MOSFET
Table 6 shows these times achieved for all possible faults. $t_2$ is recorded when all the measured values recover to 20% of their nominal values. This table clearly shows that most faults can be diagnosed quickly and once that is achieved, system recovery is also fast but is determined by the system dynamics. Some fault diagnosis, e.g. $D_2$ OC and $S_2$ SC, are slower than others and it will be shown that using fuzzy logic improves this response at the cost of added complexity.

Table 6. Combinational Logic fault diagnosis and system recovery time

<table>
<thead>
<tr>
<th>Fault</th>
<th>Fault Occurrence Time $t_0$ (s)</th>
<th>Fault Diagnosis Time $t_1$-$t_0$ (s)</th>
<th>Fault Recovery Time $t_2$-$t_1$ (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$ OC</td>
<td>0.6</td>
<td>0.4875</td>
<td>0.0424</td>
</tr>
<tr>
<td>$D_2$ OC</td>
<td>0.6</td>
<td>1.1878</td>
<td>0.0639</td>
</tr>
<tr>
<td>$S_2$ OC</td>
<td>0.6</td>
<td>0.4897</td>
<td>0.0441</td>
</tr>
<tr>
<td>$C_{out}$ OC</td>
<td>0.6</td>
<td>0.0255</td>
<td>0.0643</td>
</tr>
<tr>
<td>$S_B$ OC</td>
<td>0.6</td>
<td>0.0472</td>
<td>0.0161</td>
</tr>
<tr>
<td>$S_B'$ OC</td>
<td>0.6</td>
<td>0.0394</td>
<td>0.0159</td>
</tr>
<tr>
<td>$L$ SC</td>
<td>0.6</td>
<td>0.0660</td>
<td>0.0056</td>
</tr>
<tr>
<td>$D_2$ SC</td>
<td>0.6</td>
<td>0.0167</td>
<td>0.0636</td>
</tr>
<tr>
<td>$S_2$ SC</td>
<td>0.6</td>
<td>0.9835</td>
<td>0.2377</td>
</tr>
<tr>
<td>$C_{out}$ SC</td>
<td>0.6</td>
<td>0.0195</td>
<td>0.0631</td>
</tr>
<tr>
<td>$S_B$ SC</td>
<td>0.6</td>
<td>0.0165</td>
<td>0.0618</td>
</tr>
<tr>
<td>$S_B'$ SC</td>
<td>0.6</td>
<td>0.0165</td>
<td>0.0629</td>
</tr>
</tbody>
</table>

Example applications of experimental setup are performed. OC fault is applied to MOSFETs $S_B'$ and $S_2$ and are shown in Figures 36 and 37 and SC fault is applied to the inductor $L$ at 90W output power. The top blue curve is $V_{dc}$ with 100V/div. The second red curve is $I_{dc}$ with 10A/div, it shows the system runs at DCM mode. The green curve indicates the fault diagnosis and initiation of recovery. The bottom purple curve is $V_{ac}$. Even though the $V_{ac}$ shows switching ripples under high power due to not large enough capacitance, the effect of recovery is obvious. Because of the inrush current from fault and switching, the power supply goes into current limit mode (CCM) and results for the inductor SC are not shown for 90W. Results for the
inductor SC recovery at low power (5 W) are shown in Figure 35 to demonstrate fault diagnosis and recovery when enough supply current is available. Simulation results for recovery of each of these faults are shown in Figures 38 to 40. It is clear from these figures that simulations and experimental results match well, and that fault recovery is almost seamless within around 100 ms with minimum transient. Implementation of the diagnosis and recovery algorithms in LabView FPGA is shown in the appendix C.

![Fig. 35. Experimental: System survival by detecting the L SC and engaging another Inductor in place](image1)

![Fig. 36. Experimental: System survival by detecting the S2 OC and engaging another MOSFET in place](image2)
Fig. 37. Experimental: System survival by detecting the $S_B$ OC and engaging another MOSFET in place

Fig. 38. Simulation validation of $LSC$ experimental results

Fig. 39. Simulation validation of $S_2$ OC experimental results
5.2 Fuzzy Logic Implementation

5.2.1 Implementation in Simulink

The fuzzy toolbox in MATLAB is used to modify fuzzy logic MFs as shown in Fig. 41. Each MF is assigned a unique name and loaded to workspace, then, a fuzzy controller block in Simulink can call the MF by placing the MF name the block as shown Fig. 42.

![Fuzzy toolbox interface](image1)

![Function Block Parameters: Fuzzy Logic Controller](image2)

Fig. 41. Fuzzy toolbox interface

Fig. 42. Fuzzy logic controller (fault diagnosis) in Simulink

The SM comparison is achieved by enabling an “Enable” block which is put in the data transmission subsystem. Only signals exceeding SM will pass through the transmission (Tran) subsystem, otherwise, zero is generated at the output port of transmission subsystem. The
transmitted signal is sent to compare and obtain the largest one. Each fuzzy controller has a comparison system, and the output of the corresponding comparison demonstrates the failure or survival of a component. If no fault occurs, the output ports of the fuzzy controllers all generate the values that are below SM, which gives zeros at the comparison outputs requiring no recovery signals under healthy condition. If a fault occurs, the corresponding comparison system generates a logic “1” signal due to the appearance of the fuzzy controller’s largest value in the corresponding comparison. To maintain the switching signal, the structure in Fig. 31 is utilized. The detailed process is shown in Fig. 42 for a three-fault example.

Fig. 43. Fuzzy-logic-based fault diagnosis implementation in simulations

An example of simulation is shown here to demonstrate improvements achieved using fuzzy logic by addressing the SC and OC $C_{out}$ faults. One improvement is the reduction of measured quantities for different faults. For example, in order to detect a SC or OC fault in $C_{out}$, only $V_C$’s RMS, mean are used as inputs into the fuzzy controller while its output is $C_{out}$’s fault diagnosis decision. The other 12 quantities used in the combinational logic case are not used for this specific fault diagnosis even though they might still be needed for other fault diagnosis. Another expected improvement is faster fault diagnosis time due to the utilization of MFs compared to preset thresholds. Figs. 44 and 45 show the output and input MFs where three states are used: low (L), medium (M) and high (H). Table 7 shows the fuzzy rules of $C_{out}$’ SC and OC
conditions based on the simulation model performance. Details about the implementation of fuzzy-logic-based fault diagnosis and recovery in LabView FPGA are shown in the Appendix.

Fig. 44. Output Membership function

(a) RMS of $C_{out}$ voltage $V_C$ membership function

(b) Mean of $C_{out}$ voltage $V_C$ membership function

Fig. 45. Input membership function

Table 7. Fuzzy rules

<table>
<thead>
<tr>
<th>MF</th>
<th>RMS</th>
<th>Mean</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VL</td>
<td>VL</td>
<td>F</td>
</tr>
<tr>
<td>2</td>
<td>VL</td>
<td>L</td>
<td>L2</td>
</tr>
<tr>
<td>3</td>
<td>VL</td>
<td>M</td>
<td>L2</td>
</tr>
<tr>
<td>4</td>
<td>VL</td>
<td>H</td>
<td>L1</td>
</tr>
<tr>
<td>5</td>
<td>L</td>
<td>VL</td>
<td>L2</td>
</tr>
<tr>
<td>6</td>
<td>L</td>
<td>L</td>
<td>L1</td>
</tr>
<tr>
<td>7</td>
<td>L</td>
<td>M</td>
<td>L1</td>
</tr>
<tr>
<td>8</td>
<td>L</td>
<td>H</td>
<td>L1</td>
</tr>
</tbody>
</table>
5.2.2 Example Results

To compare with the results shown in Fig. 33 and 34, $V_{dc}$ and $V_{ac}$ with fuzzy logic fault diagnosis are shown in Fig. 46 and 47. It is clear that $t_1$ decreases significantly compared to Fig. 33 and this is mainly attributed to eliminating the wait time to achieve 90% change in the measured quantities. Table 8 summarizes times needed to detect and recover for fuzzy logic implementation. Comparing with combinational logic fault diagnosis and system recovery time, it’s clear that the system can have a much better reaction for different types of fault. Note that some recovery time is determined by the system dynamics after engaging a redundant or spare when the fault is detected, and some of them are thus not expected to change significantly.
Fig. 46. Simulation: System survival by detecting the $C_{out}$ SC and engaging another capacitor in place with fuzzy logic control

Fig. 47. Simulation: System survival by detecting the $S_2$ OC and engaging another capacitor in place with fuzzy logic control

Table 8. Fuzzy Logic fault diagnosis and system recovery time

<table>
<thead>
<tr>
<th>Method</th>
<th>Fault Occurrence Time $t_0$ (s)</th>
<th>Fault Diagnosis Time $t_1-t_0$ (s)</th>
<th>Fault Recovery Time $t_2-t_1$ (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$ OC</td>
<td>0.6</td>
<td>0.0080</td>
<td>0.0450</td>
</tr>
<tr>
<td>$D_2$ OC</td>
<td>0.6</td>
<td>0.0100</td>
<td>0.0300</td>
</tr>
<tr>
<td>$S_2$ OC</td>
<td>0.6</td>
<td>0.0088</td>
<td>0.0312</td>
</tr>
<tr>
<td>$C_{out}$ OC</td>
<td>0.6</td>
<td>0.0080</td>
<td>0.0643</td>
</tr>
<tr>
<td>$S_B$ OC</td>
<td>0.6</td>
<td>0.0130</td>
<td>0.0070</td>
</tr>
<tr>
<td>$S_B'$ OC</td>
<td>0.6</td>
<td>0.0130</td>
<td>0.0070</td>
</tr>
<tr>
<td>$L$ SC</td>
<td>0.6</td>
<td>0.0035</td>
<td>0.0065</td>
</tr>
<tr>
<td>$D_2$ SC</td>
<td>0.6</td>
<td>0.0083</td>
<td>0.0717</td>
</tr>
<tr>
<td>$S_2$ SC</td>
<td>0.6</td>
<td>0.0067</td>
<td>0.0133</td>
</tr>
<tr>
<td>$C_{out}$ SC</td>
<td>0.6</td>
<td>0.0066</td>
<td>0.0631</td>
</tr>
<tr>
<td>$S_B$ SC</td>
<td>0.6</td>
<td>0.0165</td>
<td>0.0618</td>
</tr>
<tr>
<td>$S_B'$ SC</td>
<td>0.6</td>
<td>0.0165</td>
<td>0.0629</td>
</tr>
</tbody>
</table>

Two aspects should be highlighted based on the simulation results. First, only one component fault is injected each time even though it is more realistic to have cascaded faults and failures. At this point, injection of single faults is used to demonstrate the proposed method but future work can address cascaded faults. Second, the proposed method can confuse more than one fault mode as it diagnoses other faults even when only one fault occurs. This is not a very likely scenario as every fault leaves its specific fingerprint in the system, but these fingerprints
can have minor differences. In the work presented here $S_2$ SC and $D_2$ SC are both diagnosed when one occurs, and in that case parallel components for each will be engaged for additional safety. All other faults were diagnosed without confusion.

Example applications of experimental setup are performed with fuzzy-logic-based approach. OC fault is applied to MOSFETs $S_{B'}$ and $S_2$ and are shown in Figures 48 and 49. All the divisions and curve orders are the same as combinational logic case. The fuzzy controllers utilized in the experimental setup only apply to the two example fault conditions due to the partial building of MFs. The CCM problem caused by inrush current still exists in fuzzy logic application, only under low power (5W), the diagnosis and recovery system can diagnose $L$ SC fault and recover the system. The small inductor current under low power is difficult to be utilized by a fuzzy controller since the input value range is blurry. Simulation results for recovery of each of OC faults are shown in Figures 50 to 51. It is clear from these figures that even the simulation results performs little better than experimental results, they match well with similar recovery transient time and peak value, and it shows a better performance than combinational logic case with less fault diagnosis time. Implementation of the diagnosis and recovery algorithms in LabView FPGA is shown in the appendix C.
Fig. 48. Experiment: System survival by detecting the $S_2$ OC and engaging another capacitor in place with fuzzy logic control

![Graph showing experimental results](image)

Fig. 49. Experiment: System survival by detecting the $S_B$ OC and engaging another capacitor in place with fuzzy logic control

![Graph showing experimental results](image)

Fig. 50. Simulation validation of $S_2$ OC experimental results with fuzzy logic control

![Graph showing simulation results](image)

Fig. 51. Simulation validation of $S_B$ OC experimental results with fuzzy logic control

![Graph showing simulation results](image)
VI. IMPLEMENTATION CHALLENGES

The implementation of the methodology in this thesis can be divided into three central parts: 1) Platform simulation with all fault diagnosis and system recovery, 2) Hardware design and testing, and 3) LabView real-time control with FPGA interfacing. A number of expected and unexpected problems occur when each part was established.

In the simulation part, MATLAB Simulink was used as the simulation tool due to the flexibility for control building. The use of specific blocks in Simulink were necessary to meet the desired functions. One of the challenges face was maintaining a fault flag after a fault is detected, and this led to the development of Fig. 21. Removing this subsystem can confuse the fault diagnosis system. Adding non-idealities to achieve meaningful simulations that better reflect experiments was an iterative process where non-idealities were measured and then simulations were updated with more realistic component models. This is an essential step in model-based control and diagnosis development.

In the hardware design part, having many gate drivers for fault emulation switches and switches that engage redundancy was challenging, especially that many switches (MOSFETs) need high-side gate drives with their source-side having various voltage levels. P-type MOSFETs were thus used in many locations and the board was successfully ran at 200W and nominal and many OC fault conditions.

The real-time control part is the most important step in this thesis, it determines the practicability of the new methodology. An FPGA board sbRIO-9612 and LabView software were used to implement all closed-loop control, fault diagnosis, and recovery algorithms. LabView has many differences from Simulink, such as the loop structures and very different block functions. Which caused a major redesign between the Simulink blocks and LabView
blocks while maintaining the same functionality. With the need for powerful computations before optimizing LabView block diagrams, a real-time module was considered to split computations between the FPGA and host computer—Basically, the closed-loop control and all measurements were still run on the FPGA module, while the logic systems were run on the host computer and interfaced with the FPGA through the real-time module. The communication lag between the FPGA module and the real-time module significantly increased the fault diagnosis and recovery time. To reduce the lagging of signal transmission, a first in first out (FIFO) buffer was used, but optimized LabView diagrams with minimum computations eliminated the need for the real-time module while being able to diagnose and recover from several faults and with closed-loop control all done on the FPGA.

Hardware testing was the final step of this thesis. Fault recovery under full power was limited with the power supply being limited to a certain current level that prevented SC testing at higher power. Two main methods were tried here to eliminate the inrush current—An input reverse diode in parallel with the supply, and a snubber circuit was designed to smooth the current transient parallel with each component in the system to decrease the inrush current. Neither of the methods can eliminate the inrush current thus lower power SC testing was performed.
VII. CONCLUSIONS AND FUTURE WORK

This thesis builds a generalized approach for intelligent fault diagnosis and recovery of power electronic system faults at the component level. The proposed approach was overviewed for two types of logic, combinational and fuzzy logic, and an example application platform was presented using a solar PV micro-inverter. Both methods that are part of the proposed approach were introduced and shown to have significant performance and simplicity advantages compared to the literature. The micro-inverter simulation model was validated using an experimental prototype to capture major dynamics, and then the simulation model was used to inject different OC and SC faults in the system. The number of voltage and current measurements was systematically reduced to minimized sensor requirements, and RMS, mean value, and THD of each measurement were calculated online. In experimental examples, the combinational logic was shown to be able to diagnose specific faults and engage redundant or spare components for system recovery. Fuzzy logic was also used to diagnose three faults. While combinational logic results were useful and achieved 100% fault diagnosis capability, fuzzy logic for the shown example provided faster fault diagnosis time. Future work will focus on achieving full power fault diagnosis and system recovery and applying the proposed approach to different power and energy systems. The redundancy strategy truly increases cost, however, when this method is implemented into systems that require high reliability, e.g. safety-critical systems, or systems as the micro-inverter where its lifetime does not match the rest of the system’s lifetime, e.g. PV panel vs. micro-inverter, it is reasonable to achieve higher reliability by increasing some cost.

The proposed methodology is also applied to all kinds of transmission and distribution line faults or power grid faults, and is not limited to a specific fault modes or locations. An online implementation of proposed methodology in grid system is a very important future
application, and implementation of the proposed methodology in a distributed sensor network environment is expected to achieve implementable fault diagnosis on limited digital platforms such as FPGAs and DSPs. Methods, like the one proposed in this thesis, for universal fault diagnosis and recovery instead of targeting specific components and subsystems, are expected to significantly increase the robustness of system operation and consolidate many measured signals into a set of useful measurements and quantities. They are expected to be of significant importance in higher-level control such as in supervisory control applications.
REFERENCES


[5] Yong Yu; Mingjun Zhu; Lianchun Yu; Tao Zhou; Dianguo Xu, "Novel method of inverter switch open-circuit fault diagnosis based on wavelet transform", in Proc. IEEE Power Electronics and Motion Control Conference (IPEMC), 2012, pp. 2338-2343


[19] Tamer Kamel; Yevgen Biletskiy; Christopher P Diduh; Liuchen Chang, “Failure detection of the capacitor bank of the three phase diode rectifier”, in Proc. *IEEE Electrical & Computer Engineering (CCECE)*, 2012, pp. 1 - 4


[33] Min-Jae Jung; Sang-Eun Park; Sung-Up Oh; Hak-Ju Lee; Se-Jin Seong, "Fault detection and isolation of two DC-DC converters parallel operation with single CT", in Proc. IEEE Industrial Electronics, 2001, pp. 1060-1065


Appendix A: Publications


Appendix B: Simulation Models

Micro-inverter:

DC/DC Converter

DC/AC Inverter

Fig. B1. Simulation of micro-inverter

$C_{out}$ pair with fault injection and recovery system:

Fig. B2. $C_{out}$ fault injection and recovery system
Table B1. Function of symbols in Fig. B2

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>R3</td>
<td>Capacitor current sensing resistor</td>
</tr>
<tr>
<td>OC</td>
<td>MOSFET for OC injection</td>
</tr>
<tr>
<td>SC</td>
<td>MOSFET for SC injection</td>
</tr>
<tr>
<td>S41</td>
<td>Isolation switch for failed component</td>
</tr>
<tr>
<td>B</td>
<td>Series switch for redundant component</td>
</tr>
<tr>
<td>C2</td>
<td>Fault injected component</td>
</tr>
<tr>
<td>C1</td>
<td>Redundant component</td>
</tr>
<tr>
<td>R2, R1</td>
<td>No-ideal series resistance</td>
</tr>
<tr>
<td>Cc, Csc</td>
<td>Fault injection signal with external control</td>
</tr>
</tbody>
</table>

Combinational logic quantity calculation:

![Combinational logic quantity calculation diagram](image)

Fig. B3. Quantity calculation

Threshold comparison and logic “1” lock:

![Threshold comparison and logic “1” lock diagram](image)

Fig. B4. Threshold comparison and signal lock design
Fault diagnosis:

An m-file was used to build the fault diagnosis system initial, since it has exactly the same function shown in Fig. 21. An example code is shown below.

```matlab
if isempty(Trip3) || isempty(Trip_state3)
    Trip3 = 0;
    Trip_state3 = 0;
end
if RIL1==0 && RVS2==1 && RVC1==0 && RVB==0 && RVB2==0 && MIL1==0 && MVS2==0 && MVC1==1 && MVB==0 && MVB2==0 && TIL1==1 && TVS2==0 && TVC1==0 && TVB==0 && TVB2==0
    Trip_state3=1;
    Trip3=1;
else if Trip3==0
    Trip_state3=0;
    end
end
y21=Trip_state3;
```

Fuzzy logic quantity calculation subsystem and fuzzy controllers:

![Fig. B5. Fuzzy controllers](image-url)
Transmission enable system and maximum output comparison:

Fig. B6. Transmission system and maximum output obtaining
Recovery signal lock by using signal lock design:

Fig. B7. Recovery signal lock
Appendix C: Real-time Control

PI current and voltage closed-loop control:

![Fig. C1. PI control in LabView](image)

Table C1. Function of symbols in Fig. C1

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vdc</td>
<td>Analog measurement of DC voltage</td>
</tr>
<tr>
<td>Offset</td>
<td>Offset regulation</td>
</tr>
<tr>
<td>V_ref</td>
<td>DC voltage reference value</td>
</tr>
<tr>
<td>Vdc16</td>
<td>Regulated DC voltage value</td>
</tr>
<tr>
<td>IL</td>
<td>Analog measurement of inductor current</td>
</tr>
<tr>
<td>IL_16</td>
<td>Regulated inductor current value</td>
</tr>
<tr>
<td>Sine</td>
<td>Carrier of PMW</td>
</tr>
</tbody>
</table>

PWM generation for boost converter:

![Fig. C2. PWM generation for boost converter](image)
Table C2. Function of symbols in Fig. C2

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Updated Period (DC)</td>
<td>PWM frequency of DC/DC converter</td>
</tr>
<tr>
<td>Triangle Wave 1D LUT</td>
<td>Triangle wave generation of DC/DC converter</td>
</tr>
<tr>
<td>PWM (DC)</td>
<td>DC/DC converter side PWM signal</td>
</tr>
</tbody>
</table>

PW generation for H-bridge:

![Fig. C3. PWM generation for H-bridge](image)

Table C3. Function of symbols in Fig. C3

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Updated Period (AC)</td>
<td>PWM frequency of DC/AC inverter</td>
</tr>
<tr>
<td>Triangle Wave 1D LUT2</td>
<td>Triangle wave generation of DC/AC inverter</td>
</tr>
<tr>
<td>PWM (AC)</td>
<td>DC/AC inverter side PWM signal</td>
</tr>
</tbody>
</table>

![Fig. C4. Deadband generation](image)
Table C4. Function of symbols in Fig. C4

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dead band</td>
<td>Dead band generation</td>
</tr>
<tr>
<td>Port3/DIO0</td>
<td>PWM signal for cross MOSFETs</td>
</tr>
<tr>
<td>Port3/DIO1</td>
<td>Out of phase PWM signal</td>
</tr>
</tbody>
</table>

Digital signal bank:

![Digital Switching Signal](image)

Fig. C5. Digital signal ports

Analog signal bank:

![Analog Collecting](image)

Fig. C6. Analog signal ports
Combinational logic measurements Vdc, VS2, and VD2 and injected faults L SC, S2 OC, and SB’ OC:

![Diagram of combinational logic system in LabView](image)

Fig. C7. Combinational logic system in LabView
Fuzzy logic with partial MFs:

Fig. C8. Partial fuzzy controller in LabView