2017

Development of Microgrid Test Bed for Testing Energy Management System

Shaili Nepal
South Dakota State University

Follow this and additional works at: http://openprairie.sdstate.edu/etd
Part of the Power and Energy Commons, and the Systems and Communications Commons

Recommended Citation

This Thesis - Open Access is brought to you for free and open access by Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange. For more information, please contact michaelbiondo@sdstate.edu.
DEVELOPMENT OF MICROGRID TEST BED FOR TESTING ENERGY MANAGEMENT SYSTEM

A thesis submitted in partial fulfillment of the requirements for the

Master of Science

Major in Electrical Engineering

South Dakota State University

2017
DEVELOPMENT OF MICROGRID TEST BED FOR TESTING ENERGY MANAGEMENT SYSTEM

This thesis is approved as a creditable and independent investigation by a candidate for the Master of Science in Electrical Engineering degree and is acceptable for meeting the thesis requirements for this degree. Acceptance of this thesis does not imply that the conclusions reached by the candidates are necessarily the conclusions of the major department.

Reinaldo Tonkoski, Ph.D.
Thesis Advisor

Robert Fourney, Ph.D.
Thesis Co-advisor

Steven Hietpas, Ph.D.
Head, Electrical Engineering and Computer Science

Dean, Graduate School
ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my advisor Dr. Reinaldo Tonkoski for his guidance and support and for his patience throughout my entire master’s program at South Dakota State University. His encouragement, suggestions helped me to understand and complete my research.

I am very thankful to Dr. Fourney for his guidance in organizing my thesis and also to Dr. Timothy Hansen for being my committee member. My sincere thanks goes to Mr. Jason Sternhagen for his continuous support and time. I would also like to thank Mr. Dan Flaskey for providing help and support in the microgrid lab. I am very thankful to entire Department of Electrical Engineering and Computer Science for supplying the necessary resources to complete my research work.

I would like to thank deepest to Dr. Santosh Chalise for his support and suggestions. I am thankful to Mr. Ayush Shakya, Mr. Habib Ullah, Mr. Riaz Ahmed, and Mr. Ali Aluwali and entire power research team.

Last but not the least, I am very grateful to my family for their support and encouragement.
2.5 Energy Management in Microgrid ........................................ 19

2.6 Microgrid Communication Protocol .................................... 19

2.6.1 Layer 1 Physical Layer ........................................... 22

2.6.2 Layer 2 Data Link Layer ......................................... 24

2.6.3 Layer 3 Network Layer ........................................... 35

2.6.4 Layer 7 Application Layer ........................................ 39

CHAPTER 3 PROCEDURES ....................................................... 40

3.1 Experimental Hardware and Software Procedures .................... 40

3.1.1 Load Control and Profile Generation .......................... 40

3.1.2 Generator Remote Start and Fuel Consumption Estimation .... 43

3.1.3 CAN communication interface between battery, inverter/charger,
and real-time controller ............................................. 45

3.1.4 PV Integration and Control ...................................... 55

3.2 Microgrid Test Bed Benchmark Design and Setup .................. 57

3.3 Energy Management in the Microgrid Test Bed ..................... 59

3.3.1 Case I, Microgrid with a real-time controller, a programmable load,
and 3.5 kW/6 kW generators ..................................... 59

3.3.2 Case II, Microgrid with 6kW inverter/charger, a 8 kWh battery sys-
tem, a real-time controller, a programmable load, and 3.5 kW/6 kW
generators ......................................................... 60

CHAPTER 4 RESULTS AND ANALYSIS .................................... 63

4.1 Generator Control ..................................................... 63
4.2 SoC Monitoring .................................................. 65
4.3 Inverter/Charger Control ........................................ 66
  4.3.1 Address Received for ISO Request for Address Claim .. 67
  4.3.2 ISO Address Claim ......................................... 68
  4.3.3 Charger Enable/Disable .................................... 68
  4.3.4 Charger Rate Control ....................................... 70
  4.3.5 Inverter Enable/Disable Controller ......................... 72
  4.3.6 Inverter Output Voltage and Frequency Monitoring and Control 74
  4.3.7 Inverter AC Input Voltage and Frequency Monitoring and Control . 76
4.4 PV Integration .................................................. 78
4.5 Case I, Microgrid with 3.5 kW/6 kW generators, a programmable load, and
  a real-time controller ............................................. 80
4.6 Case II, Microgrid with 3.5 kW/6 kW generators, an 8 kWh battery, a pro-
  grammable load, and a real-time controller ....................... 83

CHAPTER 5 CONCLUSIONS ........................................... 87
  5.1 Summary ....................................................... 87
  5.2 Conclusions ................................................... 88
  5.3 Future Work .................................................. 88

REFERENCES ....................................................... 89
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>Alternating Current</td>
</tr>
<tr>
<td>ACK</td>
<td>Acknowledgment</td>
</tr>
<tr>
<td>BMS</td>
<td>Battery Management System</td>
</tr>
<tr>
<td>CAN</td>
<td>Controller Area Network</td>
</tr>
<tr>
<td>CERTS</td>
<td>Consortium for Electric Reliability Technology Solutions</td>
</tr>
<tr>
<td>CRC</td>
<td>Cyclic Redundancy Check</td>
</tr>
<tr>
<td>DA</td>
<td>Destination Address</td>
</tr>
<tr>
<td>DB</td>
<td>D-Subminiature-B</td>
</tr>
<tr>
<td>DC</td>
<td>Direct-Current</td>
</tr>
<tr>
<td>DLC</td>
<td>Data Length Code</td>
</tr>
<tr>
<td>DP</td>
<td>Data Page</td>
</tr>
<tr>
<td>FACTS</td>
<td>Flexible AC Transmission</td>
</tr>
<tr>
<td>GE</td>
<td>Group Extension</td>
</tr>
<tr>
<td>HRDS</td>
<td>High-Reliability Distribution System</td>
</tr>
<tr>
<td>IIT</td>
<td>Illinois Institute of Technology</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standard Organization</td>
</tr>
<tr>
<td>Kbps</td>
<td>Kilo byte per second</td>
</tr>
<tr>
<td>kWh</td>
<td>Kilowatt-hour</td>
</tr>
<tr>
<td>ms</td>
<td>milliseconds</td>
</tr>
<tr>
<td>OSI</td>
<td>Open System Interconnection</td>
</tr>
<tr>
<td>P</td>
<td>Priority</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
</tr>
<tr>
<td>PDU</td>
<td>Protocol Data Unit</td>
</tr>
<tr>
<td>PF</td>
<td>Protocol Data Unit Format</td>
</tr>
<tr>
<td>PG</td>
<td>Parameter Group</td>
</tr>
<tr>
<td>PGN</td>
<td>Parameter Group Number</td>
</tr>
<tr>
<td>PS</td>
<td>Protocol Data Unit Specific</td>
</tr>
<tr>
<td>PV</td>
<td>Photovoltaics</td>
</tr>
<tr>
<td>RX</td>
<td>Received Signal</td>
</tr>
<tr>
<td>SA</td>
<td>Source Address</td>
</tr>
<tr>
<td>SAE</td>
<td>Society of Automobile Engineers</td>
</tr>
<tr>
<td>SOF</td>
<td>Start OF Frame</td>
</tr>
</tbody>
</table>
SPN Suspect Parameter Number

TCP/IP Transport Protocol Layer/Internet Protocol

VSI Voltage Source Inverter

TX Transmitted Signal
LIST OF FIGURES

Figure 1.1. Microgrid Test Bed Example [15]. .............................................. 3

Figure 2.1. Operation of solar cell [42]. ...................................................... 12

Figure 2.2. Equivalent I-V curve and Power curve of the PV cell. ................. 13

Figure 2.3. Grid Tied PV Inverter Connection. .............................................. 15

Figure 2.4. CAN DB9 Connector Pin Assignment [62]. .................................. 23

Figure 2.5. Voltage level of CAN bus [63]. .................................................. 23

Figure 2.6. CAN data frame format [65]. ..................................................... 25

Figure 2.7. CAN PGN [65]. ........................................................................... 26

Figure 2.8. PGN in 29 bits extended format. [69] ........................................ 28

Figure 2.9. Control field in CAN bus frame [65]. .......................................... 30

Figure 2.10. NAME format in SAEJ1939 [71][72]. ..................................... 36

Figure 3.1. LB-60-30 AC Load Bank. ............................................................ 41

Figure 3.2. Screen Shot of Load Profile Manager with Load Profile Editor. ...... 42

Figure 3.3. Required Load Profile for the 3.5 kW generator system. ................ 42

Figure 3.4. Required Load Profile for the 6 kW generator system. .................... 43

Figure 3.5. Generator connected with the simulator and the load bank. .......... 44

Figure 3.6. Battery inverter/charger connection with a real-time controller, a load
and the grid with the CAN bus control. ....................................................... 45

Figure 3.7. CAN-AC2-PCI card with two CAN channels. ............................. 46

Figure 3.8. Top view of the two modules of lithium iron battery [73]. ............. 47
Figure 3.9. Battery Management System Supervisor Control in Lithium Iron Phosphate Battery [73].

Figure 3.10. RJ45 to DB9 female adapter.

Figure 3.11. RJ45 to DB9 connection between real-time controller and the inverter/charger.

Figure 3.12. CAN ID formation for ISO Request PGN 59904 (EA0016).

Figure 3.13. Screen Shot of ISO Request for Address Claim.

Figure 3.14. CAN ID formation for Address Claim PGN.

Figure 3.15. Screen Shot of ID formation for Address Claim Send.

Figure 3.16. Microgrid with 6 kW inverter/charger, an 8 kWh battery system, a real-time controller, a programmable load, and a 3.8 kW PV.

Figure 3.17. Energy management flowchart of microgrid with 6 kW inverter/charger system, an 8 kWh battery system, a real-time controller, a programmable load, and a 3.8 kW PV.

Figure 3.18. Laboratory scaled microgrid test bed at South Dakota State University.

Figure 3.19. Microgrid with a real-time controller, a programmable load, and 3.5 kW/6 kW generators.

Figure 3.20. Microgrid with 6 kW inverter/charger system, an 8 kWh battery system, a real-time controller, a programmable load, and 3.5 kW/6 kW generators.

Figure 3.21. Energy Management Flowchart of Microgrid with 6 kW inverter/charger system, an 8 kWh battery system, a real-time controller, a programmable load, and 3.5 kW/6 kW generators.
Figure 4.1. 3.5 kW Generator Control. ........................................ 63
Figure 4.2. 6 kW Generator Control. ........................................ 64
Figure 4.3. SoC monitoring during charging the battery at 30 A charge rate. . . 65
Figure 4.4. SoC monitoring during discharging of the battery at 1kW load. . . 66
Figure 4.5. ISO Request for address claim data and NAME received from the inverter. ........................................ 67
Figure 4.6. Inverter/Charger NAME per Industry Standard. .................... 68
Figure 4.7. Charger Enable and Disable Control. .............................. 69
Figure 4.8. Charger response to enable command. .............................. 69
Figure 4.9. Acknowledgment Message Received for Inverter Control. ........ 70
Figure 4.10. Charger Rate Control. ............................................. 71
Figure 4.11. Charger response to the charge rate command. .................... 71
Figure 4.12. Charge Rate Acknowledgment Message. ......................... 72
Figure 4.13. Inverter enable and disable Control. .............................. 73
Figure 4.14. Inverter response to enable command. .............................. 73
Figure 4.15. Inverter enable and disable control acknowledgment message. . 74
Figure 4.16. Voltage and frequency monitoring of the battery inverter/charger. 75
Figure 4.17. Control of voltage and frequency battery inverter/charger through CAN controller. ........................................ 75
Figure 4.18. Response of inverter to voltage and frequency change command. 76
Figure 4.19. Control of voltage and frequency of the battery inverter AC input. 77
Figure 4.20. Inverter response to input voltage and frequency change command. 77
Figure 4.21. PV inverter integration to the battery inverter and the load system. 78
Figure 4.22. Microgrid with 3.8 kW PV, an 8 kWh battery, a 6 kW inverter/charger, and a programmable load. ................................. 79

Figure 4.23. Microgrid with the 3.5 kW generator, a programmable load, and a real-time controller. ........................................... 81

Figure 4.24. 3.5 kW generator’s efficiency measured with power output ......................................................... 82

Figure 4.25. Microgrid with a 6 kW generator, a programmable load, and a real-time controller. ................................. 82

Figure 4.26. 6 kW generator’s efficiency measured with power output ......................................................... 83

Figure 4.27. Microgrid with a 3.5 kW generator, an 8 kWh battery, two 6 kW inverters/chargers, a programmable load, and a real-time controller. ... 84

Figure 4.28. Microgrid with a 6 kW generator, an 8 kWh battery, two 6 kW inverters/chargers a programmable load, and a real-time controller. ........ 85
LIST OF TABLES

Table 1.1. Some Institutional Microgrid Test Beds in North American Region [23], [31], [33]–[35]. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

Table 2.1. Comparison of some for commercially available batteries [39][49]. . . . . 16

Table 2.2. ISO 11783 specification of OSI Model Layers. . . . . . . . . . . . . . . 20

Table 2.3. ISO 11898 scope in OSI model. . . . . . . . . . . . . . . . . . . . . 21

Table 2.4. SAE J1939 specification of OSI Model Layers. . . . . . . . . . . . . . 21

Table 2.5. CAN specification of Physical Layer. . . . . . . . . . . . . . . . . . . . 22

Table 2.6. DB9 Pin Assignments to CAN standard . . . . . . . . . . . . . . . . . 23

Table 2.7. Priority Setting in CAN bus. . . . . . . . . . . . . . . . . . . . . . . . 27

Table 2.8. PDU and PS Format. . . . . . . . . . . . . . . . . . . . . . . . . . . . 28

Table 2.9. Acknowledgment Parameter Group Format [70]. . . . . . . . . . . . . . 32

Table 2.10. Positive Acknowledgment Parameter Group Data Byte Format [70]. . . 33

Table 2.11. Negative Acknowledgment Parameter Group Data Byte Format [70]. . . 33

Table 2.12. Access Denied Parameter Group Data Byte Format [70]. . . . . . . . . 34

Table 2.13. Not Responding Parameter Group Data Byte Format [70]. . . . . . . . . 34

Table 2.14. Address Management Messages Format [71]. . . . . . . . . . . . . . . 35

Table 2.15. Address Claimed Message Format [71]. . . . . . . . . . . . . . . . . . 38

Table 3.1. Payloads of Address 0x02. . . . . . . . . . . . . . . . . . . . . . . . . 54

Table 4.1. Significance of Positive Acknowledgment Parameter Group Data Byte Format and Acknowledgment Message Received from Inverter [70]. . . 70
ABSTRACT

DEVELOPMENT OF MICROGRID TEST BED FOR TESTING ENERGY MANAGEMENT SYSTEM

SHAILI NEPAL

2017

Today the world population has reached 7.5 billion, and this number is expected to grow at the rate of 1.13% every year [1]. With this increase in population, the total demand for electricity has also increased. More people means the need for more power: electricity to power homes, schools, industries, hospitals, and so on. In today’s world, where most of the daily activities are dependent on electricity, demand for electricity, therefore, continues to rise. Currently, managing this growing need for electricity is one of the challenges the world is facing. In addition to this, approximately 1.2 billion people live in remote parts of the world where the electricity supply is either limited or non-existent [2]. Providing an affordable and easily available source of electricity to this population is another challenge. In response to these challenges, a significant number of countries are investing in the integration of renewable resources for energy production. Renewable resources such as the sun, wind, and water are free, clean, and readily available. Remote and poor parts of the world can also benefit by utilizing these available energy sources for electricity generation. The use of renewables helps to decrease the overall cost of electricity generation as well. This need for clean and safe energy has contributed to creating and promoting the concept of microgrids around the world.
Microgrids are defined as small-scale power distribution networks with distributed energy sources, loads, and storage. They can operate in either grid-connected or islanded mode. Renewable sources are intermittent in nature, and uncertainties are always present in the microgrid operation when using these resources. The Energy Management technique is required for the coordination of these resources in order to mitigate the potential risks. Some studies have been conducted in the area of microgrid operation, stability, and control, and various types of laboratory-based microgrid test beds have been developed. A microgrid test bed allows testing of scaled down systems in order to test and simulate large real-world microgrid projects. The objective of this study is to develop a reconfigurable microgrid test bed. This test bed is created on a laboratory scale and is capable of testing energy management algorithms to validate real-time operation. A novel approach to automatic microgrid operation is proposed with the use of commercial off-the-shelf equipment and the Controller Area Network (CAN) protocol. The OPAL-RT 5600 real-time simulator is used as a central controller for controlling and scheduling microgrid sources to supply the load, charge the battery and, read a state of charge values. The CAN communication protocol is used by the controller to control and coordinate different components. Different cases are studied in order to support the reconfigurability, automatic operation, and energy management in the microgrid test bed using the CAN bus.
CHAPTER 1 INTRODUCTION

1.1 Background

Today, 16% of the world’s population has limited or no access to electricity [2]. In remote parts of the world, in particular, power grid access and expansion constitute a massive problem; this is mostly due to high setup costs caused by the lack of infrastructure such as road networks. But a large number of people live in these remote areas [3]. Accordingly, the issue of supplying affordable and readily available sources of electricity to this population is a challenge that energy scientists have long sought to overcome [4].

One of the most popular sources of electricity in remote areas is fossil fuel-powered generators. Diesel generators, for example, are widely used due to their low-cost installation [5]. However, fossil fuels are expensive and tend to pollute the environment. High costs associated with fuel, transportation, maintenance, and fuel storage can drive electricity prices up to $2.45/kWh for these types of generators [6].

The use of conventional fossil fuels such as diesel in electricity production gives rise to pollution. Upon burning, fossil fuels release hydrocarbons. These hydrocarbons mix with other components in the atmosphere to produce carbon dioxide and sulfur dioxide [4]. Carbon dioxide leads to the greenhouse effect [7], and sulfur dioxide contributes to acid rains [8]. The overuse of fossil fuels also results in their permanent depletion, since sources of these elements are finite and diminish with time [9].

Another major source of energy in today’s world is nuclear power [10]. Nuclear energy is expensive, can have an adverse effect on the environment; this is mainly because of the radioactive wastes it produces. This waste is very harmful to humans and the
environment alike. Because of all these facts, researchers have focused more on the use of renewable and clean resources for electricity generation [11].

A microgrid is a localized, low-voltage distribution network comprised of various controllable loads, storage devices, and distributed generators. A microgrid can work under a grid connection or in an isolated mode. The grid connection occurs through a point of common coupling. Microgrids usually serve precisely targeted areas and are capable of detaching from the main grid when disturbances occur [12].

Microgrids can integrate distributed renewable resources in energy production, which can be a good medium for generating cheap and clean electricity [13]. There are several benefits of integrating renewable resources in the process of electricity generation. Renewable resources are mostly safe for the environment since they do not release harmful gas emissions while generating electricity [14]. These resources are also abundant in nature and are free, so energy produced from them can be more economical [14]. Distributed energy sources in microgrids include photovoltaic arrays, wind turbines, and micro-turbines. Microgrids, therefore, are receiving a great deal of attention in the research field. Their proper use and integration can offer a viable alternative to fossil fuel-powered energy generation systems [12].

However, renewable energy sources are intermittent in nature. Thus, any power generation mechanism that depends on renewable sources must take into account this unpredictable behavior. Supply and coordination of these resources in microgrids require proper control and monitoring. Energy management algorithms can partially compensate for these shortcomings and facilitate the stable operation of a microgrid. These algorithms are useful in cases where the energy sources are not dependable or are intermittent. Figure
1.1 provides an example of a typical microgrid and its components.

![Figure 1.1. Microgrid Test Bed Example [15].](image)

1.2 Previous Work

A considerable amount of work has been done all over the world in the field of microgrids [12]. Many countries and organizations are taking interest in implementing different kinds of microgrid test beds.

The Consortium for Electric Reliability Technology Solutions (CERTS) presented a microgrid concept in 1998 in Columbus, Ohio [16]. The objective of CERTS is to demonstrate the integration of micro-sources in a microgrid. The study explains the actual meaning of the CERTS concept: the automatic transition between the grid-connected and the islanded modes of operation [17]. The test bed consists of distributed energy resources with backup generators, along with storage devices and an autonomous controller; there is no central controller [16], [18], [19]. The CERTS microgrid works on the peer-to-peer connection: the type of connection in which every component of the system is connected to every other component. The microgrid can function in islanded mode when there is any fault, or when the power quality is low. In this microgrid, the central communication function is used only to dispatch the set points of the distributed generators. According to
the study, storage devices are integrated for stability [20]. The primary limitation of this test bed includes the lack of accurate reporting of State of Charge (SoC) to the power conversion system due to the absence of a reliable medium.

Another microgrid test bed, built based on CERTS’s concepts, has been installed at the Santa Rita jail in California. This microgrid consists of 1.2 MW diesel generators, 1.2 MW photovoltaics, 1 MW fuel cells, wind turbines, and energy storage [20]. The primary objective of the microgrid is to provide a continuous power supply in case of an outage or blackout. This microgrid is an example of CERTS’s microgrid concept, demonstrating a peer-to-peer connection with the integration of large-scale energy storage [21]. This microgrid includes a 2 MW lithium-ion battery to provide power during the transition phase [20]. Generators and storage inverters control the voltage and limit the reactive power flow between the sources. However, according to a study, the microgrid appears to have calibration issues [22]. This test bed lacks a reliable communication medium reporting accurate values.

The University of Wisconsin microgrid was also developed with the CERTS concept [23][24]. The test bed was established to study the integration of generators with renewable sources. The microgrid consists of a commercial 10 kW diesel generator. The test bed has a voltage controller in gen-set for regulating the voltage at a constant level. This test bed has the inverter-based source [25] with a custom-made inverter control based on Digital Signal Processing (DSP).

Coordinating with Galvin Electricity, Illinois Institute of Technology (IIT) proposed a campus-wide microgrid. The microgrid contains a gas generator unit with a capacity of 4 MW, a wind turbine, and 500 kWh of storage in the form of a battery
The objective of this microgrid is to supply the campus with a load of around 9 MW. The microgrid has a master controller architecture with hierarchical control through Supervisory Control and Data Acquisition (SCADA) systems [26]. A High-Reliability Distribution System (HRDS) was also implemented in the microgrid; this type of system experiences fewer outages and maintains the high level of system reliability. The study mentions that IIT’s microgrid works on multi-tier hierarchical control with resynchronization and the islanded mode of operation [27]. The study discusses the benefits of intelligent communication and the control system integration [28]. Nevertheless, this microgrid has very large and expensive components and is only targeted for urban areas with high-voltage networks and customized inverter control.

Three different microgrids have been installed at the University of Texas at Arlington [29]. These microgrids can work in islanded or grid-connected mode. The microgrid test bed features three microgrids that work either independently or in a connected manner. The unique approach of this project is the use of Compact Rio as an embedded control with the interconnection of three microgrids. This reconfigurable real-time controlled test bed includes a total of 2.76 kW PV, 1.2 kW wind turbines, 1.2kW Proton Exchange Membrane (PEM) fuel cells, 12 V platinum gel batteries, 1400 W (GTX21424) and 2500 W (GTX2524) rated inverters, a 6 kW diesel generator, and programmable 3.6 kW loads. This work focuses on the development of the microgrid control system by a state machine model. The system uses peer-based communication strategies with commercial off-the-shelf components for hardware setup. The study mentions the microgrid’s operation in grid-connected and isolated modes in dynamic operation but does not explain the steady operation.
Another example of a microgrid test bed demonstration is that of Kythnos Island. This microgrid has a three-phase low-voltage network designed to supply 12 houses on the island. The test bed consists of 10 kW PV in two different locations and, a 5 kW diesel generator with a 53 kWh battery capacity [30]. Battery inverters can operate in isochronous or droop mode and can vary the output frequency for providing information to the PV inverters and the load controller. However, this test bed has constraints in terms of supplying to the load when SoC is low.

In Canada, a campus-based microgrid at British Columbia Institute of Technology was implemented around 2009. This microgrid has a capacity of 1.2 MW power generation, a central control system, and two wind turbines of 10 kW with a 250 kW thermal turbine. The microgrid also has a 300 kW PV system with a battery storage of 550 kWh. The study shows that the test bed has a command control unit for the microgrid’s operation control [31].

In China, a stand-alone microgrid test bed was recently developed [32]. The microgrid has a PV capacity of 100kW, with 210 kW of wind turbines. The diesel generator has a capacity of 200 kW, and the storage system has lead acid batteries with 960 kWh capacity. The master-slave strategy is implemented with the battery and generator, respectively, in turn providing the reference voltage and frequency. The control algorithm is based on the SoC values of the batteries. However, this project has high maintenance cost due to its remote location. Table 1.1 below shows the overview of the different microgrid test beds in North American regions.
Table 1.1. Some Institutional Microgrid Test Beds in North American Region [23], [31], [33]–[35].

<table>
<thead>
<tr>
<th>Name and Location</th>
<th>Type</th>
<th>Components</th>
<th>Control</th>
<th>Capacity (MW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CERTS, Ohio</td>
<td>AC</td>
<td>Gas Powered Sources and Loads</td>
<td>Decentralized</td>
<td>0.2</td>
</tr>
<tr>
<td>Santa Rita Jail, California</td>
<td>AC</td>
<td>Diesel Generator, PV, Fuel Cell, Wind Storage and Loads</td>
<td>Decentralized</td>
<td>5</td>
</tr>
<tr>
<td>UW Madison Wisconsin</td>
<td>AC</td>
<td>Diesel Generator, PV and Loads</td>
<td>Decentralized</td>
<td>0.02</td>
</tr>
<tr>
<td>IIT Microgrid, Chicago, Illinois</td>
<td>AC/DC</td>
<td>PV, Gas Turbines, Wind, Battery Storage, Building Controllers, HRDS Switches, Meters, Backup Diesel Generator and Loads</td>
<td>Centralized</td>
<td>9</td>
</tr>
<tr>
<td>Microgrid at Albuquerque, New Mexico</td>
<td>AC</td>
<td>PV, Gas Generator, Fuel Cell, Storage, and Loads</td>
<td>Decentralized</td>
<td>2.5</td>
</tr>
<tr>
<td>UT Arlington, Texas</td>
<td>AC</td>
<td>PV, Wind, Fuel Cells, Diesel Generator, Storage and Loads</td>
<td>Decentralized</td>
<td>0.01</td>
</tr>
<tr>
<td>Sandia National Laboratories, DC</td>
<td>DC</td>
<td>PV, Diesel Generator, Storage and Loads</td>
<td>Decentralized</td>
<td>0.06</td>
</tr>
<tr>
<td>University of California, San Diego</td>
<td>AC</td>
<td>PV, Gas and Steam Turbine, Fuel Cell, Thermal Storage and Loads</td>
<td>Centralized</td>
<td>18-22</td>
</tr>
<tr>
<td>Laboratory Scale Microgrid, New Jersey</td>
<td>AC</td>
<td>PV, Storage and Loads</td>
<td>Centralized</td>
<td>0.01</td>
</tr>
<tr>
<td>UT Austin, Texas</td>
<td>AC</td>
<td>Diesel, Gas Generator, Storage and Loads</td>
<td>Centralized</td>
<td>0.01</td>
</tr>
<tr>
<td>Florida International University</td>
<td>DC</td>
<td>PV, Wind, Fuel Cell, and Loads</td>
<td>Centralized</td>
<td>5</td>
</tr>
<tr>
<td>Hawaii Hydrogen park, Hawaii</td>
<td>DC</td>
<td>PV, Wind, Fuel Cell Storage and Loads</td>
<td>Decentralized</td>
<td>0.03</td>
</tr>
<tr>
<td>BCIT Microgrid, BC, Canada</td>
<td>AC</td>
<td>PV, Wind, Steam, Storage and Loads</td>
<td>Centralized</td>
<td>1.2</td>
</tr>
</tbody>
</table>
All in all, it was found from the previous work, an economical test bed was necessary having reliable and flexible communication medium, reporting correct status, commercial off-the-shelf control, and demonstrating the reconfigurable capability to study the real microgrid projects. This thesis proposes a new approach for the development of a steady stand-alone laboratory-scale low-voltage test bed. The novel idea is the use of a real-time controller for the central management system with a CAN bus protocol. The design uses a commercial off-the-shelf approach to inverter control in order to create an economical test bed exhibiting reconfigurable capacity. The test bed detailed in this study provides a real-time digital simulator integrated with flexible CAN protocol into the system in order to provide a platform for researchers to test their energy management algorithms.

1.3 Motivation

The motivation behind this thesis is the need for a fully controllable, economical, and reconfigurable microgrid test bed in order to study different energy management systems.

1.4 Objective

The main objective of this thesis work is to develop a reconfigurable microgrid test bed which that is able to implement different energy management algorithms. In addition, this research study work includes several other tasks to support the main objective. These key tasks are as follows:
Task 1: Development of real-time control module of generator

Task 2: Development of real-time control module of battery and inverter system

Task 3: Interfacing of microgrid components to with real-time digital simulator

Task 4: Development of laboratory-scale microgrid test bed

Task 5: Testing and validation of the energy management system algorithm

1.5 Contributions

The main contribution of this thesis is to develop an automated low-voltage microgrid test bed using commercial off-the-shelf inverter control. This study uses a flexible CAN protocol for interconnection and dispatching of energy resources.

1.6 Thesis Outline

This report is organized into five chapters. Chapter 2 describes different various theories supporting the research, along with different microgrid types, their components, communication protocols used in this system. This chapter also discusses energy management and microgrid control procedures. Chapter 3 sheds light on the procedures followed in this study. This chapter covers different hierarchical designs, procedures, and setups of the microgrid test bed with all the components, and also outlines hardware and software setup requirements for the microgrid test bed. Chapter 4 provides all the necessary results and explanations to support this work. Simulation results, experimental results, and discussions are included here. Finally, Chapter 5 presents the summary and the conclusion of this thesis, along with a discussion of the limitations of this study and possibilities for future work.
CHAPTER 2 THEORY

This chapter describes the theories related to the development of a microgrid test bed. Section 2.1 discusses the different types of microgrids, Section 2.2 covers the different microgrid test bed components, Section 2.3 includes the energy management strategy used in microgrids, and Section 2.4 explains the network communication protocol used in microgrids.

2.1 Microgrid test bed

The motivation for developing the microgrid test bed is to provide a platform for the integration of different energy resources and to help study microgrid functions, stability, integration, and energy management mechanisms. According to [36], there are five categories of microgrids. The first type is a “Remote Microgrid”, which is common in remote areas such as islands, and operates solely in an isolated mode. This type of microgrid does not have a connection to a larger main grid. The second type is a “Campus Microgrid”, which is connected to the local or main grid and can also work in an isolated mode. This type of microgrid usually serves campuses, military bases, and prisons. The other three types are Community, District Energy, and Nano Microgrid.

2.2 Microgrid test bed components

The microgrid test bed presented in this study was built at South Dakota State University’s microgrid lab. The test bed has a photovoltaic system, natural gas generators, battery banks, an inverter system, a programmable AC load, and a real-time controller.
2.2.1 Generators

Generators are electromechanical devices that convert mechanical energy into electrical energy [37]. They typically operate using natural gas or diesel as input and produce electricity as an output. Generators are the primary source of energy in many microgrids and are mainly used in areas where there is no expansion of the main grid. They can also be used as an emergency backup when the main grid is unable to supply power.

An engine is the both main component of a generator and the main source of mechanical energy. The size of the generator engine depends on the amount of power it has to generate [37]. Similarly, the amount of fuel consumed by a generator depends on the electrical output it produces. Generators display improved efficiency when operated at high load [38]. The efficiency of a generator is given by the ratio between electrical output and the mechanical input.

\[ \eta = \frac{kW_{\text{output}}}{HP_{\text{input}}/0.746} \times 100 \]  

(2.1)

In Equation 2.1, \( \eta \) and \( HP \) denote the efficiency of the generator and the input horsepower respectively. The fuel consumption can be estimated by a polynomial equation of degree 2, and is measured in volume per hour [38].

\[ Y = aP^2 + bP + c \]  

(2.2)

In Equation 2.2, a, b and c are fuel curve coefficients, and P is the output power of the generator. The fuel curve coefficient’s value depends upon the type of generators [39].
Generators are equipped with a system called a governor, which controls the power output. The governor in a generator controls the speed of the generator by controlling fuel and maintaining the frequency. The frequency of the generator with engine speed is given by,

\[ f = \frac{\omega \ast P}{120} \]  

(2.3)

where, \( \omega \) (RPM) is the speed of the engine, and \( P \) is number of poles of the machine.

2.2.2 Photovoltaics

The process of converting solar power into direct electricity using semiconductor materials like solar panels is called Photovoltaic (PV). In this process, an electric current is produced from the photoelectric effect: that is, electrons produce current when the semiconductor material’s surface is exposed to light. Silicon is the most popular chemical element used in the production of solar panels [40]. However, the working principle of a solar panel is the same irrespective of what semiconductor materials it is built with [41].

Figure 2.1. Operation of solar cell [42].
Figure 2.1 shows the operating principle of the solar cell. When solar light strikes the surface, electrons are excited and leave their positions, consequently forming holes. When there is an accumulation of electrons and holes on either side, the depletion region in between shrinks and the current flows. When a wire is connected to the load through a solar cell, it creates a path for electrons to flow, thus generating current [41], [43].

The current-voltage characteristic of the solar cell is given by the I-V curve. The relation between the solar cell voltage and current is not linear. The solar curve showing the voltage and current relationship are depicted in Figure 2.2.

\[
P = V_m I_m
\]  
(2.4)
Many solar cells are connected in series and in parallel in order to form a PV module capable of generating enough output power. PV arrays consist of several such PV modules, either arranged in series, in parallel, or in series-parallel combinations to achieve the required voltage and current. Depending upon the total power requirement of the system, a single module or multiple PV arrays can be used [44]. The PV system can be configured as a stand-alone or as a part of a grid interface. Stand-alone systems usually require storage capabilities such as batteries to store power when sunlight is unavailable.

The ability of a solar cell to convert sunlight into electricity is measured by the conversion efficiency. The efficiency of a solar cell is measured by dividing the output power by the incident solar power [40]. The formula for solar cell efficiency is given by:

\[ \eta = \frac{P_m}{EA_c} \]  

(2.5)

In equation 2.5, \( \eta \) is the efficiency of the solar cells, \( P_m \) is the maximum output power, \( E \) is the average irradiance, and \( A \) is the area of the panel. The standard test conditions of the photovoltaic cell are irradiance 1000 \( W/m^2 \) at air mass (AM) 1.5, and the temperature 25\(^\circ\)C.

2.2.3 Inverters

Inverters are electronic devices that change direct current into alternating current. Inverters can operate in a grid-tied connection or stand-alone mode. They do not produce power on their own, but instead convert the power from DC sources. Grid-connected inverters work without batteries, and the stand-alone variety rely on batteries for the input voltage. Inverters can be either voltage source or current source.
A Voltage Source Inverter (VSI) has a voltage waveform as an AC output, which is independently controlled [45]. Due to this property, VSIs have many industrial applications such as adjustable speed drives and FACTS (Flexible AC Transmission) power systems [45]. A Current Source Inverter (CSI) has an AC output which is independently controlled. The output current waveform remains mostly unaffected by the load. CSIs are widely used in medium-voltage industrial applications where a high-quality waveform is required [45].

Figure 2.3 shows the connection of a grid tied inverter with a PV source and the grid. PV arrays are connected to the grid via inverters. Grid-tied inverters connect the grid directly to PV arrays and provide AC power to the grid. These inverters typically include the feature of Maximum Power Point Tracking control to produce maximum power [46].

![Diagram of Grid Tied PV Inverter Connection](image)

**2.3**

Figure 2.3. Grid Tied PV Inverter Connection.

Stand-alone inverters are used in battery connected systems. Generally, these inverters are bidirectional with both charging and inverting capabilities [39]. Stand-alone inverters can be connected to AC sources such as generators to charge the batteries.
2.3 Energy Storage

Renewable sources of energy are intermittent in nature. They are extremely stochastic and difficult to forecast with a high degree of accuracy. Energy storage within a microgrid is needed in order to mitigate this variability of renewable resources. Storage devices can be used to store the power when renewable sources are present; when these sources are not present, the stored power can be used to supply the load.

Forms of energy storage include pumped hydro, batteries, flywheels, super-capacitors, compressed air storage, and thermal energy storage [47]. The most common storage devices used in microgrids are batteries, which are electrochemical devices that convert chemical energy into electrical energy [48]. The conversion of energy from one form to another has two characteristics: power density and electrical density. Power density is related to the amount of energy stored in the device, and electrical density is the rate at which energy is transferred from the storage device [48]. Table 2.1 shows the comparison of several factors of the commercially available batteries.

Table 2.1. Comparison of some for commercially available batteries [39][49].

<table>
<thead>
<tr>
<th>Type</th>
<th>Energy Density (Wh/kg)</th>
<th>Power Density (W/kg)</th>
<th>Efficiency (%)</th>
<th>Life Cycles Cycles</th>
<th>Limitation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead Acid</td>
<td>20-35</td>
<td>25</td>
<td>70-80</td>
<td>200-2000</td>
<td>Short Life Cycle</td>
</tr>
<tr>
<td>Lithium ion</td>
<td>100-200</td>
<td>360</td>
<td>70-85</td>
<td>500-2000</td>
<td>High Cost</td>
</tr>
<tr>
<td>NiMH</td>
<td>60-80</td>
<td>220</td>
<td>50-80</td>
<td>&gt;3000</td>
<td>High self discharge</td>
</tr>
<tr>
<td>NiCd</td>
<td>40-60</td>
<td>140-180</td>
<td>60-90</td>
<td>500-2000</td>
<td>Cost</td>
</tr>
<tr>
<td>Supercapacitor</td>
<td>4.5</td>
<td>3500</td>
<td>85-98</td>
<td>1,000,000</td>
<td>High Cost</td>
</tr>
</tbody>
</table>
Lithium-ion batteries have a high energy density and efficiency compared to lead acid batteries. Although work on the lithium battery began in 1912 under G.N. Lewis, it was not until 1970 that non-rechargeable lithium batteries were available on the market for commercial purposes [50]. Today, lithium-ion batteries are rechargeable and suitable for portable devices. Due to their high energy density, lithium-ion batteries are the best-suited technology for plug-in hybrid and electric vehicles [51]. The specific energy density is roughly 200 W h/kg, which is double the energy density of nickel metal hydride or nickel–cadmium batteries [49]. But lithium-ion batteries require a protection circuit for safe operation. They have a relatively low self-discharge and low maintenance costs [50].

The condition and status of a battery for further use are given by an important factor known as the state of charge (SoC) [52]. SoC is defined as the percentage of the maximum battery capacity [53]. SoC is measured in percentage points, with 100% representing a fully charged battery and 0% representing an empty battery [54]. The formula to calculate state of charge is given by [38],

During Charging,

\[
SoC(t + 1) = SoC(t) + \frac{\eta_{\text{charging}} \cdot P_{b,t} \cdot \Delta t}{BattCap_{kWh}}
\]  

(2.6)

During Discharging,

\[
SoC(t + 1) = SoC(t) + \frac{P_{b,t} \cdot \Delta t}{\eta_{\text{discharging}} \cdot BattCap_{kWh}}
\]  

(2.7)

In equation 2.7, \( \eta \) is charging and discharging efficiency of the battery, \( P_{b,t} \) is the maximum charge and discharge rate.
2.4 Real-time digital controllers

A real-time digital simulator is the multiprocessor system designed for real time simulations. These simulators are becoming popular in the simulation of power electronics because they reduce the overall system cost [55].

There are several benefits of using real-time simulators. First, they can be time savers: these simulators allow users to design, run and validate systems with error checking in a single platform. Second, they lower the test system’s total cost: large power devices are expensive, and real-time simulators can provide a platform to model and test these devices. Third, these simulators also lower the risks involved in working with real devices, and are considered safe for testing [56].

The test bed in this study uses the real-time digital simulator manufactured by OPAL RT Inc. The simulator is an OP5600 series with OPAL RT software RT-LAB installed in the system. RT-LAB comes with its library, and is installed with different software such as MATLAB/Simulink and LabVIEW. The common platform used with RT-LAB to perform real-time simulation is MATLAB/Simulink.

The other feature of the simulator is the OP8660 data acquisition system. This system has the capability to take 600V and 15A current for data acquisition. The OPAL RT system features a host computer and a target computer connected to each other using the TCP/IP protocol. The host computer has Matlab/Simulink with RT-LAB installed. This software is used for development and compiling of the Simulink model. The target computer (on the other side) has a Linux or QNX operating system, and is responsible for real-time input/output execution.
2.5 Energy Management in Microgrid

To produce electricity, a microgrid can have distributed energy resources and can include renewable energy sources. Renewable sources like the sun, wind, and water are integrated to supply the local load in a microgrid. Such sources are intermittent and often their output cannot be predicted, hence proper coordination of resources is required to produce enough power to meet the load demand. The main function of an Energy Management System is to coordinate different types of sources, controllers, storage, and loads in a test bed. In the centralized energy management strategy, a central controller is present to control all the microgrid resources. This central controller acts as the brain of the system and stores the required information and data about the resources. The controller contains information like operating limits of the resources, solar and load forecasted values, cost functions, set points, state of charge values, security and reliability factors, and status of the components [57]. A reliable communication link is required for the master controller to communicate and control the resources.

2.6 Microgrid Communication Protocol

Microgrid components need a communication channel to communicate with each other in the system. In the case of central control, reliable communication is required in order to maintain coordinated control between the master and slave units. Controller Area Network, commonly known as CAN, is used as a microgrid communication network. CAN is the medium for transferring data, set points, and control signals, and monitoring SoC values in the microgrid network. It is a serial communication protocol used in the
system to communicate between the electronic control units [58], and was initially used in the field of automotive engineering. CAN was developed by Robert BOSCH GmBH in the early 1980s. In 1991, Bosch introduced CAN 2.0 specifications breaking CAN into part A and part B. Part A and Part B are distinguished by the types of identifiers used [59]. Later in 1993, CAN was standardized by the International Organization for Standardization in ISO 11898. The ISO 11898 standard uses the physical and data link layers in the communication protocol. With the evolution of different standards for CAN in the automobile industry, ISO 11783 was introduced in the field of agriculture to control the operation of tractors and heavy machinery. SAE J1939, which is the Society of Automotive Engineers standard, is the higher standard of a CAN vehicle bus network; this standard has been adopted by the generator industries [60].

Architectures similar to SAE J1939-11, ISO 11783, and ISO 11898 are used in this project to control the inverter/charger. The SAE J1939-11 standard is the twisted pair with the transmission rate up to 250 kbps. Table 2.2 shows the application of an Open Systems Interconnection (OSI) model to ISO 11783 [61].

<table>
<thead>
<tr>
<th>OSI Model Layers</th>
<th>ISO 11783</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Layer</td>
<td>Priority, SA, DP</td>
</tr>
<tr>
<td>Presentation Layer</td>
<td></td>
</tr>
<tr>
<td>Session Layer</td>
<td></td>
</tr>
<tr>
<td>Transport Layer</td>
<td></td>
</tr>
<tr>
<td>Network Layer</td>
<td></td>
</tr>
<tr>
<td>Data Link Layer</td>
<td>Logical Link Control</td>
</tr>
<tr>
<td></td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>Physical Layer</td>
<td>Physical</td>
</tr>
</tbody>
</table>
ISO 11783 and ISO 11898 are similar related standards. Table below 2.3 shows the application of OSI model to ISO 11898 [59].

Table 2.3. ISO 11898 scope in OSI model.

<table>
<thead>
<tr>
<th>OSI Model Layer</th>
<th>ISO 11898</th>
<th>Scope of ISO 11898</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Layer</td>
<td>Not Specified</td>
<td></td>
</tr>
<tr>
<td>Presentation Layer</td>
<td>Not Specified</td>
<td></td>
</tr>
<tr>
<td>Session Layer</td>
<td>Not Specified</td>
<td></td>
</tr>
<tr>
<td>Transport Layer</td>
<td>Not Specified</td>
<td></td>
</tr>
<tr>
<td>Network Layer</td>
<td>Not Specified</td>
<td></td>
</tr>
<tr>
<td>Data Link Layer</td>
<td>Logical Link Control Medium Access Control</td>
<td>ISO 11898-1 ISO 11898-2 ISO 11898-3</td>
</tr>
<tr>
<td>Physical Layer</td>
<td></td>
<td>ISO 11898-1 ISO 11898-2 ISO 11898-3</td>
</tr>
</tbody>
</table>

The SAE J1939 standard provides definitions for five of the seven OSI model layers; the session and presentation layers are not specified by the SAE J1939 protocol. Table 2.4 shows the specification of the SAE J1939 to the OSI model.

Table 2.4. SAE J1939 specification of OSI Model Layers.

<table>
<thead>
<tr>
<th>OSI Model Layers</th>
<th>SAE J1939</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Layer</td>
<td>SAE J1939/71 SAE J1939/73</td>
</tr>
<tr>
<td>Presentation Layer</td>
<td></td>
</tr>
<tr>
<td>Session Layer</td>
<td></td>
</tr>
<tr>
<td>Transport Layer</td>
<td>SAE J1939 /21</td>
</tr>
<tr>
<td>Network Layer</td>
<td>SAE J1939 /31</td>
</tr>
<tr>
<td>Data Link Layer</td>
<td>SAE J1939 /21</td>
</tr>
<tr>
<td>Physical Layer</td>
<td>SAE J1939 /11 SAE J1939 /12</td>
</tr>
</tbody>
</table>
2.6.1 Layer 1 Physical Layer

The physical layer in the CAN bus is divided into physical medium dependent, physical medium attachment, and physical coding layer. Table 2.5 shows the CAN bus specifications of the physical layer.

Table 2.5. CAN specification of Physical Layer.

<table>
<thead>
<tr>
<th>OSI Model Layers</th>
<th>CAN bus layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Layer</td>
<td></td>
</tr>
<tr>
<td>Presentation Layer</td>
<td></td>
</tr>
<tr>
<td>Session Layer</td>
<td></td>
</tr>
<tr>
<td>Transport Layer</td>
<td></td>
</tr>
<tr>
<td>Network Layer</td>
<td></td>
</tr>
<tr>
<td>Data Link Layer</td>
<td></td>
</tr>
<tr>
<td>Physical Layer</td>
<td>Physical Coding Layer</td>
</tr>
<tr>
<td></td>
<td>Physical Media Attachment</td>
</tr>
<tr>
<td></td>
<td>Physical Media Dependent</td>
</tr>
</tbody>
</table>

The physical medium dependent sublayer has connectors and wires. The physical layer has defined electrical and mechanical standards. In terms of electrical standards, apart from standard voltage, current, and conductor specifications, there must be 120 ohm resistors at the terminal end for high-speed CAN communications. The most common connector for the CAN bus is a DB9 pin connector. Figure 2.4 shows a DB-9 connector with CAN signals labeled [62].
Table 2.6. DB9 Pin Assignments to CAN standard

<table>
<thead>
<tr>
<th>DB9 Connector Pin</th>
<th>CAN specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin 2</td>
<td>CAN Low</td>
</tr>
<tr>
<td>Pin 3</td>
<td>GROUND</td>
</tr>
<tr>
<td>Pin 7</td>
<td>CAN High</td>
</tr>
<tr>
<td>Pin 9</td>
<td>Power V+ (External Power Supply)</td>
</tr>
</tbody>
</table>

According to the specifications of ISO 11898-2, CAN high level has a voltage level of +2.75 volts to +4.5 volts, and CAN low level has a voltage level between +0.5 volts to +2.25 volts. The CAN bus has two logic levels for CAN high and CAN low levels. Generally, CAN high refers to dominant bits and CAN low refers to recessive bits. Figure 2.5 shows CAN high and CAN low logic levels with the differential voltage levels of the CAN bus.

Figure 2.5. Voltage level of CAN bus [63].
The physical media attachment sublayer has the transceiver section. The transceiver receives the TX and RX signal from the metacontroller and gives the output to the CAN bus. The transmission rate of the CAN bus can go from 125 Kbps to 1 Mbps [59]. The physical coding layer of CAN has bit encoding/decoding, bit synchronization and bit timings [59].

2.6.2 Layer 2 Data Link Layer

The J1939-21 CAN protocol specifies the data link layer in the OSI model. The data link layer is responsible for the transfer of messages between the physical layer of CAN and the CAN controller embedded inside a system [64]. The data link layer consists of CAN data frames with necessary data, error frames, and identifiers.

CAN has two message frame formats. One is the standard frame format, and the other is the extended frame format. The standard frame has 11-bit identifiers, and the extended frame has 29-bit identifiers. CAN has two specification versions: CAN 2.0 A and CAN 2.0 B. The first supports the standard frame message format, and the latter supports both. The ISO 11783 standard indicates CAN 2.0 B specifications and the extended CAN frame format.

2.6.2.1 Data Frame

The data frame is a common type of frame in a CAN bus. As its name implies, this frame is used for the transmission of data. A data frame can take a payload of up to 8 bytes. Figure 2.6 shows the CAN data frame format [65]. Identifiers are contained in the arbitration field of the data frame. Priorities of the messages are set in this field. The
control field has the information about the type of message being sent in the bus. A Cyclic Redundancy Check (CRC) is used to verify the correctness of the message bits. Upon the reception of the frame, Acknowledgment (ACK) field sends acknowledgment message.

![CAN data frame format](image)

Figure 2.6. CAN data frame format [65].

Every CAN frame has a Start of Frame (SOF) bit in the frame to indicate the start of the data transmission. The SOF is a single bit, which is dominant “0” bit. Whenever any node starts a transmission, the SOF bit is set to the dominant level to mark the start of transmission [66]. The transmission in the CAN bus ends with the recessive bit marking the bus as idle.

The SOF bit is followed by identifier bits. In the standard frame format, identifiers are made up of 11 bits which allow for $2^{11}$ message identifiers. In the extended CAN format, identifiers are composed of 29 bits. This format allows for $2^{29}$ different message identifiers. Identifiers set the priority of the messages in the CAN bus [67]. The lower the binary value in the identifier, the higher the message priority in the bus.

The ISO 11783 and ISO 11898-1 standards specify the message format which has a single protocol data unit. The 29-bit identifiers have several predefined fields: Priority,
Data Page, Protocol Data Unit, Protocol Data Unit Specific, Destination Address, and Source Address.

Figure 2.7 shows the 29-bit identifiers specification by the ISO 11783 standard. The CAN ID which is composed of 29 bits, i.e. extended frame format is further differentiated into Priority, Parameter Group Number, and Source Address. The Parameter Group Number is composed of a Reserved bit, Data Page, and Protocol Data Unit.

1. **Priority:**

   In CAN messages, the first three bits are for priority. The highest priority is given to the value 0 (000₂) and the lowest priority is given to the value 7 (111₂). In the CAN bus, the priority settings can be taken from the rule shown in Table 2.7 [68].
Table 2.7. Priority Setting in CAN bus.

<table>
<thead>
<tr>
<th>Priority(Decimal)</th>
<th>Bitwise Priority</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>111</td>
<td>Configuration Messages (Lowest Priority)</td>
</tr>
<tr>
<td>6</td>
<td>110</td>
<td>Status and Diagnostic message</td>
</tr>
<tr>
<td>5 or 4</td>
<td>101 or 110</td>
<td>High Speed Messages (Medium Priority)</td>
</tr>
<tr>
<td>3 or 2</td>
<td>011 or 010</td>
<td>Control Messages (High Priority)</td>
</tr>
<tr>
<td>1</td>
<td>001</td>
<td>Control messages of the closed loop control</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Very high priority)</td>
</tr>
<tr>
<td>0</td>
<td>000</td>
<td>Critical and Reserved functions( Highest Priority)</td>
</tr>
</tbody>
</table>

2. **Reserve and Data Page:**

The Reserve bit has 1 bit allocated and defines the bit that can be used in the future. The bit value is set to ‘0’ for transmitted messages. The data Page, commonly known as DP, gives the page number where a particular PGN belongs.

3. **Parameter Group Number:**

The Parameter Group Number commonly known as PGN, is embedded in the identifier of the CAN frame. A PGN includes the Parameter Group (PG), which defines the priority of the messages and specifies the group of specific parameters. Each PGN is identified by a unique parameter group. PGNs have total of 8672 different parameter groups per data page. A PGN in 29-bit extended frame format is shown in figure 2.8.

4. **PDU Format:**

The Protocol Data Unit (PDU) has the information about the messages being sent on the CAN bus. This unit defines the PGN assigned to that field. The PDU is divided into a Protocol Data Unit Format (PF) and a Protocol Data Unit Specific
Figure 2.8. PGN in 29 bits extended format. [69]

(PS). When more than 8 bytes of data need to be sent, CAN sends multi-packet messages. For data less than 8 bytes, CAN sends a single-frame message [69].

5. **PDU PS Format:**

The PS format also has 8 bits allocated, and its value depends on the PDU format.

The information in the PS field depends on whether the message is allocated to a specific address or to the Group Extension.

<table>
<thead>
<tr>
<th>PDU Format</th>
<th>PF</th>
<th>PS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDU 1</td>
<td>0-239</td>
<td>Destination Address</td>
</tr>
<tr>
<td>PDU 2</td>
<td>240-255</td>
<td>Group Extension</td>
</tr>
</tbody>
</table>

The value between 0-239 has a particular address on the CAN bus. Any node with an address other than the destination address (DA) ignores the message and does not take any action. If the value of the PF is between 0-239, then the PS is the destination address. When the value of PF is in the range 240-255, then the PS is the Group Extension [69].
6. **Destination Address:**

The Destination Address has a 1-byte field, and it specifies the address to which the message from the controller is being sent. The global destination address is given by 255 (Hex FF) in the network.

7. **Source Address:**

The Source Address field has 1 byte allocated in the CAN frame. This address in the CAN bus should be unique, and hence there is only one controller in the network with the given source address.

In a CAN frame, the data field has 8 bytes of space allocated. When 8 or fewer data bytes are required, the data field can use all 8 bytes of the data frame. This type of PGN is known as a single-packet PGN. The number of data bytes used is specified by setting the Data Length Code (DLC). The number of data bytes allocated for a particular PGN cannot be changed, and should have the same DLC specified by the manufacturer. When between 8 and 1,785 data bytes are used to express a PGN, then the CAN messages are sent in multiple CAN data frames. This type of PGN is known as multi-packet PGN [70].

The control field in the data frame includes Identifier Extension (IDE) bits, Reserve bits, and the DLC bits. In the standard frame format, the first bit is an IDE bit for the control field, and the second bit is a Reserve bit denoted by r0. The IDE bit is dominant in the standard and recessive in the extended format. In the extended format, there are two Reserved bits denoted by r0 and r1. The Reserve bits are set as recessive in both formats. The DLC field contains the total number bytes of data sent through the CAN bus. Figure 2.9 shows the control field of the data frame format. Figure 2.9 above shows
the control field of the data frame format.

2.6.2.2 Remote Frame

The remote frame in a CAN bus is used to request information or frames from the remote nodes. Remote frames are similar to data frames, with some fundamental differences. Identifiers in the remote frame format indicate the identifiers of the requested message. In a data frame, identifiers indicate the address of the sent messages. The data field bit in the remote frame format is always set as empty with 0-byte data [60].

2.6.2.3 Error Frame

When a node in the CAN bus detects an error, it generates and sends the error frame in the network. When an error frame is transmitted from a single node, other nodes also produce error frames. The transmitter then resends the original data to the respective location in the bus [60].
2.6.2.4 Overload Frame

While transmitting messages, nodes can become busy and unable to receive and transmit messages on time. In this situation, the node generates an overload frame to generate a delay between messages [64].

2.6.2.5 Messages Types

The CAN bus supports five different message types according to ISO 11783-2 standard which is listed below:

1. **Command**:

   The command message carries a command from the source to the destination. The destination can be specific or global depending upon the type of task. Command messages can include both PDU 1, PDU 2 and the Group Extension messages.

2. **Request**:

   The request message requests certain information about the specific PGN in the network. The destination address can be specific or global. The global destination address is given by $255 \ (FF)_{16}$ [70].

3. **Broadcast/Response**:

   The broadcast message is sent by the controller to all the nodes in the network. The receiving node, which is responsible for that particular task, accepts broadcast messages while other nodes can ignore them.
4. Acknowledgment:

The acknowledgment message is generated by the controller or the recipient upon receiving the CAN frame. This message specifies a handshake mechanism between the controller and the node [70].

Table 2.9. Acknowledgment Parameter Group Format [70].

<table>
<thead>
<tr>
<th>Parameter Group Name</th>
<th>Acknowledgment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission Rate</td>
<td>Per Requirement</td>
</tr>
<tr>
<td>Data Length</td>
<td>8 bytes</td>
</tr>
<tr>
<td>Data Page</td>
<td>0</td>
</tr>
<tr>
<td>PDU Format</td>
<td>239</td>
</tr>
<tr>
<td>PDU Specific</td>
<td>Destination Address</td>
</tr>
<tr>
<td>Default Priority</td>
<td>6</td>
</tr>
<tr>
<td>Parameter Group Number</td>
<td>59392(00E800_{16})</td>
</tr>
<tr>
<td>Data Bytes</td>
<td>Control Byte</td>
</tr>
<tr>
<td></td>
<td>Group Function Value</td>
</tr>
</tbody>
</table>

Table 2.10 shows the parameter group format for the positive acknowledgment message format. This format is received by the controller upon acceptance of the PGN messages. The positively acknowledged message is indicated by the value 0 in data byte 1.
Table 2.10. Positive Acknowledgment Parameter Group Data Byte Format [70].

<table>
<thead>
<tr>
<th>Data Bytes</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte 1</td>
<td>0; Positive acknowledgment</td>
</tr>
<tr>
<td>Byte 2</td>
<td>Group Function Value</td>
</tr>
<tr>
<td>Byte 3 to 4</td>
<td>Reserve for future by SAE</td>
</tr>
<tr>
<td>Byte 5</td>
<td>Acknowledged Address</td>
</tr>
<tr>
<td>Byte 6</td>
<td>LSB of requested PGN</td>
</tr>
<tr>
<td>Byte 7</td>
<td>Requested PGN’s second byte</td>
</tr>
<tr>
<td>Byte 8</td>
<td>MSB of requested PGN</td>
</tr>
<tr>
<td>Data Length</td>
<td>8 bytes</td>
</tr>
</tbody>
</table>

Table 2.11 shows the parameter group format for the negative acknowledgment message format. This format is received by the controller upon the negative acknowledgment of the PGN messages. The negatively acknowledged message is indicated by the value 1 in the data byte 1. This message also signifies that the PGN is recognized, but all parameters may not be available.

Table 2.11. Negative Acknowledgment Parameter Group Data Byte Format [70].

<table>
<thead>
<tr>
<th>Data Bytes</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte 1</td>
<td>1; Negative acknowledgment</td>
</tr>
<tr>
<td>Byte 2</td>
<td>Group Function Value</td>
</tr>
<tr>
<td>Byte 3 to 4</td>
<td>Reserve for future by SAE</td>
</tr>
<tr>
<td>Byte 5</td>
<td>Negatively Acknowledged Address</td>
</tr>
<tr>
<td>Byte 6 to 8</td>
<td>Requested PGN</td>
</tr>
</tbody>
</table>

Table 2.12 shows the parameter group format for the denied of access message format. This is received by the controller upon denial of access of the PGN message. The value in the data byte 1 should be 2 in order to indicate denial of access.
Table 2.12. Access Denied Parameter Group Data Byte Format [70].

<table>
<thead>
<tr>
<th>Data Bytes</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte 1</td>
<td>2; Access Denied</td>
</tr>
<tr>
<td>Byte 2</td>
<td>Group Function Value</td>
</tr>
<tr>
<td>Byte 3 to 4</td>
<td>Reserve for future by SAE</td>
</tr>
<tr>
<td>Byte 5</td>
<td>Access Denied Address</td>
</tr>
<tr>
<td>Byte 6 to 8</td>
<td>Requested PGN</td>
</tr>
</tbody>
</table>

Table 2.13 shows the parameter group format for the not responding message. This message is received by the controller when the node is busy and cannot respond at that moment. The controller must make the request again after a delay. The value in the data byte 1 should be 3 in order to indicate this message.

Table 2.13. Not Responding Parameter Group Data Byte Format [70].

<table>
<thead>
<tr>
<th>Data Bytes</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte 1</td>
<td>3; Cannot Respond (PGN supported but did not respond due to controller being busy)</td>
</tr>
<tr>
<td>Byte 2</td>
<td>Group Function Value</td>
</tr>
<tr>
<td>Byte 3 to 4</td>
<td>Reserve for future by SAE</td>
</tr>
<tr>
<td>Byte 5</td>
<td>Address Busy</td>
</tr>
<tr>
<td>Byte 6 to 8</td>
<td>Requested PGN</td>
</tr>
</tbody>
</table>

5. **Group Function:**

The Group Function messages are for specific group messages. Specific group messages include proprietary messages, multi-packet messages, and messages related to network management [69].
2.6.2.6 Message Packetization

A CAN frame has a data field allocated for 8 bytes of data. Data with more than 8 bytes in one packet must be correctly configured. The data need to be assembled in different frames. The first byte of the data field gives the sequence number of the packet, which can be assigned from 0 to 255. Packets are sent by ascending order of the sequence [69].

2.6.3 Layer 3 Network Layer

The SAE J1939/31 standard specifies the network layer. Network devices in the network should have unique addresses. There are 255 possible addresses [71].

Table 2.14. Address Management Messages Format [71].

<table>
<thead>
<tr>
<th>Address</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-253</td>
<td>Valid Source address for Controller Application</td>
</tr>
<tr>
<td>0-127 and 248-253</td>
<td>Controller Application with Preferred Addresses</td>
</tr>
<tr>
<td>128-247</td>
<td>Available for all Controller Application</td>
</tr>
<tr>
<td>254</td>
<td>Null address</td>
</tr>
<tr>
<td>255</td>
<td>Global Address</td>
</tr>
</tbody>
</table>

2.6.3.1 NAME and Address

All devices and controllers in the network must have unique NAMES. NAMES are required so that Controller Applications can transmit messages in the network. A NAME is a 64-bit identifier and is composed of different fields. Each field follows the standard assigned by the ISO [71]. The sending unit address is the Source Address, and the receiver address is known as the Destination Address. An address has 1-byte value, and its purpose
is to differentiate the CA in the network [72]. Figure 2.10 shows the industry format of NAME defined by the SAE J1939 standard.

<table>
<thead>
<tr>
<th>Arbitrary Address Capable</th>
<th>Industry Group</th>
<th>Vehicle System Instance</th>
<th>Vehicle System</th>
<th>Reserved</th>
<th>Function</th>
<th>Function Instance</th>
<th>ECU Instance</th>
<th>Manufacturer Code</th>
<th>Identity Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 bit</td>
<td>3 bit</td>
<td>4 bit</td>
<td>7 bit</td>
<td>1 bit</td>
<td>8 bit</td>
<td>5 bit</td>
<td>3 bit</td>
<td>11 bit</td>
<td>21 bit</td>
</tr>
</tbody>
</table>

Figure 2.10. NAME format in SAEJ1939 [71][72].

2.6.3.2 Network Management

Network management defines the procedures involved in handling the source address management, and dealing with network errors [72].

2.6.3.3 Address Management

Address Management includes procedures to request addresses and NAMEs in the network [71]. The Electronic Control Unit (ECU) in the network can ask to claim a particular address. This is the same unit that sends the “cannot claim address” message if unable to claim an address. The ECU also generates a command for its Controller Application (CA) to acquire a new address.
2.6.3.4 Address Claimed Request

The address claimed request message is sent by the CA in the network to request a unique NAME and address. This message is sent either to a particular address or to the global address [72].

2.6.3.5 Address Claimed Messages

The address claimed message request in the network can receive two answers from the particular Controller Application. Either the address is claimed, or the controller is not able to claim the address. When the CA can claim an address, it responds to the controller. The response is then sent back to the global address. Table 2.15 shows the address claimed message format defined by the SAE J1939/31 message format.
Table 2.15. Address Claimed Message Format [71].

<table>
<thead>
<tr>
<th>Parameter Group Name</th>
<th>Address Claimed Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission Rate</td>
<td>Per Requirement</td>
</tr>
<tr>
<td>Data Length</td>
<td>8 bytes</td>
</tr>
<tr>
<td>Data Page</td>
<td>0</td>
</tr>
<tr>
<td>PDU Format</td>
<td>238</td>
</tr>
<tr>
<td>PDU Specific</td>
<td>255 (Global Address)</td>
</tr>
<tr>
<td>Default Priority</td>
<td>6</td>
</tr>
<tr>
<td>Parameter Group Number</td>
<td>60928 (00EE0016)</td>
</tr>
<tr>
<td>Source Address</td>
<td>0 to 253</td>
</tr>
<tr>
<td>Data field</td>
<td></td>
</tr>
<tr>
<td>Byte 1</td>
<td>LSB of Identity Number</td>
</tr>
<tr>
<td>Byte 2</td>
<td>Second byte of Identity Number</td>
</tr>
<tr>
<td>Byte 3</td>
<td>Least significant 3 bits of Manufacturer Code Most significant 5 bits of Identity Number</td>
</tr>
<tr>
<td>Byte 4</td>
<td>Most significant 8 bits of Manufacturer Code</td>
</tr>
<tr>
<td>Byte 5</td>
<td>Function Instance</td>
</tr>
<tr>
<td>Byte 6</td>
<td>Function</td>
</tr>
<tr>
<td>Byte 7</td>
<td>Vehicle System</td>
</tr>
<tr>
<td>Byte 8</td>
<td>Arbitrary Address Capable Industry Group Vehicle System Instance</td>
</tr>
</tbody>
</table>

2.6.3.6 Cannot Claim Address

When the controller fails to claim an address, it either sends cannot claim address message or no self-configurable capability message. This message also has the same PGN, but has a different Source Address (SA). The SA in this message format is a null address which is 254.
2.6.3.7 Network Error Management

When an error occurs due to the controller failing to claim an address, or if a duplicate NAME conflict arises, then the controller follows the network error management procedure to detect and manage errors. The network error management procedure includes address claim requirements, initialization rules, and message sequences for initialization, address claim prioritization, and construction of address-to-NAME tables [72].

2.6.4 Layer 7 Application Layer

The SAE J1939/71 and SAE J1939/73 standards specify the application layer in the OSI model. This application includes specifications for all PGNs and SPNs in the network. This layer can broadcast or send PGNs with destination-specific messages at the specific transmit rate. The application layer specifies the software used, component identification in the network, and the parameter information.
CHAPTER 3 PROCEDURES

Chapter 3 describes the steps and procedures followed to achieve the objectives of this research. Section 3.1 lists the hardware and software configurations needed to set up a laboratory-scale microgrid test bed. Section 3.2 then explains the test bed benchmark design and setup procedures. Finally, Section 3.3 covers different energy management cases applied in this test bed.

3.1 Experimental Hardware and Software Procedures

The test bed designed in this study includes a generator, PV, a battery, a programmable load, and a real-time controller. This section discusses the manner of generating load profiles for various energy management cases. It then explains the method used to turn generators on and off remotely. The section ends with a discussion of a CAN bus set up between a real-time controller, a battery, and the inverter/charger system, followed by the procedures to integrate PV in the test bed.

3.1.1 Load Control and Profile Generation

A Cannon LB-60-30 Automatic Load bank (ALB) was used to set the load profile. This ALB, shown in Figure 3.1, is capable of providing a 120/240 volt single-phase 20 kW resistive load.
Figure 3.1 also depicts the control panel layout of the LB-60-30 load bank. The load bank had rate switches to set the load, and was programmed using Load Profile Manager software. The software can connect the load controller with any personal computer (PC) and schedule the load.

The load controller was connected to the PC through an Ethernet switch. The default Internet Protocol (IP) address of the explorer control board was 10.10.33.203, and the default IP address of the PC with load profile manager was 10.10.3.1, with subnet mask 255.255.255.0.

The load profile manager was provided with the desired load power in watts and seconds to schedule the load. Figure 3.2a shows the screen shot of the load profile manager and the load profile editor. Figure 3.2b shows watts rating and seconds in power usage events.
Figure 3.2. Screen Shot of Load Profile Manager with Load Profile Editor.

The load profile was generated from the 24-hour actual load profile of Nemiah Valley for May 9, 2012, [6]. The original load was scaled down to match the load and the generator capacity of the South Dakota State University’s microgrid lab. The procedure of scaling down load profiles allows researchers to study the real world load profiles easily, and is widely used in experimental studies [39]. Figures 3.3 and 3.4 show the load profile generation for the 3.5 kW and the 6 kW generator system with 13 and 7.3 scale factors, respectively. The ALB load had limited step resolution of 0.66 kW. Hence, approximate values were selected to match the original load profile.

Figure 3.3. Required Load Profile for the 3.5 kW generator system.
3.1.2 Generator Remote Start and Fuel Consumption Estimation

The Power Laboratory at South Dakota State University was connected to Yamaha EF6600DE 6kW and EF4000DE 3.5kW natural gas generators on the rooftop. The generator interface had a relay inside, and required a 5V signal to turn on the generator’s switch. Generators were remotely turned on and off using a 5V signal through the generator interface from the power laboratory. An OPAL RT 5600 real-time simulator had the algorithm to generate a control signal to turn the generator on or off.

Figure 3.5 shows the connection of generators with the load and a real-time controller. The automatic load bank was scheduled from the load profile manager. Generators were loaded from 0% to 80% of the rated load at different time settings. The load was scheduled to run for up to 14 minutes to observe the remote control feature.
The configuration in Figure 3.5 was again used for the calculation of the fuel consumption estimation equation for each generator. The load was scheduled at 0%, 25%, 50%, 75% and 100% of generators’ ratings. The fuel consumption values were recorded with each step increase in the load power. These values were interpolated for each case to obtain a polynomial equation of degree 2. The fuel estimation equation for the 3.5 kW generator is calculated as,

$$Y = 10^{-6}X^2 + 0.006X + 36.336$$  \hspace{1cm} (3.1)

Similarly, the fuel estimation equation calculated for the 6 kW generator is,

$$Y = 1.19^{-6}X^2 + 0.005661X + 40.89613$$  \hspace{1cm} (3.2)

In equations 3.1 and 3.2, $X$ is the loading of the generator in kW.
3.1.3 CAN communication interface between battery, inverter/charger, and real-time controller

Figure 3.6 shows the experimental setup of the microgrid with two 6 kW inverters/chargers, an 8 kWh Lithium Iron Phosphate battery, the OPAL RT 5600 real-time controller, and the grid.

![Figure 3.6. Battery inverter/charger connection with a real-time controller, a load and the grid with the CAN bus control.](image)

The grid was connected in the system to test CAN signals. A 240/208 V transformer was used to connect the 240 V AC bus and the utility grid. The ALB load was scheduled through the load controller. A control algorithm was built and implemented using the OPAL RT 5600 simulator. Voltage and current measurements of the components were taken through the OP8600 DAQ system in order to calculate the power.
3.1.3.1 Real time Controller and the CAN bus interface card

For the CAN bus interface, the OPAL RT 5600 simulator had a CAN-AC2-PCI CAN bus interface card provided by Softing. This card was embedded in the system and capable for use by RT-LAB in real-time simulations.

The CAN-AC2-PCI card had two channels, each of which was capable of communicating with 100 CAN nodes. Both the channels were configured as a 9-pin D-Sub CAN bus. RT-LAB used CAN controller blocks to control these two channels. These channels extended the CAN bus connection to other CAN devices in the network.

Figure 3.7. CAN-AC2-PCI card with two CAN channels.

3.1.3.2 Battery and Battery Management System

The battery used for the storage system in the microgrid test bed was a Lithium Iron Phosphate battery (LiFePo4). The battery bank was a 48 VDC 8.2 kWh lithium-ion battery. It had two modules, each with eight large format Lithium Iron Phosphate cells connected in series. LiFePo4 battery had a continuous discharge rate of 160 A and a maximum charging rate of 80 A [73].
Figure 3.8. Top view of the two modules of lithium iron battery [73].

The Lithium Iron Phosphate battery had the supervisor board set up containing fully a programmable controller board which had a Battery Management System (BMS). The BMS was contained in a supervisor control box enclosure, and data could be read using RS232 or CAN communication protocol.

Figure 3.9. Battery Management System Supervisor Control in Lithium Iron Phosphate Battery [73].

Figure 3.9 shows the supervisor control box enclosure with 24 Vdc input power and the communication protocol. The BMS was able to store the status of the battery current, state of charge, battery voltage, number of ampere hours remaining, individual
cell voltage, ambient temperatures on the system, and detected fault conditions.

The RS232 serial port was present on the supervisor control board to interface with an external device to monitor the BMS. The serial port communicated using 9600 baud rate, 8 bits, and no parity. Tera Term was the Graphical User Interface (GUI) used to interface the PC with the BMS to monitor the status of the BMS.

Apart from the RS232, a CAN port with baud rate up to 1 Mbps with CAN 2.0 specifications was used for communication purposes. The BMS was interfaced with the OPAL RT 5600 through a CAN card to monitor the SoC of the battery.

3.1.3.3 Real-time Controller and Inverter/Charger CAN interface

Two XW6048 battery inverters/chargers were used to facilitate the communication between the inverter/charger and a real-time simulator. The OPAL RT simulator was interfaced with each inverter/charger through the CAN bus. Two separate XW6048s were used: one as an inverter and the other as a charger. This method of using separate inverters/chargers avoids a delay that occurs when switching between inverting and charging functions. The destination node address for the charger was 0x02, and that of the inverter was 0x06. The CAN bus had a bit transfer rate of 250 kbps.

3.1.3.3.1 CAN Connectors

CAN specifications mandate that the cable used in a CAN network be less than or equal to 40 meters in length. Therefore, an 18-foot RJ45 male Ethernet Coaxial cable was selected for the communication between the real-time controller and the inverter/charger. The CAN card had a 9-pin D-Sub male connector pin, and the inverter/charger was
equipped with the RJ45 female connector. An RJ45 female to 9-pin D-Sub connector interface device was used for the connection.

![RJ45 to DB9 female adapter](image)

**Figure 3.10.** RJ45 to DB9 female adapter.

![RJ45 to DB9 connection](image)

**Figure 3.11.** RJ45 to DB9 connection between real-time controller and the inverter/charger.

### 3.1.3.3.2 Request for Address Claim Procedure

After the CAN devices had been connected to the physical network, the first procedure was to request for NAME and address. Two-way communication is only possible when every device involved in the communication acquires a unique NAME and address. For this purpose, an ISO Request for Address Claim message was used. The PGN for the ISO request was 59904($EA00_{16}$). In the CAN network, the address of the OPAL RT controller was 0x00. There were 255(0xFF) nodes in one inverter/charger, and
the address pool was in the range of 0x00-0xFF. The transmission rate of the data frame messages in the CAN network was set to 100 ms for all messages.

Figure 3.12. CAN ID formation for ISO Request PGN 59904 (EA00₁₆).

Figure 3.12 shows the making of CAN ID from the PGN provided. The destination address was taken as global address i.e 255 (0x00FF).

Figure 3.13. Screen Shot of ISO Request for Address Claim.

Figure 3.13 shows the ID maker and the message sending configuration in the RT-LAB model. The OpCANAc2 Send block sends the required payload to the destination address given by the Identifiers. DLC is the data length code, data is the required payload, and frequency is the transfer rate.
3.1.3.3 Address Claimed Procedure

After receiving a device’s NAME from the inverter, the controller must claim that NAME and address. The global destination address was taken for the address claiming procedure. The PGN for the ISO address claim was 60928, and the CAN Hex ID of the Address Claim message was 0x18EEFF00x. The controller was set to send this message every 100 ms.

<table>
<thead>
<tr>
<th>Parameter Group Number (PGN)</th>
<th>11 Bits Identifier</th>
<th>18 Bits Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>R</td>
<td>D</td>
</tr>
<tr>
<td>PDU Format (PF)</td>
<td>S</td>
<td>R</td>
</tr>
<tr>
<td>PF Cond.</td>
<td>PDU Specific (PS)</td>
<td>Destination Address (DA)</td>
</tr>
<tr>
<td>Binary Values</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Equivalent Hexadecimal Values</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>Decimal Values</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>CAN ID</td>
<td>0x18EEFF00x</td>
<td>0x18EEFF00x</td>
</tr>
</tbody>
</table>

Figure 3.14. CAN ID formation for Address Claim PGN.

Figure 3.15 shows the ID maker for address claim message

Figure 3.15. Screen Shot of ID formation for Address Claim Send.
3.1.3.3.4 Charging Command

After establishing two-way communication between the controller and the inverter/charger, the first task tested was to control the charger by sending ON and OFF commands. When the generators were turned ON in the system during the energy management test cases, the controller was able to send the charging command to turn the charger ON. The ON and OFF commands included their PGNs, which were single CAN frames with a DLC of 3. These PGNs were requested through an ISO request PGN, and the inverter sent information about the configuration.

The controller sent ON/OFF commands as per the requirements, and the charger positively acknowledged these commands. The next task was to change the charging rate of the charger. The charge rate PGN was used to increase or decrease the charge rate. This CAN message had two frames and required a counter value to match the counter on the charger. The counter information was requested through an ISO request. Its value incremented after each new frame of data was received, and could fall between 0 and 255. The frequency of this command was set to 100 ms.

In a multi-packet CAN inverter message, the first byte was used for sequence and frame bits. The first 3 bits were sequence numbers, and the remaining 5 bits were frame numbers. The sequence remained constant and the frame increased with every multi-frame message. The second byte was used for the number of payloads sent in a particular CAN message, and the counter was located at the fourth byte of the first frame.
3.1.3.5 Discharging Command

The load was connected to the inverter side of the inverter/charger. When no sources such as generators or a grid were present, the battery itself supplied the load. The battery power was enabled or disabled to the load through inverter control commands. The inverter enable/disable command was a single-frame CAN message with DLC equal to 2. This command was sent by the controller when the generator was turned OFF in the energy management test cases.

3.1.3.6 Inverter AC In/Out Voltage and Frequency Control

When the battery was operating alone in the test bed, voltage and frequency were monitored to operate at the standard level. The voltage was set at 240 V and frequency was set at 60 Hz. The controller was able to send the voltage and frequency reference points to the inverter. A multi-packet AC Out PGN with six frames of data was used to monitor and change the frequency and voltage of the inverter.

An AC In PGN was used to set the constant voltage and frequency at the inverter’s AC input side. The voltage was maintained at 240V and the frequency at 60 Hz. This PGN used a six-frame CAN message and required counter information.

3.1.3.4 CAN Communication between BMS and OPAL RT Simulator to monitor and store SoC

The RS232 serial communication protocol was used to monitor the status of the battery. The application used to monitor different parameters of the battery system was Tera Term, which is a free application that is readily available and can be downloaded
from the Internet. Tera Term is capable of displaying state of charge, battery voltage, current, temperature, and fault conditions.

The most critical task for the system was to monitor and store the SoC. Storing and manipulating this value in an RT-LAB model with Tera Term was not possible. Therefore, the CAN bus was used to store and track the SoC value of the lithium-ion battery.

The BMS communicated with the CAN card inside the OPAL RT 5600 simulator through the CAN bus. The CAN bus was able to store the SoC values. These values were decoded and stored in the RT-LAB model. The CAN bus used the standard 11-bit identifiers with a bit rate of 250 Kbps to transfer the data from the BMS. A complete set of messages was sent every 250 ms. The stored SoC value was given by a 0x02 identifier. The payload of the 0x02 identifier had values significant to the SoC and voltage values.

<table>
<thead>
<tr>
<th>Byte</th>
<th>No. of bits</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0 to 7</td>
<td>MSB (SoC)</td>
</tr>
<tr>
<td>1</td>
<td>0 to 7</td>
<td>LSB (SoC)</td>
</tr>
<tr>
<td>2</td>
<td>0 to 7</td>
<td>MSB (Voltage)</td>
</tr>
<tr>
<td>3</td>
<td>0 to 7</td>
<td>LSB (Voltage)</td>
</tr>
<tr>
<td>4</td>
<td>0 to 5</td>
<td>MSB (Average Cell Voltage)</td>
</tr>
<tr>
<td>5</td>
<td>4 to 8</td>
<td>LSB (Average Cell Voltage)</td>
</tr>
<tr>
<td>5</td>
<td>0 to 3</td>
<td>MSB (Maximum Cell Voltage)</td>
</tr>
<tr>
<td>6</td>
<td>2 to 7</td>
<td>LSB (Maximum Cell Voltage)</td>
</tr>
<tr>
<td>6</td>
<td>0 to 1</td>
<td>MSB (Minimum Cell Voltage)</td>
</tr>
<tr>
<td>7</td>
<td>0 to 7</td>
<td>LSB (Minimum Cell Voltage)</td>
</tr>
</tbody>
</table>

For the purpose of interfacing, a CAN cable was used according to OSI layer one specification. A 9-pin D-Sub connector cable was used for the communication between the OPAL RT controller and the BMS. The BMS side had a female connector, and the
controller side had a male connector.

The battery was connected to a load, an inverter/charger system, and a generator to facilitate the SoC monitoring and storing. For the charging process, the charger was set at 30 A and the generator was turned ON. The charger was then allowed to charge the battery for 30 minutes. In the discharging process, the load was scheduled at 1 kW and the generator was turned OFF. The battery was allowed to supply the load for 30 minutes.

3.1.4 PV Integration and Control

Twelve modules of 318 W SunPower and Future Growth solar panels were used to form a solar array of 3.8 kW. This array was integrated into the system, along with the SMA Sunny Boy 5000TL grid-tied PV inverter. Figure 3.16 shows the connection of the PV inverter with the battery inverter supplying the load. The PV inverter was connected to the output of the battery inverter, and the battery inverter’s firmware was updated to allow current to flow bi-directionally through the AC out port. The manufacturer provided the firmware update. According to IEEE 1547 standards [74], the grid-tied PV inverter has an adjustable delay or fixed delay of 5 minutes to connect to the AC bus formed by the battery inverter, which has standard voltage and frequency. When the PV inverter provided more power than the load could consume, the battery inverter allowed the excess power to charge the battery.
Figure 3.16 shows the test bed with a 6 kW Inverter/Charger, an 8 kWh battery system, a real-time controller, programmable loads, and a 3.8 kW PV. In this system, PV was integrated to charge the battery and supply the load. A 240/208 V transformer was used to interface between the PV inverter operating at 208 V and the battery inverter at 240 V. The battery inverter enable and disable commands were sent through the CAN bus from the real-time controller. The load was scheduled through the load controller to match the load profile.
Figure 3.17. Energy management flowchart of microgrid with 6 kW inverter/charger system, an 8 kWh battery system, a real-time controller, a programmable load, and a 3.8 kW PV.

Figure 3.17 shows the energy management algorithm designed for this case. Since there was no other source, an algorithm was designed to turn the system OFF when the battery SoC fell below 60%. A day load profile of Nemiah Valley for May 9, 2012 was scaled down to a 100-minute load profile to show the energy management for this case. The scale factor used was around 8.5.

3.2 Microgrid Test Bed Benchmark Design and Setup

Each control module and the microgrid components were connected to form a test bed. The CAN bus protocol was used as a communication medium between master and slave units. The test bed was fully automated and had reconfigurable capabilities. These capabilities are mainly possible because the CAN bus is a very flexible communication protocol that supports a multitude of devices. Accordingly, any component can be
replaced by another in the test bed without impacting the microgrid operation. Figure 3.18 shows the set up of the microgrid test bed. The testbed has a PV with an inverter, generators, a controllable AC load bank, inverter/charger and a battery system. These devices and this configuration are used to understand the dynamics of the microgrid. The OPAL RT 5600 was the real-time simulator used for the data acquisition system.

Figure 3.18. Laboratory scaled microgrid test bed at South Dakota State University.

This test bed was a laboratory scale low voltage 240V AC bus microgrid system. Voltage and current measurements were fed through the OPAL RT 8660 DAQ system to calculate the required power. OPAL RT 5600 communicated with inverters and also monitored the SoC through CAN bus. CAN charger and CAN inverter control signals were generated by the controller to control the operation of the inverter/charger.
3.3 Energy Management in the Microgrid Test Bed

An energy management algorithm was proposed in this study in order to understand the functioning of the microgrid. Two different cases were studied for the demonstration of coordination of sources and the load. The controller design in this test bed relied on CAN for communication, and the controller was capable of monitoring the SoC signal as well as providing current, voltage, and frequency reference values to the inverter.

3.3.1 Case I, Microgrid with a real-time controller, a programmable load, and 3.5 kW/6 kW generators

Figure 3.19 shows the experimental setup of the generator with a programmable load and a real-time controller. This was a test case to study the performance of the generator in the absence of a battery system. This case demonstrated the total fuel consumed by the generator. Yamaha EF6600DE 6 kW and EF4000DE 3.5 kW natural gas generators were used to calculate the fuel estimation. However, during experimentation a 10.4 kW generator had to be used since both the original generators were switched off for maintenance purposes.

This microgrid test bed had a maximum generator capacity of 6 kW; the battery used in this system had maximum charge rate of 80 A. For this reason, the 10.4 kW generator was not used for typical energy management operation. The OPAL RT 5600 simulator was the central controller to give ON/OFF commands to the generator. Voltage and current signals were fed to the OP8660, which calculated the generator and the load
power. The load was scheduled through the load controller to match the required load profile.

Figure 3.19. Microgrid with a real-time controller, a programmable load, and 3.5 kW/6 kW generators.

3.3.2 Case II, Microgrid with 6kW inverter/charger, a 8 kWh battery system, a real-time controller, a programmable load, and 3.5 kW/6 kW generators

Figure 3.20 shows the experimental setup of the microgrid with generators, a load, a battery, and a real-time controller. This was a test case to study the performance of the microgrid with the battery system integrated. An algorithm was designed to run the generators at approximately 80% of their capacity. The generators had a high efficiency when loaded to about 70-80% of their capacity. The controller maintained the battery
charging and discharging currents within the limits. This setup initially had 3.5 kW or 6 kW generators, two 6 kW inverter/charger systems, an 8 kWh lithium iron phosphate battery, and a real-time simulator. However, during experimentation a 10.4 kW generator was used to demonstrate microgrid operation. The SoC was monitored through the CAN SoC signal. The operation of the inverter/charger was controlled through CAN charger and CAN inverter signals. The ALB load was scheduled through the load controller to match the required load profile. All voltage and current signals were fed to the OP8660 DAQ system to calculate the power of each connected component. The generator ON/OFF signals were provided by the real-time simulator.

Figure 3.20. Microgrid with 6 kW inverter/charger system, an 8 kWh battery system, a real-time controller, a programmable load, and 3.5 kW/6 kW generators.
Figure 3.21 shows the energy management algorithm for this case. Whenever the load power was more than 80% of the generator's capacity, the generator was always turned ON to supply the load. The battery charger was disabled at this time. However, when the load power was less than 80% and the battery SoC was between 60% and 90%, the generator was turned OFF and the battery supplied the load. If the SoC was below 60%, the generator was turned ON and the charge rate was calculated. Thus, the generator could run at 80% of its capacity while supplying the load and charging the battery.

Figure 3.21. Energy Management Flowchart of Microgrid with 6 kW inverter/charger system, an 8 kWh battery system, a real-time controller, a programmable load, and 3.5 kW/6 kW generators.
CHAPTER 4 RESULTS AND ANALYSIS

This chapter presents the results of the experiments conducted in order to support the objectives of the research. Included is a discussion of the results relating to the generator control, SoC monitoring, and battery inverter control procedures. This chapter also explains the energy management results for different microgrid configurations.

4.1 Generator Control

This section provides experimental results of controlling generators from a remote location. Figures 4.1 and 4.2 show results when attempting to power (ON and OFF) the 3.5 kW and 6 kW generators, respectively.

(a) Scheduled load power, control on/off signal and the generator power.

(b) Spikes observed at 240 seconds and 600 seconds

Figure 4.1. 3.5 kW Generator Control.
The load was scheduled at 40%, 60%, and 80% of the generators’ ratings. The control signals were generated through the real-time controller. The controller generated a 5 V signal for the ON signal, and 0 V for the OFF signal. Generators were turned ON by sending the ON signal from the real-time controller. OFF control signals were transmitted at 240 and 600 seconds, and the duration of each OFF signal was 60 seconds. According to the manufacturer’s specifications, generators usually take between 5 to 30 seconds to turn ON. In figure 4.1a and 4.2a, around 5 seconds delay can be observed at the 300 and 660 seconds marks. This was when the generator received the ON signals.
Spikes were observed in Figures 4.1b and 4.2b when the generators receive the OFF signals. One possible reason for this observation could be that the load used in the experiment was not purely resistive, and contained inductive components. In such scenarios, when there is a sudden discontinuation of current to the inductor, inductor kick back voltage occurs. This voltage can be seen in the form of spikes in these figures.

4.2 SoC Monitoring

This section covers the SoC monitoring results of the battery. The measured SoC values were read from the CAN bus and the calculated SoC values were computed using equations mentioned earlier in the Chapter 2. The BMS of Lithium Iron Phosphate battery was able to communicate the battery’s SoC value to the CAN controller.

![Figure 4.3. SoC monitoring during charging the battery at 30 A charge rate.](image)

Figure 4.3 shows the monitoring of the SoC while the battery was charging. The measured initial SoC value was 62%. After 30 minutes, the final measured SoC value was 71%. However, the final calculated value of SoC at the end of 30 minutes was 70.5%.
This value was computed using Equation 2.6. Therefore, the SoC accuracy of the charging process was found to be 99.29%.

Similarly, Figure 4.4 shows the SoC monitoring while discharging the battery. The measured initial SoC value was 71% when the battery was allowed to discharge. The final SoC value measured at the end of 30 minutes was 65%. However, the final calculated SoC value at the end of 30 minutes was 64.2%. This value was calculated using Equation 2.7 for discharging. Therefore, the SoC accuracy for the discharging process was 98.7%.

4.3 Inverter/Charger Control

A CAN protocol-based controller was designed to control the operation of the battery inverter/charger. Several network hierarchies were considered for the design of the controller. The results from all required inverter control hierarchies are presented in this section. A CAN analyzer device was used to measured the time response of inverter/charger to the CAN bus. The CAN bus analyzer was a low cost CAN bus monitoring tool from Microchip.
4.3.1 Address Received for ISO Request for Address Claim

The first step in any ISO 11783 and ISO 11898 network is to request an address and a unique NAME in the network. The ISO request PGN 59904 was used to request the address information. This message was sent by the controller in the network. The data length code of this message was 3. Figure 4.5 shows the NAME the inverter received upon request, along with the corresponding decimal and hexadecimal values. The value 129 in the sixth byte stands for inverter/charger.

Figure 4.5. ISO Request for address claim data and NAME received from the inverter.

Figure 4.6 shows the industry format of NAME of the inverter provided by the manufacturer. The corresponding decimal values were analyzed per industry standard.
4.3.2 ISO Address Claim

The second procedure in an ISO 11783 and ISO 11898 network is to claim the given network address. The ISO address claim PGN was sent with the NAME to claim an address. For the user interface, the sixth-byte data value should be 128. A unique address for the controller with the sixth byte equal to 128 was claimed. The data length code of this message was 8, and the controller sent this message every 100 ms.

4.3.3 Charger Enable/Disable

Figure 4.7 shows the battery current and the battery power for charger enable and disable commands, respectively. In this case, the charge rate was set at 10 A. At 100 seconds, enable and disable commands were first sent with a 100-second duration. Then at 300 and 500 seconds, another set of enable and disable commands were sent with the same 100-second duration. The battery voltage and current signals were given to the OP8660 DAQ in order to calculate the power.
The controller sent enable/disable message frames every 100 ms. The CAN bus analyzer also showed that the time difference between the two consecutive frames (delta time) was 100 ms. The controller displayed an additional delay while ramping up between the disable and enable commands. Figure 4.8 indicates the amount of time the charger took to respond to the enable command; this was measured by the real-time controller and found to be 700 ms.

The two-way communication between the transmitting and receiving devices was acknowledged with the ISO Acknowledgment message. Figure 4.9 shows the packet received for the acknowledgment PGN. The control byte 0 states that PGN was positively acknowledged.
Table 4.1. Significance of Positive Acknowledgment Parameter Group Data Byte Format and Acknowledgment Message Received from Inverter [70].

<table>
<thead>
<tr>
<th>Data Bytes</th>
<th>Significance</th>
<th>Received Acknowledgment Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte 1</td>
<td>0; Positive acknowledgment</td>
<td>0</td>
</tr>
<tr>
<td>Byte 2</td>
<td>Group Function Value</td>
<td>255</td>
</tr>
<tr>
<td>Byte 3 to 4</td>
<td>Reserve for future by SAE</td>
<td>255</td>
</tr>
<tr>
<td>Byte 5</td>
<td>Acknowledged Address</td>
<td>255</td>
</tr>
<tr>
<td>Byte 6</td>
<td>LSB of requested PGN</td>
<td>0</td>
</tr>
<tr>
<td>Byte 7</td>
<td>Requested PGN’s second byte</td>
<td>***</td>
</tr>
<tr>
<td>Byte 8</td>
<td>MSB of requested PGN</td>
<td>1</td>
</tr>
<tr>
<td>Data Length</td>
<td>8 bytes</td>
<td>8 bytes</td>
</tr>
</tbody>
</table>

4.3.4 Charger Rate Control

Figure 4.10 charger’s response to the charge rate control command. The charge rate PGN was used to control the amount of current supplied to the charger. The controller
sent the charge rate starting at 5 A. After 100 seconds, the charge rate was increased by 1 A every 50 seconds until the peak charge rate was reached. The peak charge rate for this case was 10 A. The charge rate was then decreased by 1 A every 50 seconds until it reached 5 A once again.

Figure 4.10. Charger Rate Control.

The controller sent this message frame at 100-ms intervals. The CAN bus analyzer showed the delta time of this message to be 100 ms. The real-time controller recorded around 200 ms for each change in the charge rate. Figure 4.11 shows the time the charger took to respond to the charge rate change command.

Figure 4.11. Charger response to the charge rate command.
Figure 4.12 presents the acknowledgment packet received from the charger. The control byte 0 in the first byte indicates positive acknowledgment. The corresponding hexadecimal values are also indicated in Figure 4.12.

4.3.5 Inverter Enable/Disable Controller

Figure 4.13 shows the real-time control of the inverter enable and disable commands at different time intervals. When the inverter mode was enabled, the inverter supplied the power from the battery to the load; when disabled, the power supply to the load was stopped.
Figure 4.13. Inverter enable and disable Control.

The controller sent this message frame every 100 ms. Figure 4.14 shows the controller-recorded delta time of 100 ms between disable and enable commands.

Figure 4.14. Inverter response to enable command.

Figure 4.15 presents the acknowledgment message received from the inverter for an enable/disable command. The control byte 0 specifies that the message was positively acknowledged.
4.3.6 Inverter Output Voltage and Frequency Monitoring and Control

Figure 4.16 shows the monitoring of the inverter’s voltage and frequency. These values were measured by the CAN bus. It must be noted that measuring signals from a CAN bus may not provide accurate data; the CAN bus does not store values in decimal places, and the values are decoded to whole numbers.

A CAN bus typically stores constant values for fluctuating signals. Using the ISO request PGN, the inverter’s voltage and frequency information were requested in the network. The inverter positively acknowledged and sent the frequency and voltage values. These values were then decoded and stored using the Matlab/Simulink environment.
Figure 4.16. Voltage and frequency monitoring of the battery inverter/charger.

Figure 4.17 depicts the control of voltage and frequency. At 240 and 540 seconds, the controller sent six frames of AC Out PGN CAN messages requesting to change the voltage values. Similarly, using the same frame at 300 and 600 seconds, the controller sent a request to change the frequency value. These messages were positively acknowledged by the inverter.

Figure 4.17. Control of voltage and frequency battery inverter/charger through CAN controller.
Figure 4.18 indicates the amount of time the inverter took to respond to frequency and voltage change commands. The CAN bus analyzer recorded this response time to be 100 ms, while the real-time controller recorded approximately 800 ms.

![Figure 4.18. Response of inverter to voltage and frequency change command.](image)

4.3.7 Inverter AC Input Voltage and Frequency Monitoring and Control

Figure 4.19 shows the monitoring and control of the inverter’s AC input voltage and frequency. These values were measured by the CAN bus. The ISO request PGN was used to request voltage and frequency information from the inverter. This message was positively acknowledged, and the inverter sent the required information.
The AC In PGN was used to change the voltage and frequency values of the inverter’s input side. This message had six frames in total, and hence required counter information. Figure 4.20 shows the inverter’s response to voltage and frequency change commands. This curve was measured by the OP8660 DAQ system. The CAN bus analyzer showed the delta time between the CAN data frames to be 100 ms. The inverter took 100 ms to acknowledge this message, but the OPAL RT controller recorded the inverter’s response time as 600 ms.
4.4 PV Integration

A 3.8 kW PV array was integrated with the battery, a generator, and the load. The PV inverter used was a Sunny Boy 5000TL series grid-tied inverter. The generator was turned OFF in this system.

When the battery inverter was in the invert mode, this inverter acted as an AC voltage source. The battery inverter produced a controlled voltage and frequency AC bus. The PV inverter sensed whether the output voltage and frequency were in the normal range in order to be synchronized with this bus.

Figure 4.21. PV inverter integration to the battery inverter and the load system.

Figure 4.21 depicts the PV inverter integration with the battery and the load. This PV inverter shows a delay of 5 minutes in connecting to the grid with normal voltage and frequency. The positive power of the battery indicates that power is being provided to the load, while the negative power shows that the battery is in charging mode. When the PV inverter started to harvest power in the system, fluctuations were seen in the battery power.
The PV power generated power that was sufficient only to supply the load, and the battery switched between charging and discharging, depending upon the power provided by the PV inverter. After 500 seconds, the PV power began over-supplying the load and charging the battery. At that point there were no fluctuations.

Figure 4.22 shows the energy management using a 3.8 kW PV, an 8 kWh battery, a programmable load, and a real time controller. In this configuration, the PV power, battery power, and load power are measured with respect to time. As in figure 4.21, the negative battery power indicates discharging mode and positive battery power indicates charging mode.

Figure 4.22. Microgrid with 3.8 kW PV, an 8 kWh battery, a 6 kW inverter/charger, and a programmable load.

The experiment was conducted on November 16, 2016, from 11 am to 12:30 pm.

The peak PV power achieved for the day was 2.7 kW. For the first 10 minutes, the load
power was 2.65 kW. The PV power was sufficient to supply the load, and charged the battery with a negligible amount of current. Then for the next 20 minutes, between 0.167 to 0.5 hours, the load was scheduled to 3.8 kW. The PV power was unable to meet the load power in this case. Therefore, the battery together with the PV supplied the load. The load was then scheduled to 1.2 kW between 0.67 and 1 hour. The PV power was enough to provide the load, and the surplus power was used to charge the battery. The SoC value was regularly monitored and always maintained above 60%.

4.5 Case I, Microgrid with 3.5 kW/6 kW generators, a programmable load, and a real-time controller

In the first case, the microgrid operated with generators, a load, and a real-time controller. The load was scheduled by the load controller to run for 24 hours. For the energy management cases, 3.5 kW and 6 kW generators were replaced by a 10.4 kW generator. As mentioned earlier, the original generators were replaced because of scheduled servicing that required them to be permanently powered off; furthermore, it was not possible to replace the generators with new ones in such a short period. The fuel estimation, however, was calculated using the real 3.5 kW and 6 kW generators instead of the 10.4 kW generator. The battery used in this system had an absorption current limited to 80 A, which meant the 10.4 kW generator could not be used for real microgrid operation. Figure 4.23 shows the 24-hour result of operating the microgrid for the 3.5 kW generator system. This graph also indicates the generator and load power measured over time. The load was supplied by the generator for the entire period.
The estimated total volume of the natural gas for the 3.5 kW generator, calculated using Equation 3.1 was $865.21 ft^3$ \((24.5 m^3)\) for 24 hours. However, the measured value of the total volume of natural gas used by the generator for this case was $1730.70119 ft^3$ \((49.008 m^3)\) for 24 hours. The measured value was thus nearly double the estimated value, as the fuel consumption was measured using a 10.4 kW generator.

Figure 4.24 shows the experimental efficiency measured based on the power output of the generator. In this figure, the manufacturer’s efficiency is indicated by the dashed line. The specification data were taken from the report [39]. The efficiency for a 3.5 kW generator operating at 80% of its rating is 17%. The figure reveals that operating the generator at higher load resulted in an improved efficiency.
Figure 4.24. 3.5 kW generator’s efficiency measured with power output

Similarly, Figure 4.25 shows the 24-hour result of operating the 6 kW generator system with the load. In this figure, the generator and the load power are measured over time. The load was supplied by the generator for a full 24 hours.

Figure 4.25. Microgrid with a 6 kW generator, a programmable load, and a real-time controller.

The estimated natural gas consumption of the 6 kW generator from equation 3.2 was 1017.4155 ft$^3$ (27.81 m$^3$) for 24 hours. The measured value of the total volume of natural gas consumed by the generator was 2017.5269 ft$^3$ (57.13 m$^3$) for 24 hours. Once again, this value was nearly double because the 10.4 kW generator was used to measure
the fuel consumption.

Figure 4.26 shows the measured efficiency of 6 kW generator on power output.

The efficiency of the generator at 80% of its rating is 17%.

![Figure 4.26. 6 kW generator’s efficiency measured with power output](image)

4.6 Case II, Microgrid with 3.5 kW/6 kW generators, an 8 kWh battery, a programmable load, and a real-time controller

Figure 4.27 shows the energy management for a 3.5 kW generator system with two 6 kW inverters/chargers, a programmable load, an 8 kWh battery, and a real time controller. The generator was always turned ON when the load was greater than or equal to 2.8 kW (i.e. 80% of the generator capacity). When the load power was less than 80% and the battery SoC was between 60% and 90%, the generator was turned OFF in order to allow the battery to supply the load. The load power was approximately 2.8 kW between 8 to 10 hours and 19 to 24 hours. During these times, the generator was ON and the battery charger was OFF. Hence, SoC remained constant for this period.
Figure 4.27. Microgrid with a 3.5 kW generator, an 8 kWh battery, two 6 kW inverters/chargers, a programmable load, and a real-time controller.

For remainder of the time, the load was below 2.8 kW and the generator was OFF when the battery SoC was between 60% and 90%. When the battery SoC fell below 60%, the generator was again turned ON. The controller calculated the charge rate needed by the generator to charge the battery and run around at 80% capacity. The control signals were sent through the CAN signals. Similarly, in this case, the positive battery power indicated the discharging process and the negative battery power indicated the charging process. The total natural gas consumption measured for this case using 10.4 kW generator was 1448.6076 ft³ (41.02 m³) for 24 hours.

Figure 4.28 shows the measurement of the generator power, the load power, the battery power, and SoC values with respect to time. This is the energy management
scenario for a microgrid with 6 kW generator, two 6 kW inverters/chargers, a programmable load, an 8 kWh battery, and a real-time controller.

The generator was always turned ON when the load was greater than or equal to 4.8 kW (i.e. 80% of the generator capacity). As in the 3.5 kW generator case, when the load was less than 80% of the generator rating and the battery SoC was between 60% and 90%, the generator was OFF and the battery supplied the load. For SoC less than 60%, the controller sent the ON signal to the generator with a calculated charge rate as shown in Figure 3.21. This also allows the generator to run at 80% of its rating. The charge rate was dispatched through a CAN signal. In this case, the positive battery power indicated the discharging process and the negative battery power indicated the charging process.

Between 0 and 2 hours, the load power was nearly equal to 2.6 kW. The generator was OFF when the battery SoC reached 90%. The battery inverter enable command was sent
by the controller in order to select invert mode. The load power was greater than or equal to 4.8 kW between 8 to 10 and 18 to 24 hours. At all other times, the load power was less than 4.8 kW. The total natural gas consumed by the 10.4 kW generator in this case was 1705.698 $ft^3$ (48.3 $m^3$) for 24 hours.

The experimental results showed that the battery system integration lowered the fuel consumption volume. The fuel consumption using the 10.4 kW generator was decreased by 16.2% and 15.4% for the 3.5 kW and 6 kW generator systems, respectively.

Although the use of the 10.4 kW generator resulted in higher fuel consumption than what was estimated, the designed test bed demonstrates flexibility and reconfigurability. The use of different generators with varying capacities throughout the experimentation process validated multiple microgrid functionalities without affecting the overall system design.
CHAPTER 5 CONCLUSIONS

5.1 Summary

In today’s world, electricity has become one of the most basic necessities of life. Almost all aspects of modern life require electricity for continuous operation. However, there is still a significant proportion of the world’s population that is deprived of this fundamental amenity. In particular, people residing in remote parts of the world that are not connected to the main power grids lack a regular and dependable electricity supply. The expansion of main grids to remote areas is usually uneconomical and unfeasible, so these regions are routinely forced to find alternate sources of electricity. Generators are typically used in these places, but fossil fuel-powered generators can be costly and can have an adverse effect on the environment. A cheap and localized solution for electricity generation is needed that can make use of renewable resources; a microgrid can be a solution.

This research focused on the design and development of a microgrid test bed for experimental coordination of various microgrid components using the CAN protocol with a well-defined energy management examples tested. In this thesis, a laboratory-scale low-voltage microgrid benchmark was developed. Further tasks included the development of a control module to remotely control generators, real-time control of the battery inverter, interfacing microgrid components with a real-time digital controller, and validating energy management test cases. This study used a commercial off-the-shelf inverter control approach. A real-time controller controlled and monitored the inverter using the CAN bus protocol. The same protocol was also used to monitor the battery’s
SoC. In addition, the test bed featured an integrated 3.8 kW PV. Two different energy management cases were studied. The first case was without the battery system; in this case, the load was provided by the generator. In the second case, a battery was integrated into the system; it was found that the generator’s fuel consumption decreased in this scenario.

5.2 Conclusions

The main contribution of this work is the development of an automatic and economic microgrid test bed using commercial off-the-shelf components. The microgrid uses the flexible CAN bus protocol as a communication medium, and a real-time controller to control various operations of the microgrid. In this project, the CAN bus is used to send charge rate, charger enable/disable commands, inverter enable/disable commands, discharge rate, and voltage and frequency values to the inverter. Other contributions of this thesis include the remote control of generators, and testing of an energy management algorithm in a microgrid. Although, the experimental validation conducted using different generator, but the system design demonstrated the flexible nature. Any components can be replaced but the overall microgrid functionality remains the same.

5.3 Future Work

Future work should include upgrading the components of the microgrid. The power generation of the microgrid can be increased. The fuel consumption should be calculated using a real-time controller. Other communication protocols such as Modbus and Flexray should be explored.
REFERENCES


E. Technology, Introduction to solar cell or photovoltaic cells.


H. A. Gurganus, “Battery energy storage systems to mitigate the variability of photovoltaic power generation,” 2013.


