A FORMALIZED METHOD FOR STATE MACHINE SOFTWARE IMPLEMENTATION IN SMART MICROGRID CONTROL SYSTEMS

by

GREGORY KYLE TURNER

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2014

Acknowledgements

I wish to thank all of my committee members for their support during the development of this dissertation. To Dr. Davoudi and Dr. Kenarangui for their valuable service on the committee and as consultants. To Dr. Dillon for allowing me to come back and pick up the pieces of my studies after shelving my dissertation for a few years when work got in the way. To Dr. Wetz and Dr. Lee for their support and for bringing me into the Microgrid lab to perform research. And especially Dr. Wetz and the Pulsed Power and Energy Lab for hiring me on the spot when he heard I knew LabVIEW (your welcome for the new golf partner).

I would like to thank all my family for their love and support during the process of obtaining my doctorate degree. And especially my beautiful and very patient wife Karen. To my son Michael, thank you for your understanding during this process. If I had to choose one thing to do for the rest of my life, it would be to go out in the backyard and play catch with you. To the newest member of the family, Wolfgang, I would like to say rolf, rowr and grrr.

March 21, 2014

Abstract

A FORMALIZED METHOD FOR STATE MACHINE SOFTWARE

IMPLEMENTATION IN SMART MICROGRID

CONTROL SYSTEMS

Gregory Kyle Turner, Ph.D.

The University of Texas at Arlington, 2014

Supervising Professor: David Wetz

The introduction section begins with a discussion of the electric power grid in general and a specific definition for Smart Grids and Microgrids. These two different but complimentary technologies combine to form the overall Smart Microgrid concept. The control theory for Smart Microgrid applications is broken down into a three-tiered hierarchical scheme in order to simplify the control systems used at each level of the hierarchy. Some background information is presented about current Smart Microgrid projects.

A review of recent literature focuses on three areas of interest. The review of control theory focuses mainly on the highest abstract level of the Smart Microgrid control hierarchy. Software design literature applies formalized design techniques directly to Smart Microgrid applications. Some specific applications are reviewed in order to help in the consideration of specific use cases for study.

The hardware design chapter contains specific details of the complete UTA Microgrid test platform.

A proposed method for implementing Smart Microgrid application software is presented. State machine design is discussed as an abstract concept leading to the

formalized description of an embedded machine for a Microgrid. Once the formalized state machine is designed, guidelines for implementing the machine into LabVIEW software are presented.

A specific use case of a Smart Microgrid is defined and taken through the formal process proposed in the previous chapter. This software application is tested on the UTA Microgrid and the performance of the application is presented.

Finally, some ideas for future work are presented.

Table of Contents

List of Illustrations

List of Tables

Acronyms

- DoE – Department of Energy
- DG – Distributed Generation
- DER – Distributed Energy Resource
- NETL – National Energy Technology Laboratory
- TLA – Three Letter Acronym
- UTA – University of Texas at Arlington
- RTOS – Real Time Operating System
- GPOS – General Purpose Operating System
- DAQ – Data Acquisition
- RAM – Random Access Memory

Chapter 1

Introduction

For well over 100 years, the power supplied to the electric power grid has been created by large remote power stations [1]. These stations use a variety of resources including coal, hydro, natural gas, wind, solar, and nuclear energy. Generating stations produce power at a voltage that is transformed to very high voltages (on the order of 500kV) in order to maximize transmission efficiency over long distances. At a utility substation inside an urban area, the voltage is transformed back down for distribution. Finally, an additional transformer is used to convert the power to a voltage used by the equipment and appliances at a customer location. For residential customers in the United States, this is 120V RMS for most appliances. Homes typically have two of these circuits in order to produce 220V RMS for clothes dryers and air conditioners.

Population growth and age have caused the transmission system to become fragile. The addition of electronic devices have significantly increased demand per customer in the system. Newer appliances and electronic devices such as computers, high definition televisions and the chargers used for portable devices are more sensitive to variations on the voltage level that the system provides. All of these factors negatively affect the reliability of the electric power system overall [2].

It may be possible to extend the life of the system simply by adding new transmission and distribution lines. This is an expensive solution both in the actual monetary cost and the bureaucratic cost of installing new power lines through areas that do not want them. What is really needed is an approach that significantly increases the efficiency of the overall electrical transmission and distribution system. An increased efficiency that promotes reliability while reducing greenhouse emissions is a primary goal of the Smart Microgrid concept.

1.1 Definition of Smart Grid

In what may be considered the first commercial electric lighting and power system in 1882, the Pearl Street station in lower Manhattan, Thomas Edison used an electromechanical system of magnets to illuminate different lights to indicate the condition of the power grid. The lights indicated a rising or sagging voltage in the system which could be offset by an attendant at the station by adjusting the field strength in the station generators. As early as 1889, electric utility companies began using devices to meter the amount of electricity consumed by a customer. Except for improvements in the accuracy of electric meters they have more or less remained the same from the customer perspective since they were first introduced.

As utility companies pooled resources in order to improve the efficiency of distribution and transmission networks, the modern electrical grid was formed. The continental U.S. grid has grown to form a nationwide interconnection of three national grids, the eastern Interconnect, western interconnect, and the Texas Interconnect (see Figure 1-1) [3]. Although the figure shows only the continental United States, the interconnection cover a good part of Canada, Mexico and Alaska.

Although many technological developments have helped the grid's effectiveness and operability in the middle part of the twentieth century, the grid has not kept up with the pace of modernization when compared to the level of development in computer and communication technology. Today, the energy producers simply feed energy in a constant flow on the grid. Unused electricity is wasted since it must always be available on demand and no consideration is given for the cost because it is always the same no matter the time of day.

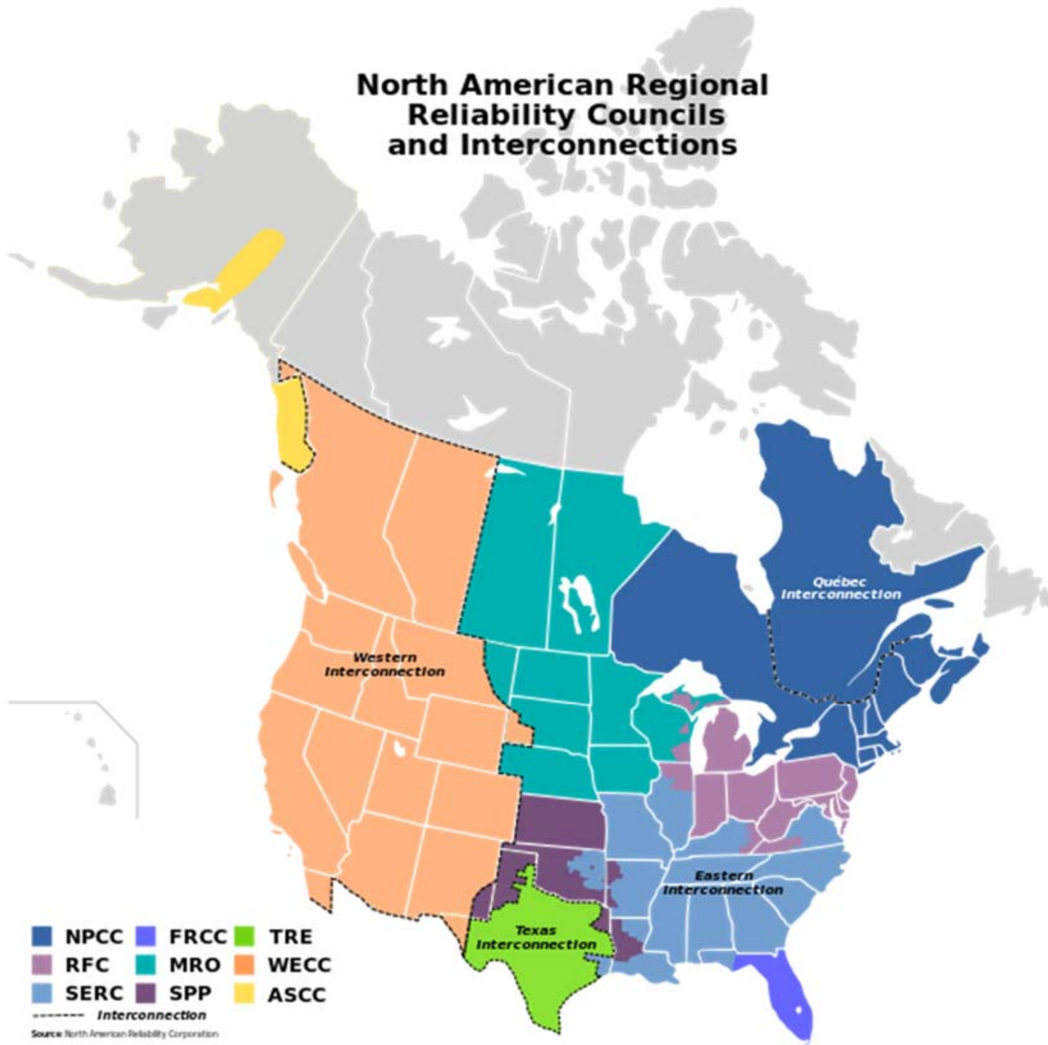


Figure 1-1 - United States Electrical Grid Interconnection

The basic Smart Grid concept is to add monitoring and communication technology to the electricity grid in order to maximize the transmission and distribution capacity of the system. Utility companies can use the information gathered from the system to make decisions about how to move power through the system as efficiently as possible. The information can also be provided to the end customer to allow them to make decisions about their energy usage. Incentives can be provided to the customer to make them more likely to use energy in a way that promotes stability in the system. Today the implementation of Smart Grid ideas has stopped at the introduction of smart meters that gather power usage information at an end customer location. But this is the perfect introduction for the Smart Grid to take advantage of new technologies that are beginning to emerge, such as automated homes, plug-in hybrid electric vehicles, and all forms of distributed renewable generation [4].

1.2 Definition of Microgrid

A Microgrid may be defined as a localized electrical grid that contains sources of electric power and the loads that use that power [5]. The simplest form of a Microgrid would exist in an 'islanded' mode and have no connection to the greater utility grid. Since a typical Microgrid would not exist near a fuel source, the primary sources in a Microgrid would be renewable ones like solar and wind. A more sophisticated Microgrid that requires loads to be powered 24 hours a day, would need energy storage elements or a backup source. Backup power in an islanded Microgrid would have be in the form of a generator using an internal combustion engine or a fuel cell. Any backup power would need fuel storage in some capacity either gasoline, diesel, or hydrogen. Backup power in a Microgrid could also come from a utility grid, thereby changing the Microgrid from islanded mode to 'connected' mode. A very sophisticated Microgrid would combine all of these elements into a system that is capable of selling extra power back to the utility. This requires the cooperation of the utility of course and is usually only available in a location with strict regulations on how this energy exchange is managed.

A Microgrid connected to a reliable utility system would not need energy storage in the form of local batteries if it can effectively use the utility grid as its storage for excess power. Selling excess power to the utility grid can also be a great benefit to the utility grid. Peak usage in a utility grid often occurs during the hottest part of a day when air conditioners are at their peak usage. This is also when solar panels are near their maximum power output. Offsetting peak utility demand with distributed solar power can be an effective way to ease the stress on an overburdened utility system.

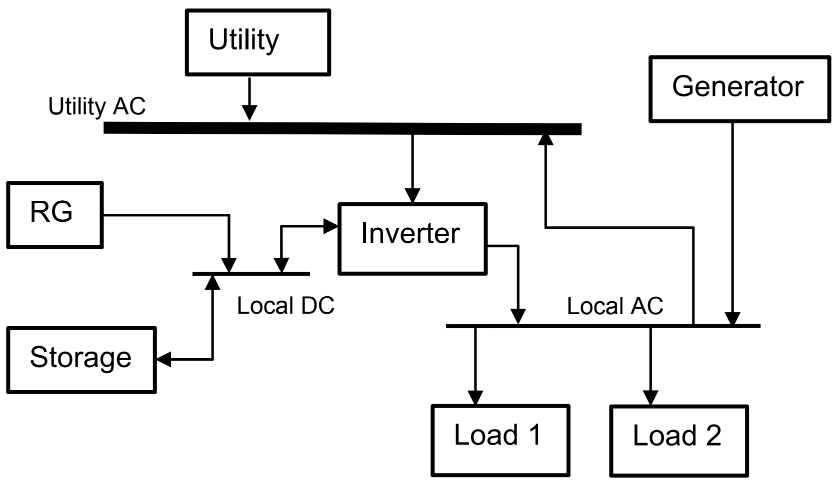


Figure 1-2 Microgrid Block Diagram

The left side of the diagram shows how renewable generation (RG) sources and energy storage elements can simultaneously provide the local DC power bus the supplies energy to an inverter which generates local AC power for the Microgrid loads. Multiple loads can be connected to the AC bus at any time. In this block diagram, the inverter is also shown as being supplied from an optional external bus (denoted by the thick line). This external bus may be supplied from a utility source or from a local backup generator. On the right side of the diagram the AC bus has a connection back to the utility/backup bus in order to supply power from the Microgrid back to the utility if possible.

A Microgrid capable of interacting with the utility grid in a cooperative manor requires the kind of monitoring and control architecture defined in the Smart Grid concept. Therefore, the term ‘Smart Microgrid’ is introduced to refer to a Microgrid that can not only generate power for its own load, but also integrate into the greater electrical power grid for the benefit of the overall power system.

1.3 Smart Microgrid Control Theory

So the Smart Grid concept is technically different from the definition of a Microgrid. However, the ideas are actually complimentary and it is easy to combine the two into the fundamental idea of a Smart Microgrid. A Smart Microgrid is a sophisticated grid-tied Microgrid that uses fundamental Smart Grid sensor and communication technology to support the overall power grid.

There are many different control strategies needed to implement a Smart Microgrid. It is helpful to think about them in terms of a three tiered hierarchy [6]. Figure 1-3 shows this hierarchy in a block diagram. At the base of the hierarchy, the primary control strategy is focused on voltage and frequency stability of the device which supplies power to the AC bus of the Microgrid from the local sources. Typically this is an inverter which takes in DC power and converts it to the AC power needed by the local loads. The control system operates the power electronics that provide the switching needed to perform the DC to AC conversion. In primary control this stability is for the benefit of the local loads only without regard for the requirements or characteristics of any external utility connection. In addition, primary control must manage the local sources and storage elements for the benefit of the local grid.

The secondary control level is concerned again with voltage and frequency stability of the power at the local grid. However, at the secondary level the deviations may be complicated by the primary control system. The secondary control system will monitor the local sources of generation, the local loads on the Microgrid, as well as the inverter system in an effort to control voltage and frequency deviation at the local level.

The tertiary (third) level of control in the Microgrid encompasses the strategies of the first two levels and also incorporates strategies for tying the Microgrid to a greater grid system. At this level, the control strategies become more abstract. With the first two levels of control making sure that the power is clean, the third level may be concerned with implementing use cases that promote increased reliability and efficiency in the overall utility grid. Tertiary control can be implemented in different ways depending on higher level goals. Any electrical grid will have loads that can be categorized in terms of how critical it is that they are operational.

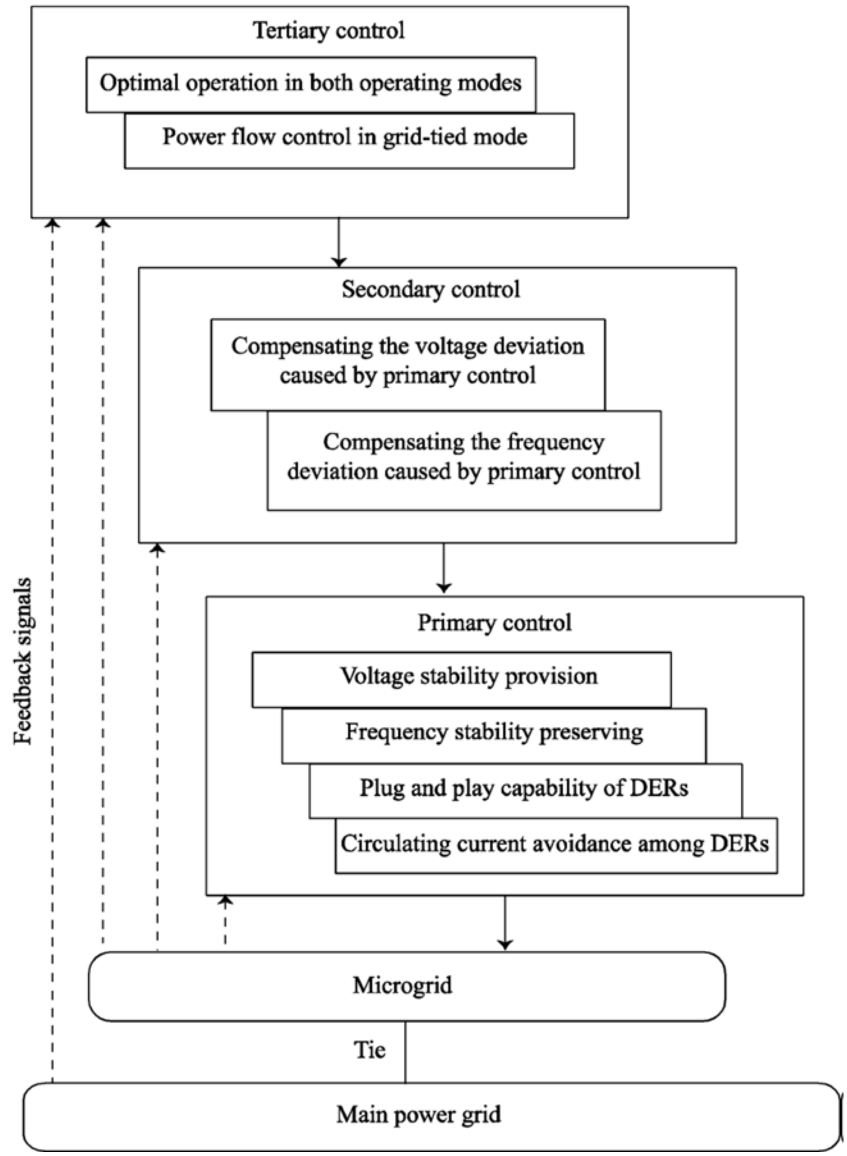


Figure 1-3 Microgrid Control Strategy Hierarchy [6]

1.4 Background

Several legislative events have accelerated Smart Grid/Microgrid research. The Energy Policy Act of 2005 made IEEE 1547 the standard governing the interconnection of distributed electrical generation sources [7]. This essentially established the standard for how any Microgrid must be connected to the greater utility grid. In this same act, the

Department of Energy (DoE) National Energy Technology Laboratory (NETL) was charged with establishing the safety and practices that must be used by developers and operators of Smart Grids. In 2007 the Energy Independence and Security Act was enacted to (among other things) promote energy independence and security, increase production of renewable energy and increase efficiency in all energy related systems [8]. The 2009 American Recovery and Investment Act apportioned $4.5 billion to the DoE specifically to implement Smart Grid programs. A breakdown of major expenditures from this legislation is shown in Table 1-1.

Table 1-1 - Federal Recovery Act Funding in Smart Grid programs [9]

Major Smart Grid Program Activities	Total Obligations ($Million)
Smart Grid Investment Grant	$3,425
Smart Grid Regional and Energy Storage Demonstration Projects	$685
Workforce Training and Development Program	$100
Interconnection Transmission Planning	$80
State Assistance for Recovery Act Related Electricity Policies	$49
Enhancing State Energy Assurance	$44
Interoperability Standards and Framework	$12
Enhancing Local Government Energy Assurance	$8

Overall the programs have a large number of success stories [10]. The vast majority of this direct investment involves the installation of automation equipment in distribution grids and the installation of smart meters at customer locations. When usage data is combined with customer involvement to allow time-based rate decisions to be made, peak demand has been shown to be reduced by as much as 37%.

However, most of these direct investments have not involved the implementation of interconnected Microgrids with distributed generation. In general, Microgrids have enjoyed a general deployment. And many Microgrids are still being studied as part of research and development projects.

There are several Microgrid projects that are part of academic institutions.

1.4.1 CERTS Microgrid Lab

The Consortium for Electric Reliability Technology Solutions (CERTS) has developed a test bed Microgrid project with three goals in mind [11]. Two of the goals involve electrical safety within a Microgrid and stability under islanded conditions. The third goal is particularly applicable to the subject of this dissertation. It is to discover effective methods to automatically transition Microgrids from isolated (islanded) to grid-connected operational modes. This goal is effectively the same as the tertiary control strategy described in the introduction.

The CERTS Microgrid test bed uses a peer-to-peer concept where there is no main controller. Individual Microgrids must be able to behave autonomously in order to provide power for their local loads. Each Microgrid must also be able to behave in an environment where other neighboring Microgrids may interconnect in order to share power for the good of the overall utility grid [12].

1.4.2 Santa Clara University (SCU) Microgrid

SCU has an ongoing Smart Microgrid project (see Figure 1-4) with the intent to make the entire campus completely sustainable with zero carbon footprint. All data from sources, distribution, and consumption are combined to maximize the efficiency of the system overall [13].

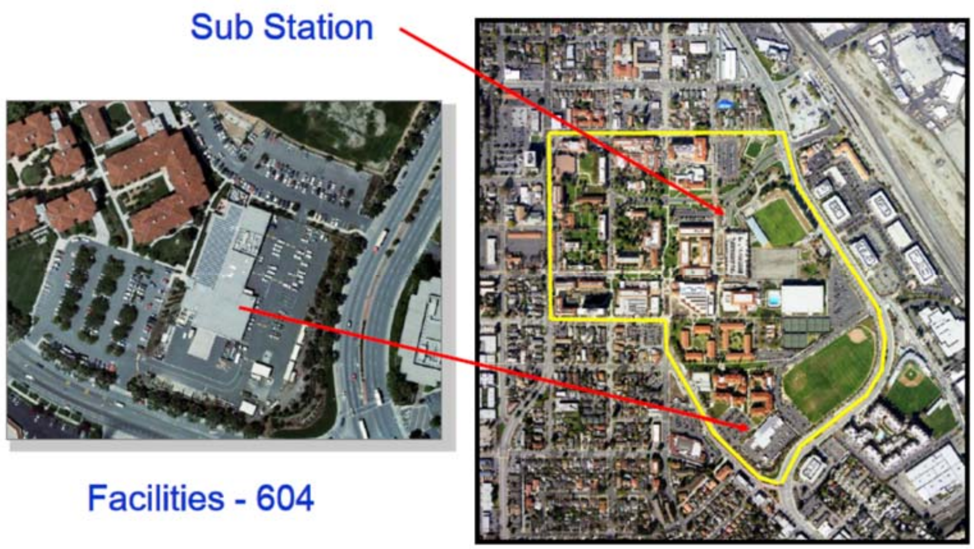


Figure 1-4 - Santa Clara University Campus Microgrid

Power is sourced from a variety of renewable sources including solar, wind and geothermal. Usage data is delivered in real time and the system also performs calculations concerning carbon emissions. This essentially provides detail about how the Microgrid is offsetting usage from utility sources to demonstrate the environmental performance of the Microgrid. The system can withstand major power outages at the utility level and can keep a large segment of the campus operational for an extended period of time.

The project has been rolled out in phases. The first phase included a comprehensive installation of smart meters in strategic locations in the campus. Subsequent phases integrated existing renewable sources with new sources that had been located in critical areas to meet localized needs. At the completion of project, the estimated reduction in energy consumption should be around 50% with an overall cost savings for the university of about 20%

The Santa Clara Microgrid project is a collaboration between SCU, Cisco and Serious Energy which is a company founded by former students who completed in the 2007 Solar Decathlon for the university.

1.4.3 Illinois Institute of Technology Microgrid

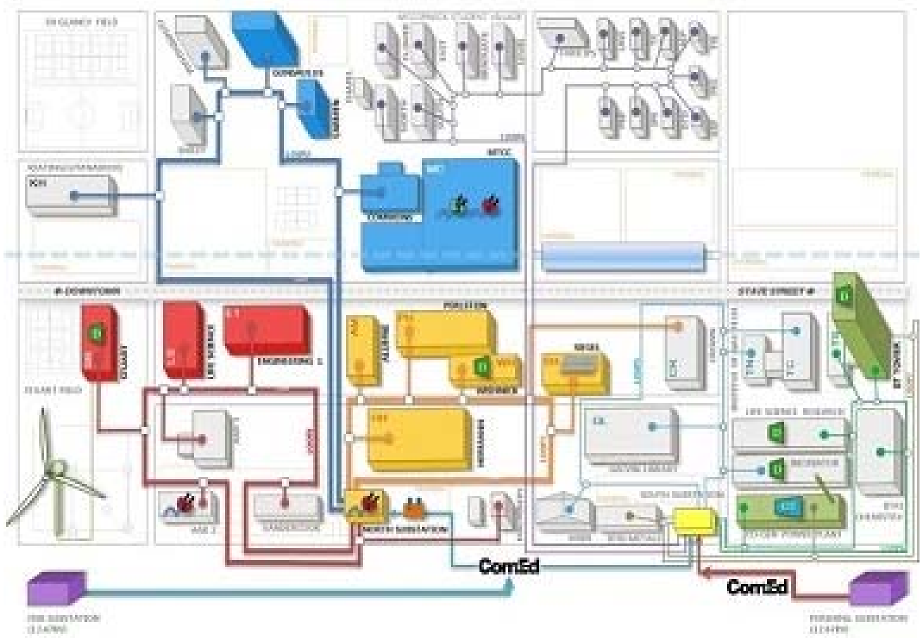


Figure 1-5 - Illinois Institute of Technology Microgrid

The Illinois Institute of Technology has installed a functional Microgrid in major parts of the campus [14]. The grid has been designed with a unique distribution network where power flows in a loop through several buildings. These loops are in place of traditional radial type circuit networks. Switches located in the loop circuit detect and react to faults. These switches are able to isolate specific buildings in the circuit while maintaining power to other building in the loop.

Each building has an individual controller that communicates with a central master controller. This communication network provides every building with real-time electricity pricing information and a load shedding priority scheme.

Sustainable energy sources are installed for the purpose of supporting the overall system. These sources include wind and solar sources. Flow battery systems are used to integrate electric vehicle charging stations

1.4.4 National Energy Technology Laboratory

In 2010, the University of Texas at Arlington received an earmark managed by the National Energy Technology Laboratory to fund and develop a Smart Microgrid testbed [15]. From the sponsoring organization Energy Delivery Technologies Division, the testbed proposal is intended to explore the development and integration of Microgrid technologies into a future electrical grid infrastructure. The project was funded as a two year project.

Today the UTA Microgrid lab contains three independent Microgrids capable of islanded mode operation. A unique distribution architecture allows each grid to cooperative support any of the other grids. Control of the grid has been designed as a peer-to-peer cooperative strategy with an end goal of supporting any imagined use case scenario.

1.5 Synopsis of Chapters

This dissertation begins with a review of recent literature in Chapter 2. This consists mostly of recent papers from IEEE journals related to the field of Smart Grid and Microgrid integration.

In Chapter 3, a full description of the UTA Microgrid hardware is given.

Chapter 4 will present the proposed control system design that will be implemented for the Microgrid. This will include a description of the software and how it was implemented in LabVIEW.

In Chapter 5, experimental data from the control system software is presented. The Microgrid will be operated in a defined use case and the dynamic performance of the Microgrid will be shown in terms of the power flow in the system as it reacts to changes in the system.

Chapter 6 will discuss the outcome of experiments from the previous chapter and present topics that are in need of further research.

Chapter 2

Literature Review

A literature review has been performed on IEEE journals and conference articles through the IEEE Xplore Digital Library [16]. A secondary and perhaps more valuable resource has been the technical book library at Safari Books Online [17]. The review has been broken into three subject matter areas of interest as they relate to Smart Grids and Microgrids in general. These are control theory, software design and applications.

2.1 Control Theory

Control theory in the field area of Microgrids is focused in two general areas. The first is a direct control in the manipulation of power electronics to create household AC power from various sources of power in the Microgrid. The second is the more abstract control dealing with power flow and system stability of the power grid at large.

In [18], researchers are focused on power quality factors in a system that can be both islanded and grid-tied. The system contains a mixture of renewable sources. The control is focused on a direct manipulation of the power electronics of an AC inverter for the purpose of maintaining proper voltage and frequency stability in the system.

Optimal policies as described in [19] are a valuable way to think about key parameters in the overall system level performance. Defining a key control parameter as a cost function, the parameter can be either maximized or minimized to serve the needs of the system. In this case, battery performance is used as the key parameter to control the performance of the Microgrid in the presence of energy price fluctuations.

In [20], researchers present control methods for the direct control of power electronics that convert power into usable household AC. Of particular importance in this article is the manipulation of the Microgrid power to match the characteristics of a synchronous generator. This is a very important function in a system that intends to sell back power to the utility grid and therefore must stay in synch with that system.

Another optimal control method is described in [21]. Here the focus is on the creation of a scheduling parameter that is then optimized to create an efficient demand response system in home energy management. This system relies heavily on the information that would be received from a typical smart meter to make decisions that affect optimal system performance.

In [22], researchers design a control scheme for a Microgrid in which multiple inverters/converters act cooperatively in the Microgrid. The focus here is on completely a system that has the necessary redundancy for a system with an increased need for reliability.

Another paper that is focused on the low level of control of power electronics for a Microgrid is [23]. This paper focuses on the PWM switching control characteristics of a sophisticated Microgrid containing multiple sources and energy storage options.

The focus of [24] is a system very similar to the use case of this dissertation. The system uses de-centralized control schemes in a network of interconnected Microgrids that act in a cooperative manner for the benefit of each Microgrid and the overall system as a whole. Each Microgrid is capable of requesting power from a neighboring grid and offering excess power to any connected grid.

The final resource listed in this section is similar to the one in the last paragraph. In [25], a distributed control system uses information about local generation and generation of close neighbors to make control decisions about how the overall grid should cooperate. The distributed nature of the scheme eliminates the need for a central controller and reduces the need for very complex communication infrastructure.

2.2 Software Design

Sources for software design focused on those that combined embedded systems and Microgrid applications. In addition, resources that included material regarding state machine design and implementation were very valuable.

In [26], researchers describe a layered service oriented architecture that provides open interfaces toward the goal of creating plug and play hardware and software components for Microgrid design. Software components are designed to be scalable for any size or configuration of Microgrid use case that can be imagined.

Building functional software blocks that meet the automation requirements of the IEC 61850 standard is the primary goal of [27]. A simulation of a voltage control system at the electrical substation level is presented.

Another resource focusing on the software framework to be in a Smart Grid system is [28]. Here researchers present steps for chaining the simulation tools that are used to perform basic simulations in various Smart Grid components. Integrating the data from these various components allows a more sophisticated simulation to provide a better solution to the control system.

In [29], researchers present a method for modeling and implementing embedded control systems from nested state machines. A formalized method of defining the state machine is presented as well as some guidelines for the implementation in software. There is no specific application made here to Smart Grids or Microgrids.

The researchers at the Pacific Northwest National Lab [30] have introduced a concept called GridOPTICS as a software architecture for the design of plug and play Smart Grid systems. Based on actual use cases developed in the lab's Future Power Grid Initiative, advanced software technologies are described as solutions for Smart Grid management and control.

In the conference paper [31], researchers describe a hybrid control system for a Microgrid based on the Multiple Input Mulitple Output (MIMO) control problem. A systematic approach is used to design a supervisory control system that can handle this type of system. This abstract approach is very suited to the type of sophisticated Smart Microgrid imagined as having a variety of power sources and a variety of loads.

A high level abstract approach to the Smart Microgrid concept is provided in [32]. This paper describes the application of formalize methods in the design of software for Smart Microgrids. Comparisons are made to existing complex systems that deal in a mixture of engineering disciplines. A case study involving how formal methods can be applied to Smart Microgrids is presented.

The resource [33] may be the closest to the practical method presented in this dissertation. It is the only conference paper to describe a state machine implementation in the LabVIEW programming language. Although the application is not in the field of power grids, the methods do adapt to the Smart Microgrid concept.

A pilot project focused on distribution system automation is the subject of [34]. The use case and control system description are more applicable to the Smart Grid concept rather than a Microgrid system, but the description of the software methods used is very helpful in understanding the system level testing and the evaluation of distribution protection and control equipment.

2.3 Applications

The most useful resources on Smart Microgrid applications has been described in the background section of Chapter 1. Other resource include the following: [35] [36] [37] [38] [39] [40] [41].

2.4 Review Conclusions

Formal software methods have not been effectively applied to the design of an abstract Microgrid control system. A formally applied method used in a practical application could establish a standard for how much processing power is needed in an embedded system used to implement a Microgrid control system. If a plug and play architecture is a desire for a flexible Microgrid system, then measurements from a formally applied software design will help the initial design of a system before any hardware is assembled. It is therefore the intent of this dissertation to provide a formal method to design an embedded system with guidelines for the purpose of implementing that system into LabVIEW software.

Chapter 3

Hardware Design

3.1 General Description

The University of Texas at Arlington (UTA) houses a clustered microgrid setup, consisting of three Microgrids as shown in Figure 3-1, that encompasses renewable sources (e.g., solar, wind, and fuel cell), conventional sources (e.g., diesel generator) and electrochemical energy storage. National Instruments (NI) CompactRIO embedded control system provides a platform to study peer-based communication and control schemes. Each sub-Microgrid shown in Figure 3-1 has its own distributed energy resource (DER), dynamically reconfigurable real-time control system, and dedicated local loads. This enables each of the sub-Microgrids to function as an independent islanded or grid-tied Microgrid. The central ring architecture enables each sub-Microgrid to serve the other two via a central AC bus. Each of the three sub-Microgrids harnesses wind energy through the installation of 300 W HiVAWT vertical axis wind turbines and solar energy via 230 W Schott solar panels. A total of 1.2 kW of wind turbines and 2.76 kW of solar panels are installed. Each of the three sub-Microgrids has four, ~30 V, 230 W solar panels installed that are connected in a 2 series / 2 parallel configurations. The solar panels installed on sub-Microgrid one and three are fed into each grid's own dedicated FLEXmax 60 maximum-power-point-tracking (MPPT) charge controller manufactured by Outback Power. Each of the series stacked solar panels connected on sub-Microgrid two is fed into its own Xantrex C40 PWM charge controller. Microgrid 1 has two DS300 wind turbines installed while the other two microgrids only have one each. Wind turbines are fed directly into their own 400 W MPPT charge controller. The regulated DC output voltage from each charge controller ties into the energy storage which provides energy when the AC grid is unavailable or the cost of energy from the legacy grid is prohibitive.

Each sub-Microgrid is setup to have a 17 VDC – 31 VDC bus. Sub-Microgrid one has a NEXA™ 1.2 kW proton exchange membrane (PEM) fuel cell manufactured by Ballard Power Systems. A 5 kW DC-DC buck converter developed by Zahn Electronics conditions the fuel cell's unregulated power and ties it into the grid's 24 V DC bus.

Figure 3-2 shows the detailed schematic of the clustered microgrid setup. As seen, the DC bus on each sub-Microgrid connects directly to a dedicated grid-tied inverter manufactured by Outback Power Systems. A GTFX1424, 1400 W inverter is installed on sub-Microgrid's two and three while a GTFX2524 2500 W inverter is installed on sub-Microgrid one due to the higher input power available there. Each inverter has its own dedicated control system. The control system has user-defined inputs which can be manipulated using an external interface device. Each inverter supplies a 120 VAC, 60 Hz, single phase voltage and the output is first passed through a no-fuse-breaker (NFB) and then through a digitally controlled solid state relay (SSR) providing real time control of the inverter output.

Each grid has dedicated loads connected to its central AC bus via a series connection of a NFB and a SSR to ensure safety, controllability, reliability, and flexibility. A 3.6 kW, AC/DC, 63803 programmable load manufactured by Chroma Systems is connected on each grid to simulate a realistic load profile and implement transient loading capability. Conventional loads such as light bulbs, fans, etc. can also be installed on each grid and controlled using a dedicated SSR on each load line. The digital controllability enables researchers to develop load-shedding algorithms to ensure that the most critical loads have power in the event of a shortage. This architecture was chosen since the DoE's Smart Grid concept involves the communication and interconnection of several independent Microgrids (in what could be considered Smart Nodes) to promote system reliability and efficiency.

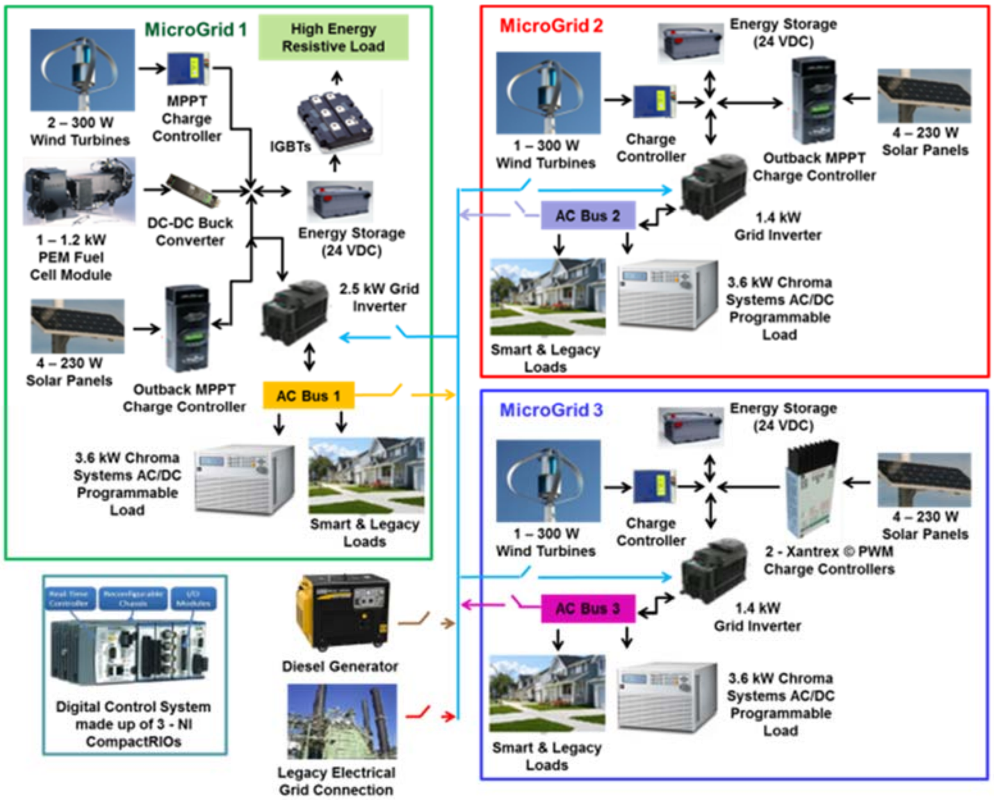


Figure 3-1 - Block diagram of the current UTA Microgrid testbed

The inverters normally operate in a grid-tied mode using the legacy electrical grid as a voltage and frequency reference. This mode of operation allows the inverters to use the legacy grid to power loads and charge energy storage devices. In the event of a legacy grid outage, the inverters automatically convert over to an islanded configuration, using the batteries to source the loads. If any particular grid's energy storage runs low and critical loads still need power, the AC output bus from any of the other Microgrids can connect to the central AC bus, acting as the new reference grid, and source power to the grid(s) in need. Photographs of the Microgrid testbed are shown in Figure 3-3.

Each sub-Microgrid has its own real-time monitoring and control system. A dedicated NI CompactRIO operates each sub-Microgrid. The CompactRIO's hardware architecture incorporates a reconfigurable field-programmable gate array (FPGA) chassis, an embedded controller, and the ability to easily swap out different I/O modules as needed. The CompactRIO is programmed using NI's LabVIEW graphical programming tools and can be used in a variety of embedded control and monitoring applications. Each CompactRIO has I/O modules that provide a wide range of digital output, digital input, analog output, and analog input capability. The digital and analog output signals are primarily used for relay and device control. The digital and analog input channels are used for real-time monitoring, data collection, and feedback control. Each CompactRIO is controlled using a remotely accessible visual user interface. The voltage and current at every relevant input and output is monitored using voltage probes and Hall effect current sensors whose outputs are fed directly to the analog input channels on the sub-Microgrid's respective control system.

Resembling the control hierarchy of the legacy grid, a hierarchical control structure is conventionally adopted for Microgrid operation [6]. The highest hierarchy, the tertiary control, is in charge of economical operation and coordination with the distribution system operator. It assigns the Microgrid voltage to carry out the scheduled power exchange between the Microgrid and the main grid. To satisfy the voltage demand of the tertiary control, the secondary control measures voltages across the Microgrid and, accordingly, updates the voltage set points for the primary controllers. The primary control, typically implemented locally with a droop mechanism, regulates the output voltage of individual inverters. Using the NI CompactRIO, this three-tiered hierarchical control system architecture has been employed. In particular, the primary and secondary tiers of this hierarchy deal with inverter controls to provide proper voltage control, frequency synchronization, and proportional power sharing among inverters. A flexible decision-making infrastructure, that enables interplay among clusters of microgrids while serving a diverse set of dynamics loads, is the ultimate research goal of the UTA Microgrid lab. Commercial off the shelf (COTS) devices are considered since this will be most likely how the DoE's Smart Grid will evolve as it becomes reality.

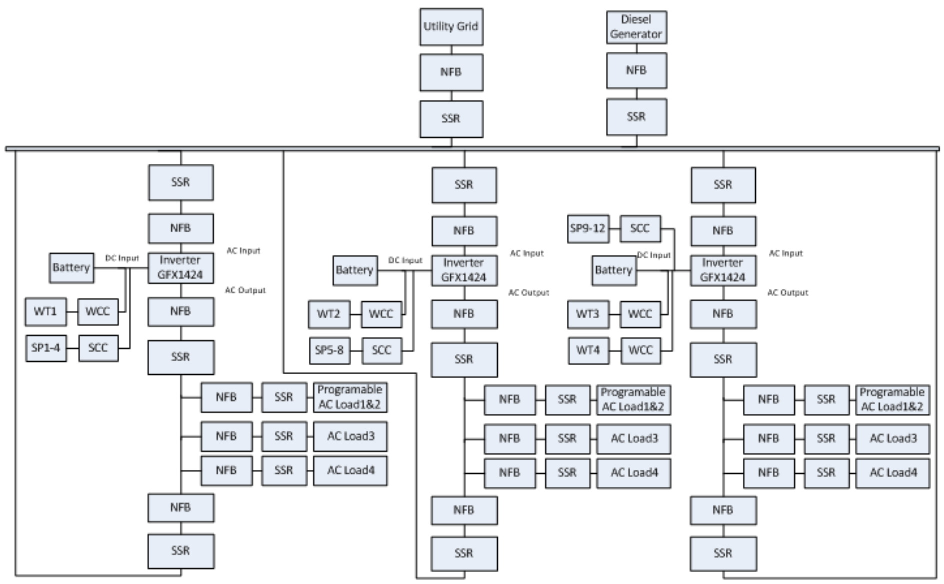


Figure 3-2 - Schematic of Microgrid testbed installed at UTA

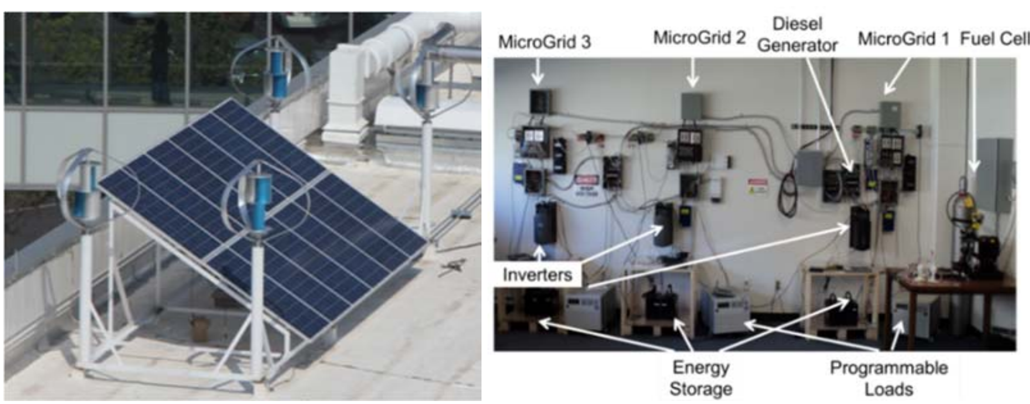


Figure 3-3 - UTA Microgrid's roof installation of solar panels, wind turbines, and diesel generator (left) and internal laboratory setup (right)

Chapter 4

Proposed Control System Design

The use case of a Microgrid can be imagined in many unique ways. Some examples might be a single family residence, a university campus, a hospital, a military forward operating base. A specific use case can be normalized in a number of ways. To do this the loads in the system have to be categorized and prioritized. Some loads will be given critical priority status. Other loads will be categorized as non-essential. In a hospital, critical loads would include life support systems and operating rooms. In a forward operating base defensive weapons, communications and radar system may be the most critical loads. For a single family residence many loads may be considered non-essential. Loads that are non-essential may be shed or turned off as a way to optimize the efficiency of the overall Microgrid. Load shedding may be done in the event of insufficient power from the local sources or for the purpose of avoiding excessive pricing from the utility.

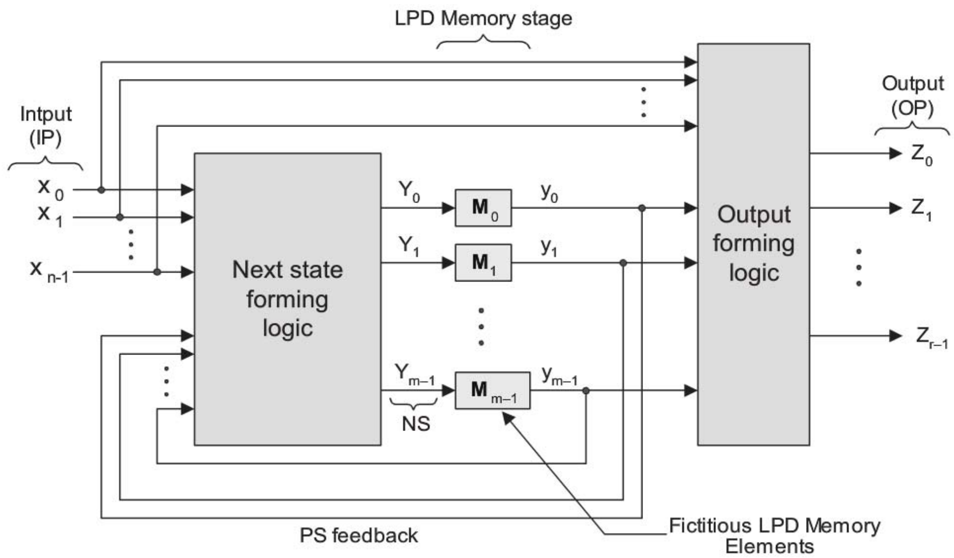
The design of the control system for the Microgrid can be broken into two parts. First the abstract definition of the use case will define the overall desired behavior of the grid. Second the actual implementation of this use case must be written in a programming language for the grid.

4.1 Control System Design

For the Microgrid system at UTA using CompactRIO embedded controllers, it is helpful to think of the system in terms of a state machine. In fact, it can be said that any embedded system is easiest to conceptualize as a state machine. A state machine is a mathematical model used to develop a logical process or in this case, design a software application [42]. It can be thought of as a machine with a finite number of operational conditions called ‘states’. The machine can be in only one state at a time and can transition to another state based on some event or trigger. Therefore designing the software is a matter of defining the states and deciding on the events which cause the states to transition from the current state to the next state.

For the design presented here, an asynchronous finite state machine is chosen [43]. Finite because the machine will have only a finite number of states. Asynchronous mainly because the machine is going to be implemented in software. Synchronous clock driven machines are more of a problem to design in hardware.   Very large scale clock distribution becomes a problem. Accounting for delay and clock skew in every possible scenario becomes increasingly complex. This will not be a problem with a software driven machine where the procedural execution can be controlled programmatically [44].

Figure 4-1 shows a diagram of the generalized state machine that will be the basis of the design. This is considered the Mealy model of the state machine (named after G. H. Mealy) where the input values X_i can directly affect the output Z_i as well as the next state (NS) Y_i. Delay between NS and the output logic is represented by memory elements that may be cumbersome in a hardware design, but simple in software.



[42]

Figure 4-1 - Generalized LPD Model for Asynchronous Mealy FSM

For a software implementation, the NS forming logic is considered ideal since the memory elements encapsulate any delay in the logic structure. The very nature of software allows for easy implementation of the memory elements needed to store information during a state transition.

To begin the realization of this state machine, a state diagram must be designed. The example state diagram shown in Figure 4-2 is a portion of a state diagram that contains all of the elements of a fully documented state machine. Each unique state is represented as an oval on the diagram (labeled here as states a, b, and c). The definition of the present state (PS) is given by the set of the PS variables. Branching paths are shown as arrows drawn from one state to the other and represent the machine transitioning from one state to another. Branches can also loopback to from a state back to itself. The conditions (or events) which cause a transition to occur are shown in functional notation on the branch path. The notation denotes the path and the input x_i which caused the transition to occur.

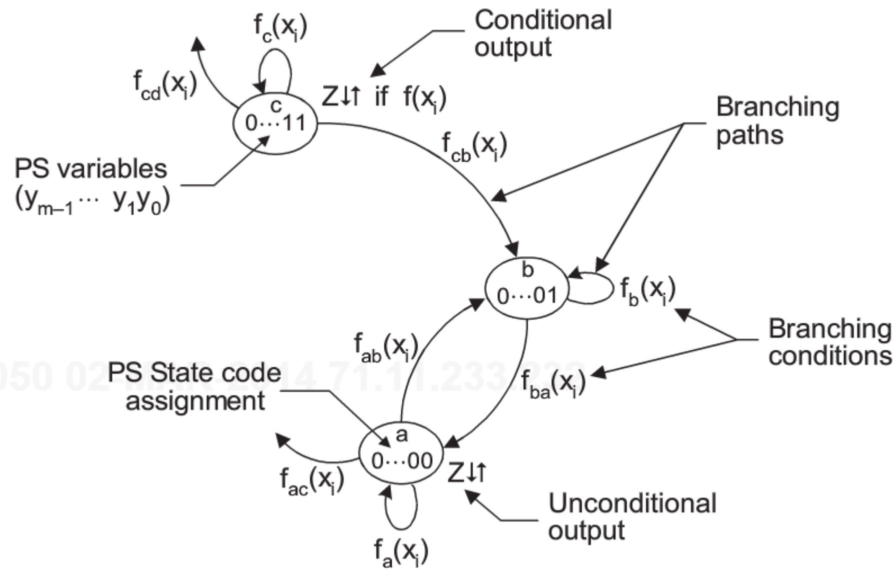


Figure 4-2 - Fully documented state diagram for the asynchronous FSM [42]

Branching accountability is met by the sum rule. The sum rule states that the logical sum of all branching conditions from a given state must have a logic value of 1. The sum rule ensures that all branching conditions are accounted for. Any branch that is not accounted for in the state machine may cause the machine to malfunction. A special exception could be made to the branch rule if it can be assured that a certain input condition is never allowed by the system. This exception will become very important to the matter of building a machine for the purpose a Microgrid. Branches must also be mutually exclusive. That is to say that each possible branching condition can only control one branching path. A violation of this mutually exclusive requirement forces a machine to be technically non-deterministic. Again, this violation will come up in the process of building a machine for the Microgrid, but a proper methodology will allows these violations to be handled in software.

4.2 Implementation of a Specific Use Case

The specific states a Microgrid may need to transition into could vary widely depending on the application for which it is being used. In any case, there are almost always critical loads which must be powered at all times and loads which can be shed in the event of a power shortage. The state machine must transition between states as efficiently as possible ensuring that first the critical loads are powered at all costs and second that the grid is operating in the most efficient and reliable manner possible at any given time. With this later demand in mind, it can be imagined that the state machine would prioritize use of the renewable sources as much as possible before switching to the utility or other backup for power. In residential applications, efficiency will often be given the highest priority as this is an application where few critical loads may exist. On the other hand if a Microgrid is employed in a military forward operating base (FOB), higher priority would be given to ensuring that essential needs such as the communications headquarters, medical station, and radar/weapons systems are always powered. This will often require the grid to automatically shed any unnecessary loads in order to always ensure critical loads are being powered. In either residential or FOB applications, it can be envisioned that multiple Microgrids may be located in close proximity and the ability of the state machine to seamlessly interconnect them as individual grids fall short of the power demanded from them is critical.

The first step in the creation of a state machine is to decide on the possible operational states of the Microgrid. These states used in the machine developed here include Renewable Generation (RG), RG Charging, Battery Backup, Grid Tied, and Dark.

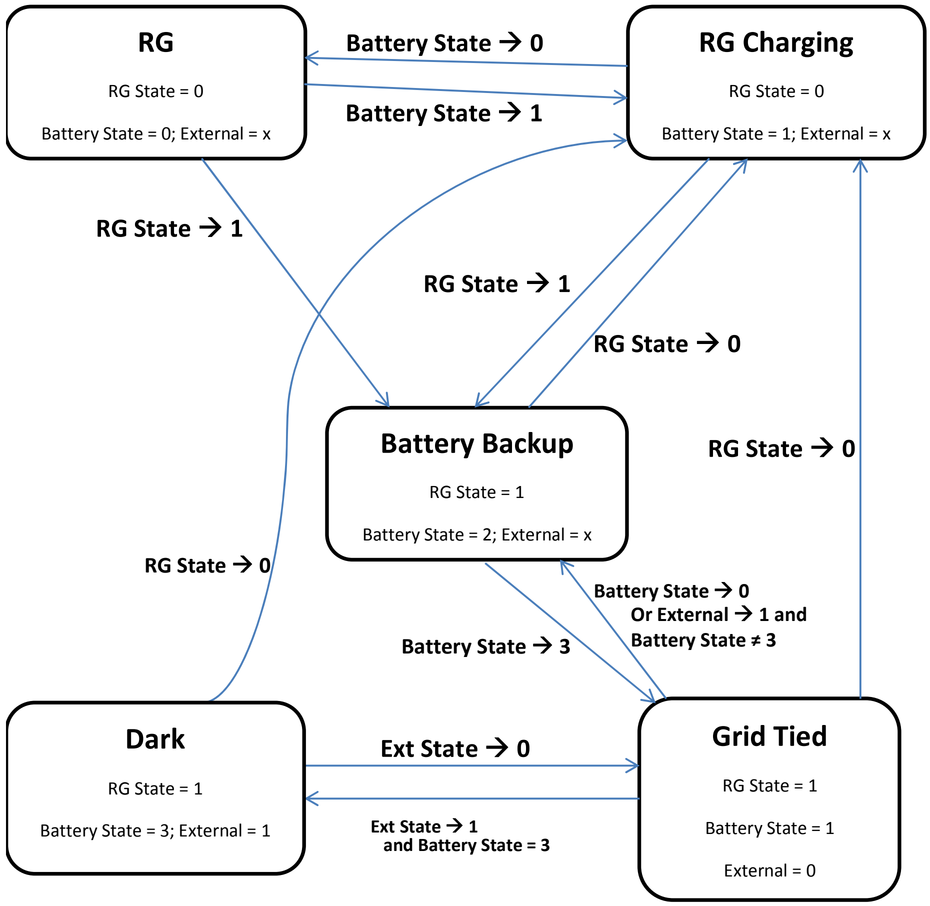
Renewable Generation (RG) is the state where the system is being powered by renewable resources and the batteries are fully charged. This would be the most desirable state for the system. In the RG Charging state, the system is also running off of renewable generation, but the batteries are also being recharged. Again this is a highly desirable state where power is being supplied by the renewable generation. The Battery Backup state is for when the system is running off of the batteries. In this state, renewable generation is not available for some reason and there is a possibility of load shedding procedures coming into play. Grid Tied is the state where the system has switched to being powered from an external AC source. Load shedding is also a possibility in the Grid Tied mode. The actual source of external power can vary. The power fed into each inverter's utility grid connection can either come from the legacy electrical grid, the backup diesel generator, or even another Microgrid. In this state, renewables are unavailable and the batteries have been depleted beyond some predetermined amount. Also, in this state the batteries may be re-charging from the external source. Finally, if all possible sources are unavailable there is a Dark state where all loads are left unpowered.

To complete the state machine, a set of transition events must be defined for each state. For each transition event, one of other states must be chosen as the 'next' state. Each transition event causes the state of the system to move from the 'current' state to a different or 'next' state. Table 4-1 describes the states for a general purpose Microgrid.

Figure 4-3 provides a graphical representation of Table 4-1. The legend at the bottom of the figure describes some numerical values to conditions that define the state. The renewable generation can be in one of two states, available or unavailable. Battery state requires two bits of information to describe the four conditions of the battery (charged, charging, supplying, or depleted). Finally, the state of external power availability is kept in a single bit of information as either available or not. The actual implementation of the state machine is achieved using LabVIEW software embedded in each CompactRIO's FPGA. Utilizing the real time data monitoring and input/output capability, the CompactRIO acts as an autonomous control system based on the state machine just discussed. The LabVIEW application used to run the Microgrid can also use pre-established use cases. One useful property of the application is to set the battery level to a depleted state in order to affect the state machine. This allows test case transitions to be triggered without actually waiting for the batteries to discharge.

Table 4-1 – Abstract State Transition Definitions

Current State	**Next State**	**Transition Event**
RG	RG Charging	Energy storage level drops below certain level
	Battery Backup	Renewable sources become unavailable
RG Charging	RG	Energy storage reaches capacity
	Battery Backup	Renewable sources become unavailable
Battery Backup (possible load shedding)	RG Charging	Renewable sources become available
	Grid Tied	Energy storage becomes depleted
Grid Tied (possible load shedding)	RG Charging	Renewable sources become available
	Battery Backup	Energy storage level reaches capacity
	Dark	External source becomes unavailable
Dark	RG Charging	Renewable sources become available
	Grid Tied	External source becomes available



RG State – Renewable Generation Available = 0; Unavailable = 1

Battery State – Battery Fully Charged = 0; Charging = 1; Supplying = 2; Depleted = 3

External Source State – External Power Available = 0; Unavailable = 1

Figure 4-3 - Microgrid State Diagram

It should be noted that the state diagram in Figure 4-3 is not a fully documented state diagram as described in section 4.1. The loopback branches are not shown in order to keep the diagram neat and easier to read. The state diagram is fully realized as the formal state transition table is created. The state transition table contains all the information from a fully documented state diagram and is ready to be used to implement the machine software. As shown in Table 4-2, the state transition table lists all of the present states (PS) in the left side column.

Table 4-2 - Formalized State Transition Table

Events / States	RG0	RG1	BS0	BS1	BS2	BS3	Ext0	Ext1
RG	RG	BB	RG	RGC	x	x	x	x
RGC	RGC	BB	RG	RGC	x	x	x	x
BB	RGC	BB	x	x	BB	GT	x	x
GT	RGC	GT	BB	GT	x	x	GT	(BB or D)
Dark	RGC	D	x	x	x	D	GT	D

The events that cause a transition are listed along the top row. The body of the table contains the next state (NS) corresponding to that transition. Next states that correspond to a loopback transition have curved arrow in the table. Next states that are “not possible” next states are shown with an x. The notation used in the events row at the top of the table corresponds to the values shown in the legend of state diagram in Figure 4-3.

It should be noted that the transition table shows the exact kinds of exceptions and violations described in section 4.1. In the table there are many places where instead of a next state there is an x. This is an exception to the branching accountability rule and denotes that this is an illegal transition. This must be accounted for in the software. There is also a violation of the mutual exclusion requirement. When the present state is grid tied (GT) and the external source becomes unavailable, the next state could possibly be either to go to battery backup or to dark mode. This violation must also be handled properly in software.

The guidelines for the proposed implementation method and rules for handling the exceptions and violations are in the next section.

4.3 Software Implementation

The fully documented state machine can now be implemented in LabVIEW software. In the discussion that follows the term ‘vi’ will refer to a LabVIEW program. Executable programs written in LabVIEW are known as ‘virtual instruments’ or vi’s. The computer files are saved with the extension .vi instead of .exe as with traditional executable programs.

These are the details of the proposed method for this dissertation.

Guidelines for software implementation:

- The software will contain **one** while loop which continuously executes the states
- A case statement inside the while loop contains one case for each state
- Code in each case determines the next state in the sequence

- A shift register is used to carry state information (in effect acting as memory) to the next iteration of the while loop
- The "impossible" states are trapped in the transitional code and disallowed
- Non-deterministic transitions require additional decode in the transitional code to allow a reasonable transition.

These guidelines are now used to implement the Microgrid from Table 4-2.

The front panel of the top level software is shown in Figure 4-4. There are no user controls as this is a 'headless' embedded system. The intent is for this software to work autonomously with no user interaction. It may be necessary to have the program report data and respond to a network management system, but there is no direct user interface.

Figure 4-4 - Top Level Microgrid Application

The block diagram of the application shows the controls of the system as laid out in the guidelines. One outer while loop contains one case structure. The state of the system is defined as an enumerated data type indicated by the blue wire coming from outside the while loop on the left side of the screen. This enumerated data type has a value defined for each of the five states. The renewable generation state is the initial state and is the first to activate when the program begins.

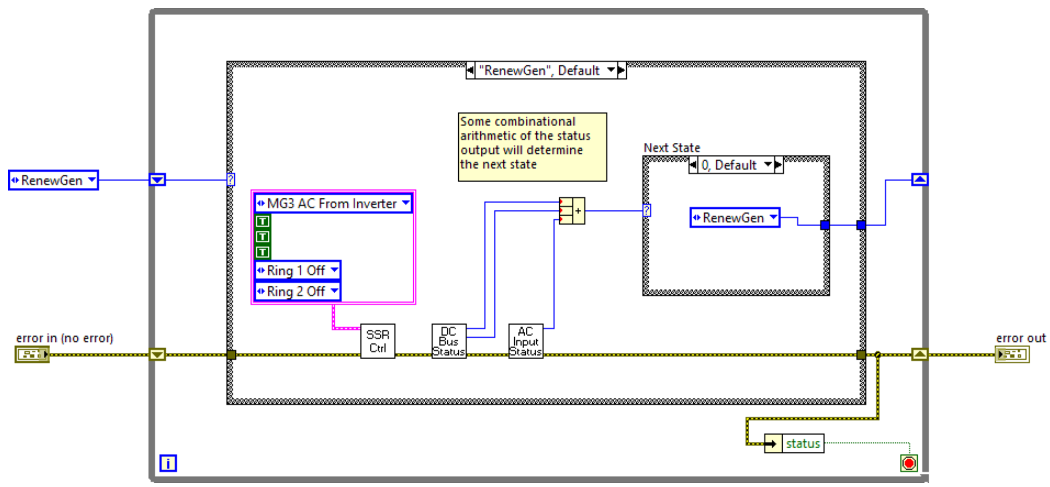


Figure 4-5 - Microgrid Block Diagram

Inside the renewable generation case are three LabVIEW sub programs (sub-vi's) that represent the transitional code that will determine the next state for the machine. The sub-vi's show up as the three white squares and execute in order from left to right. The first block is the sub-vi that operates the solid state relays in the system. This vi is always chosen to run first to satisfy the protection requirements of the system. The input to this vi is the fixed information about the relays for how the Microgrid should be connected in the renewable generation state. The second vi monitors the DC bus of the Microgrid. This vi returns two things: the status of the renewable generation equipment and the status of the battery. The third vi is a monitor for the AC bus of the Microgrid which includes any external AC source that the Microgrid could potentially use. The output of the two monitoring vi's is combined into a value for the next state of the Microgrid. The next state case on the block diagram operates from that combined logic. The next state case from the renewable generation case can only put out three values as seen in the formalized state transition table, Table 4-2. The loopback state is the default state. The other two states are, battery backup and renewable generation charging. These cases are shown in Figure 4-6.

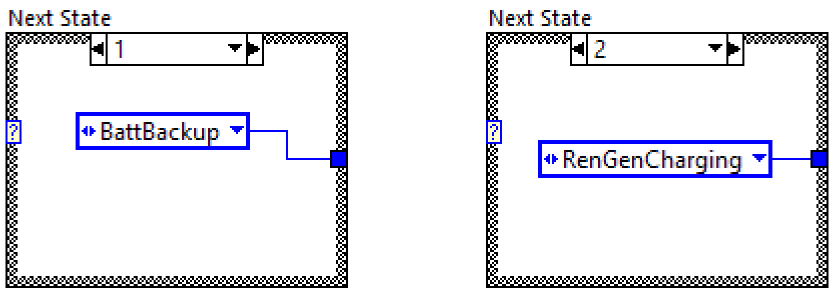


Figure 4-6 - Transitions from Renewable Generation

The rest of the states will look very similar to this first one. The differences will be in how the solid state relays are set and the arithmetic logic that determines the next state. But in each state, the same general procedure is followed. First, operate the solid state relays according to the desired state. Second, check the status of the renewable generation, batteries, and external sources. Third, calculate the next state and make the transition to that state.

In Figure 4-7, the renewable generation charging state is shown to be almost identical to the renewable charging state. The main difference is that the default loopback is to itself and the last transition is back to renewable generation.

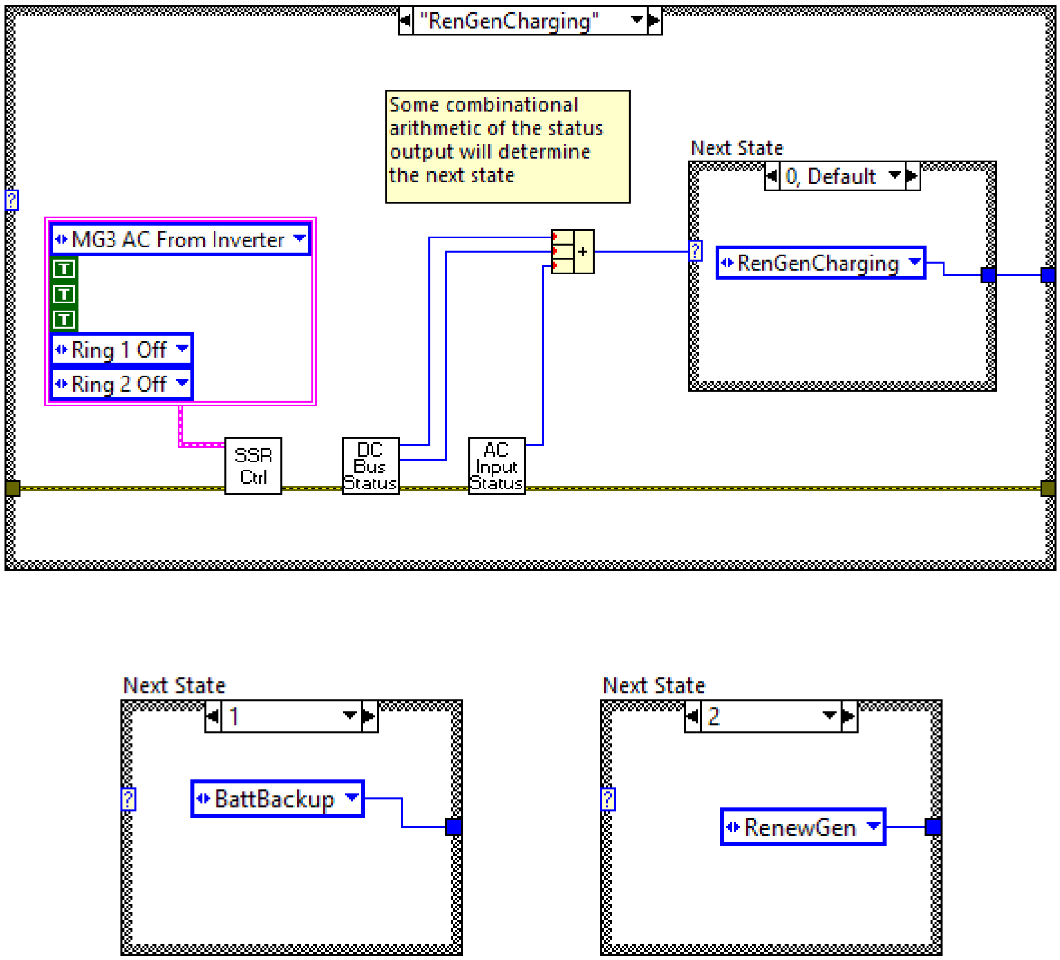


Figure 4-7 - Renewable Generation Charging State

For the battery backup state shown in Figure 4-8, the next state transitions are shown as renewable generation charging and grid tied.

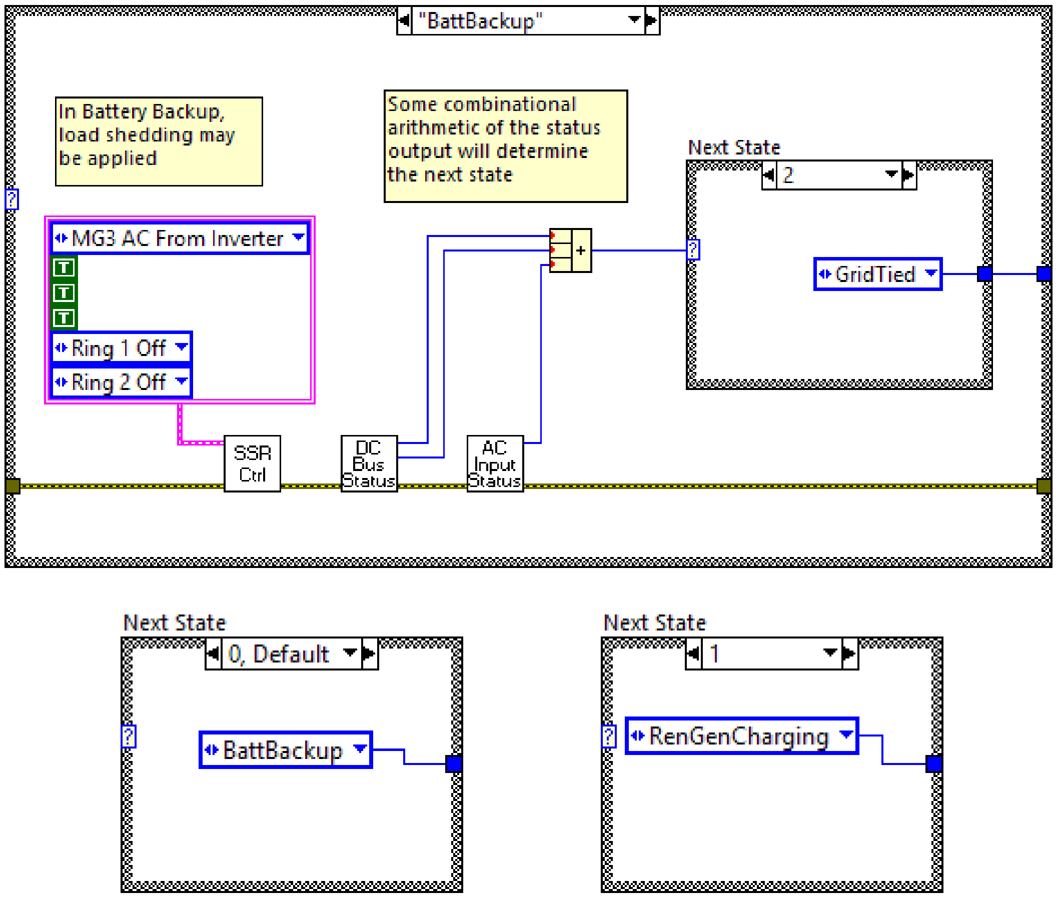


Figure 4-8 - Battery Backup State

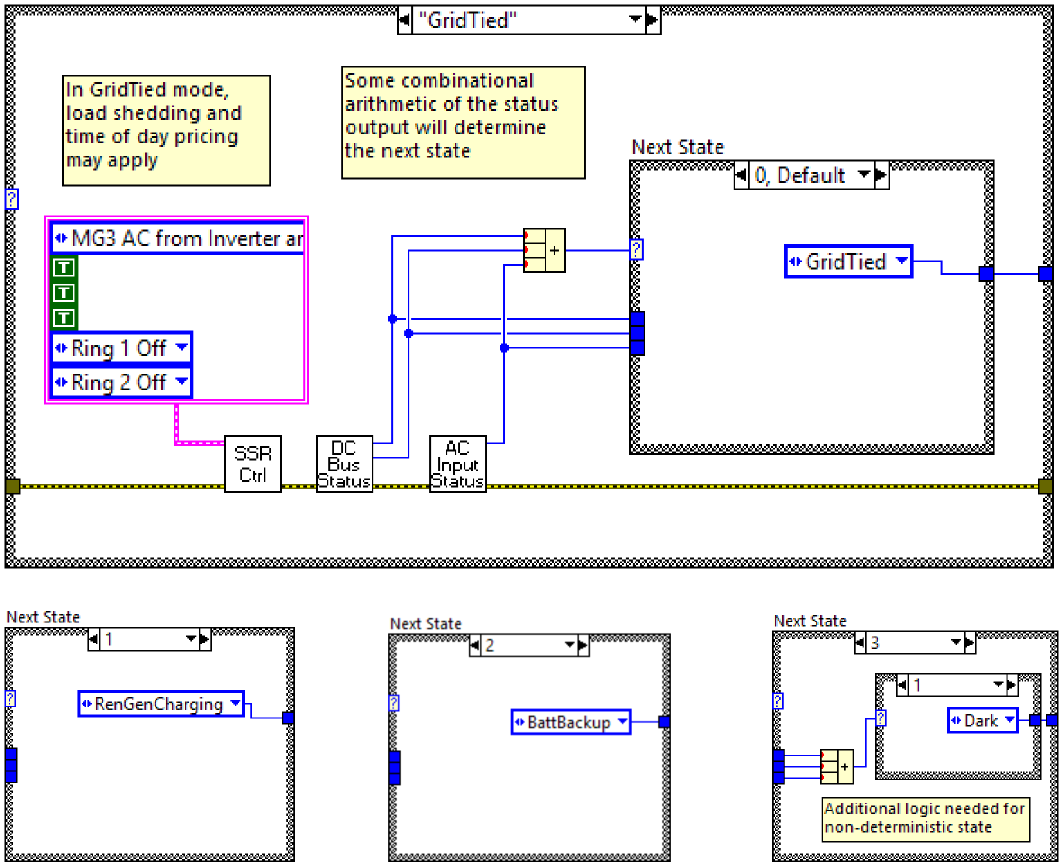


Figure 4-9 - Grid Tied State

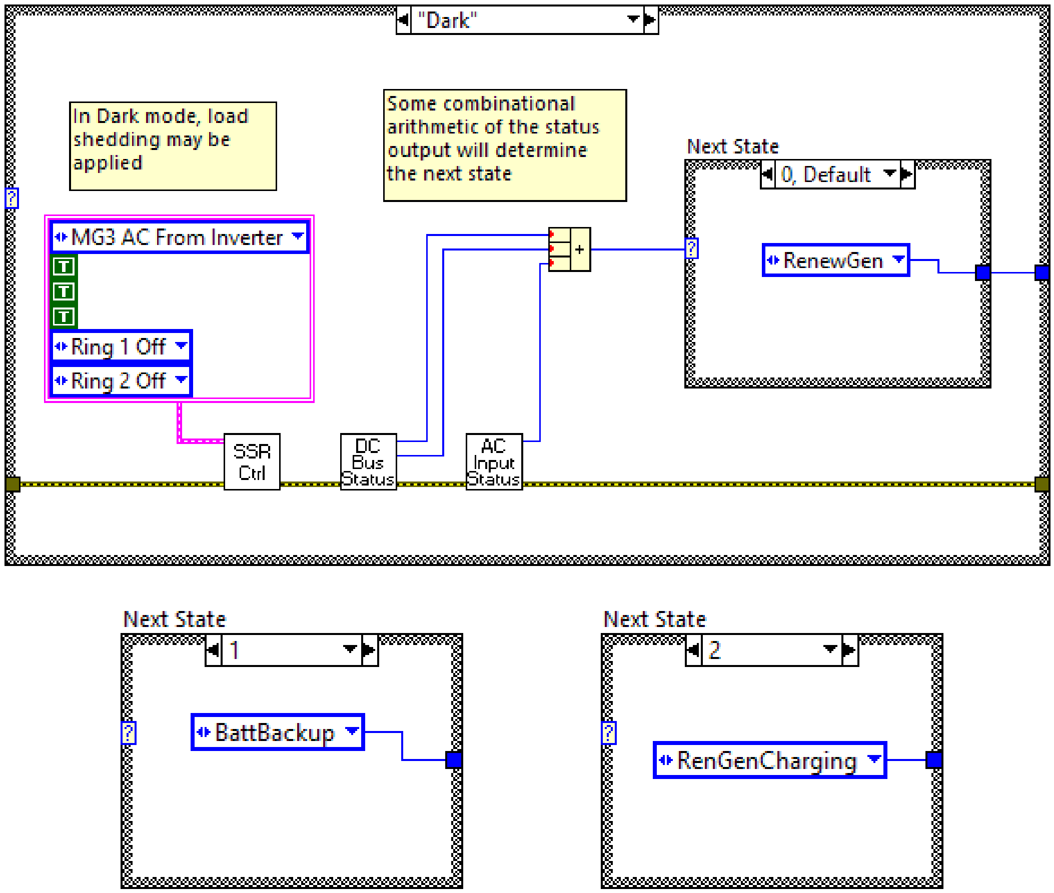


Figure 4-10 - Dark State

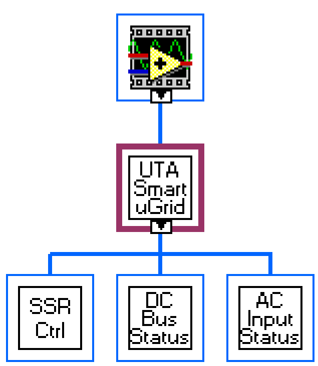
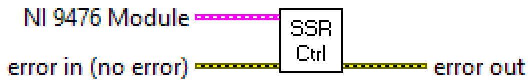


Figure 4-11 - LabVIEW Subroutine Hierarchy



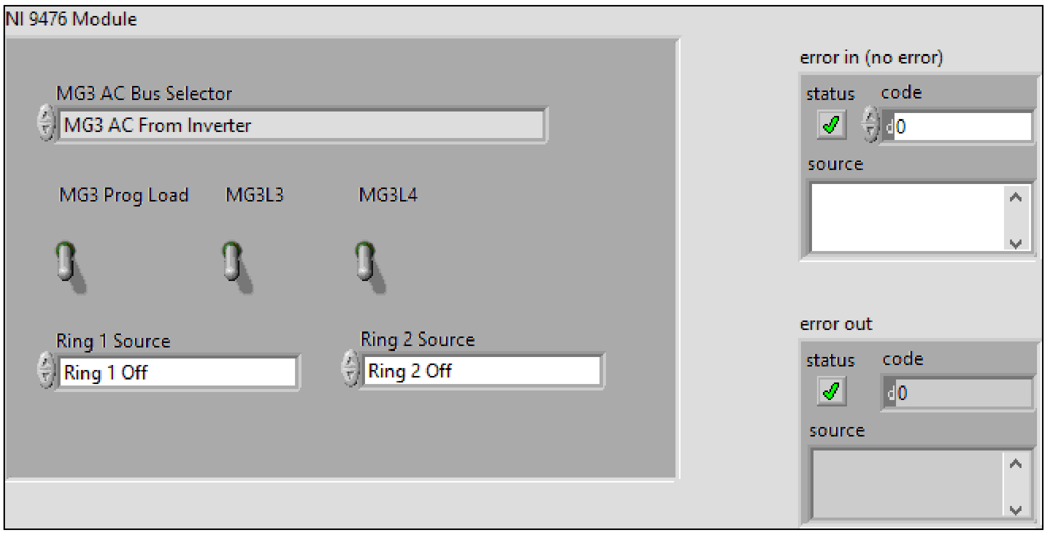


Figure 4-12 - Solid State Relay Control Subroutine

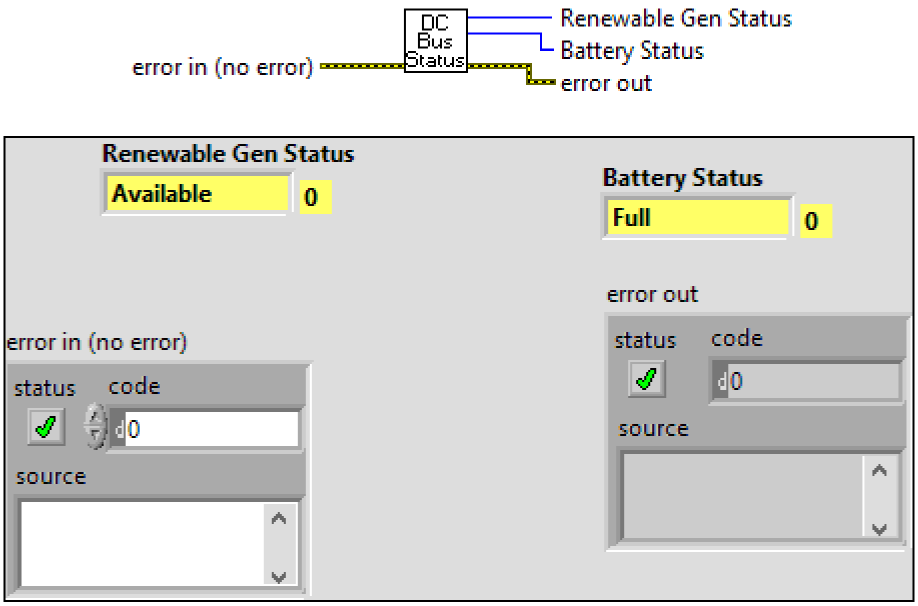


Figure 4-13 - DC Bus Status Subroutine

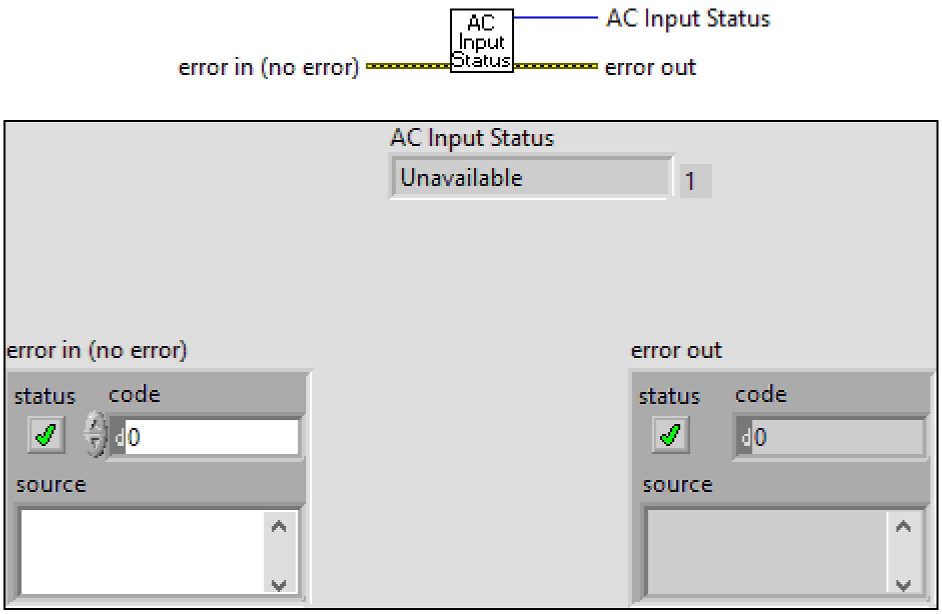


Figure 4-14 - AC Bus Status Subroutine

4.4 Embedded Software Execution

A description of how the software is running with regard to the hardware platform is necessary to fully appreciate the power of a fully autonomous Microgrid. It is important to understand that this software is running in a real-time operating system (RTOS) that is physically located on the Compact RIO system. This is an important distinction from software that runs on a Windows or Macintosh general purpose operating system (GPOS) that is logically connected to the Microgrid through data acquisition (DAQ) equipment. Application software running in a RTOS is capable of providing deterministic timing as data is sampled. This allows the system to process and react to changes in the system as they occur. For an application running in a GPOS using DAQ equipment, it is possible to have the equipment sample data at a specific rate, but the data can only be post processed. Control is performed after the fact so a deterministic reaction to an event is not possible.

Developing the software is a different story. For the purpose of developing the software, a Windows or Macintosh workstation is vital. The software for this dissertation is being developed on a Windows 8.1 workstation with 12.0GB of RAM and an Intel Core i7 microprocessor running at 2.40GHz. This is a fairly high end workstation at the time of this writing and has a great amount of processing power for the purpose of software development and debugging. It is a very stark contrast with the amount of processing power on the Compact RIO's where the application will actually execute. Of the three Compact RIO's in the system the one with the most processing power contains a PowerPC class microprocessor running at 533MHz. It has 256Mbytes of RAM and is capable of handling deterministic control loops of about 2kHz. This is quite a contrast in processing power between the development system and the system to which the software will actually be deployed. In addition, the development tools are aware of the RTOS that exists on the Compact RIO. The tools allow the developer to place deterministically timed loops in the software in order to implement real time monitoring and control.

The software is developed on this workstation, cross-compiled and then deployed to the Compact RIO. The cross-compilation step targets the microprocessor that exists inside the Compact RIO.

The control software for the Microgrid is written in LabVIEW. Specifically, LabVIEW 2012 applications have been written for each sub grid and execute on in the real time operating system of each Compact RIO controller. In addition each Compact RIO contains an FPGA that can be utilized to improve the real time processing of the application. The capabilities of each controller limit the power of the application. Specifically the amount of processing that can be done is very dependent on the processor in the Compact RIO and the FPGA that is in the controller.

Chapter 5

Experimental Data

5.1 Resource Allocation

The compilation of each component of the Microgrid FPGA software is shown below in Figure 5-1. The FPGA software is developed in LabVIEW as if it were a regular LabVIEW program. However, each FPGA program is actually synthesized into a digital circuit that is programmed into the FPGA. The basic logical unit in the FPGA is called a configurable logic block or CLB. These are made up of two basic components: flip-flops and lookup tables (LUT). In the FPGA, a register is a group of flip-flops used to store a bit pattern. Each application that runs in the FPGA must go through the process of first synthesizing the digital circuit.

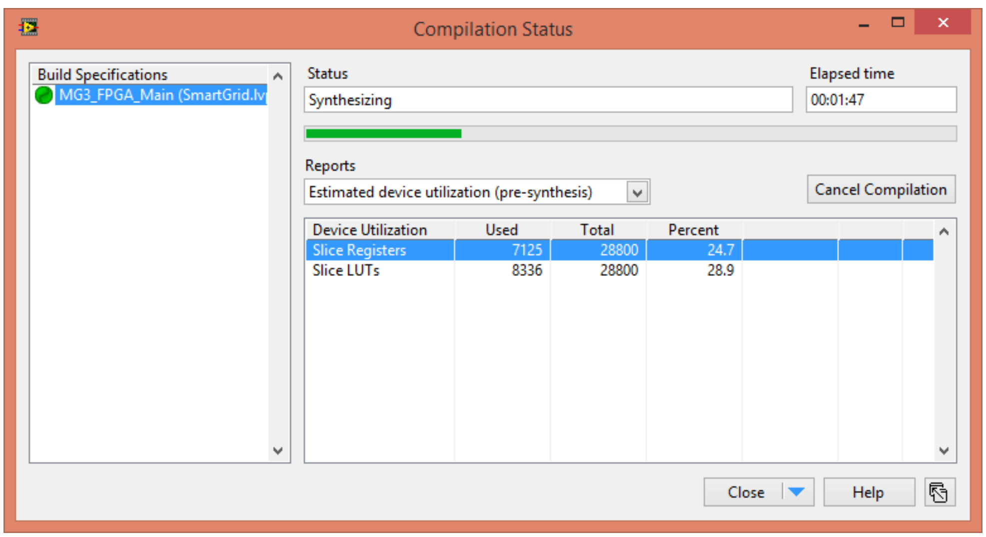


Figure 5-1 - Compilation Status (Synthesizing)

During the synthesis of the circuit the compilation tool reports on the amount of processing units that the program will need in the FPGA. Then the circuit is placed and routed within the FPGA. Finally it is programmed in to the FPGA and is ready for execution. The Xilinx tool used to perform the compilation gives statistics of the resource allocation within the FPGA.

When the circuit is finished being synthesized, the tool will report on the final total of device utilization (see Figure 5-2).

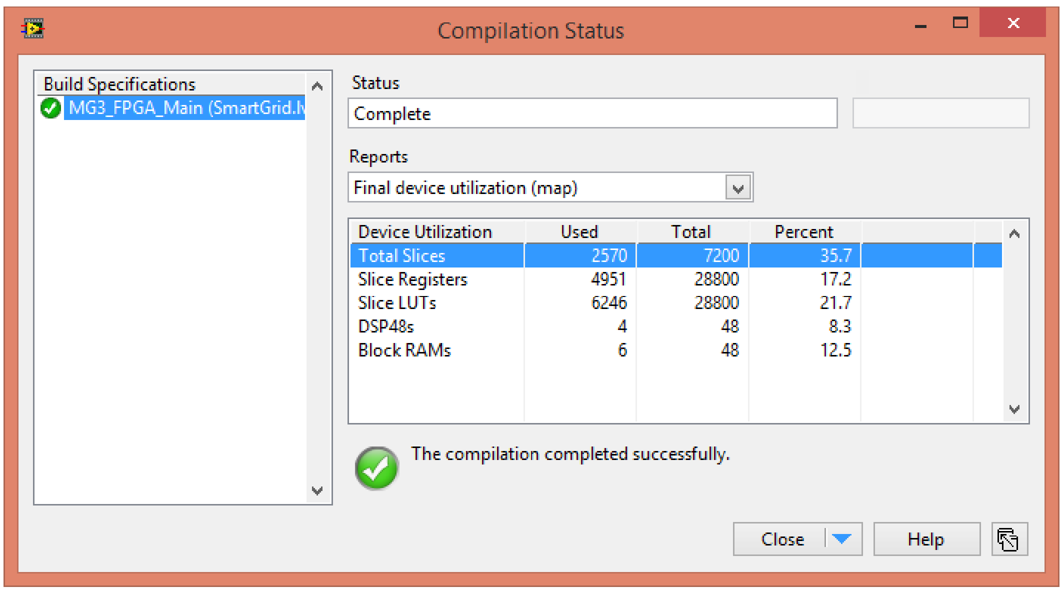


Figure 5-2 - Compilation Status (Completed)

And a final summary report is generated (see Figure 5-3).

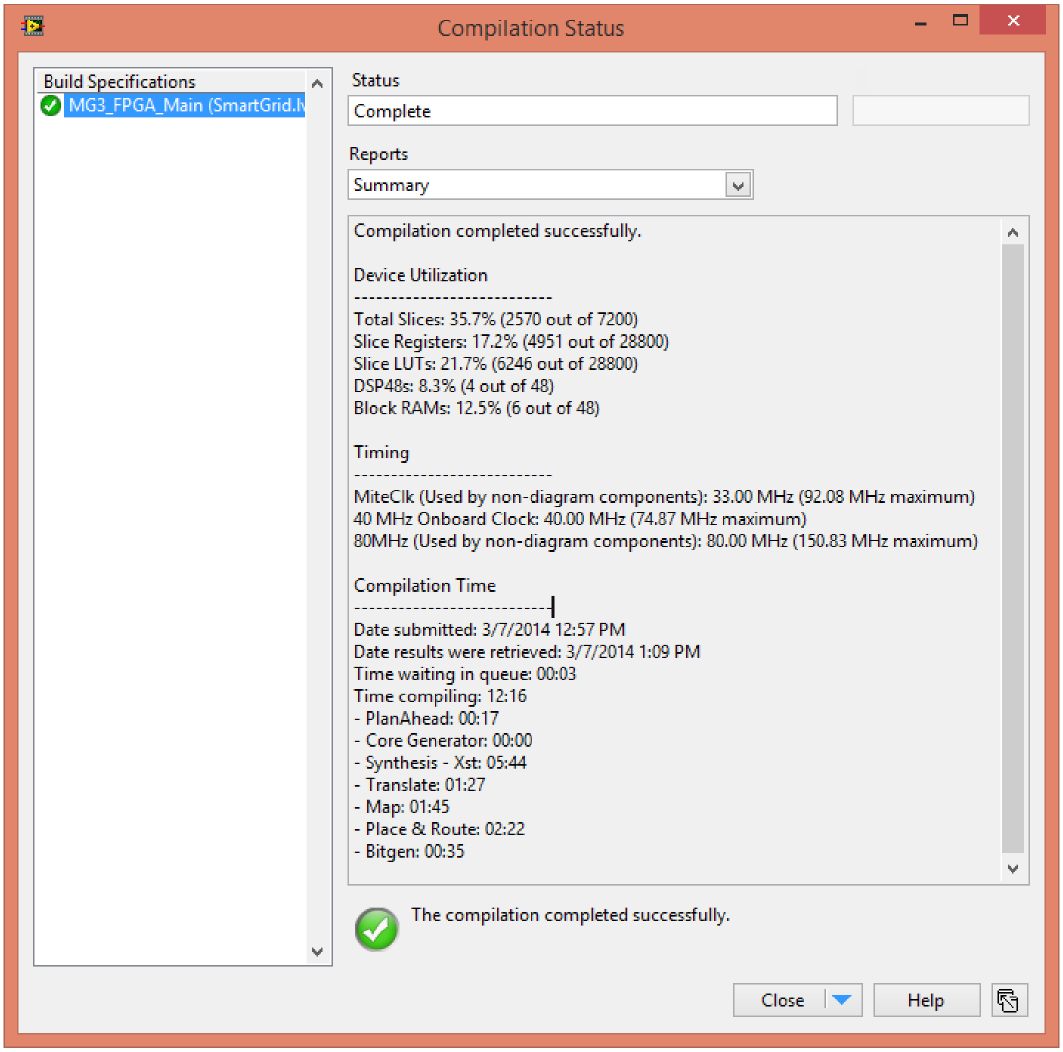


Figure 5-3 - Compilation Status (Summary)

5.2 Microgrid Performance

The data presented here demonstrates the dynamic Microgrid operation. The data is sampled at a rate of 1ms per sample. The system processes these samples every 48 samples or about every three cycles of the 60Hz AC waveform.

In Figure 5-4, the DC bus currents are shown in a steady state operation for about five seconds in time along the x-axis. At this particular moment in time, the wind is calm and no usable energy is provided by the wind turbines (orange line). There is no load connected to the AC bus, but the inverter (gray line) is still drawing some positive current from the DC bus for stand-by operation.

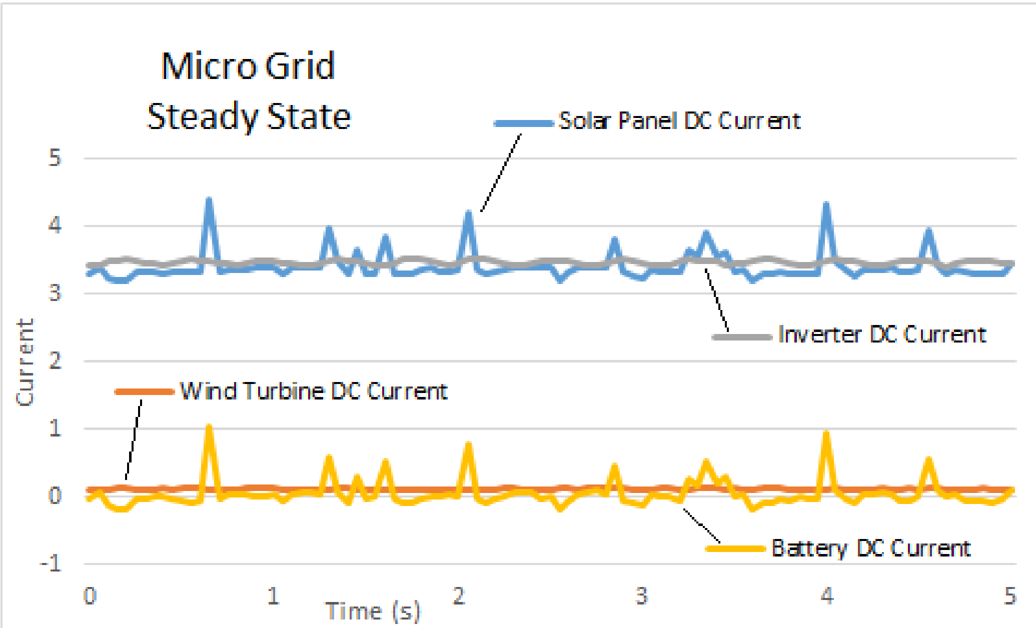


Figure 5-4 - Microgrid Steady State

This current is coming mostly from the solar panels (blue line). The solar panel charge controller is operating as to provide just enough current to the inverter. The battery current (yellow line) shows that it follows precisely the current from the solar panels. When the solar panel current is above the required inverter current, the battery current is also positive indicating that the battery is in a charging state (positive current going into the battery). In the areas where the solar panel current is below the inverter current, the battery current becomes negative indicating that it supplying current to the inverter. In either case, this plots shows a renewable source and energy storage elements working together to support the overall system operation.

Figure 5-5 starts out similar to Figure 5-4 in that it shows how the solar panel charge controller is supplying current to the inverter with excess current going to charge the batteries. One additional plot line has been added to show the AC current output of the inverter (dark blue line). The connection of two large loads on the AC bus produce a current draw from the inverter which cannot be satisfied by the solar panels. The batteries supplement the solar panels on the initial spike and also on the subsequent steady current draw. The loads on the AC bus operate normally and there is no disruption of power service to the loads. When the loads are turned off, the system returns to the initial operating mode. This data has been scaled and normalized so that the AC currents measured on the inverter output can be visually compared to the DC bus current going into the inverter. The AC currents are measured as RMS values.

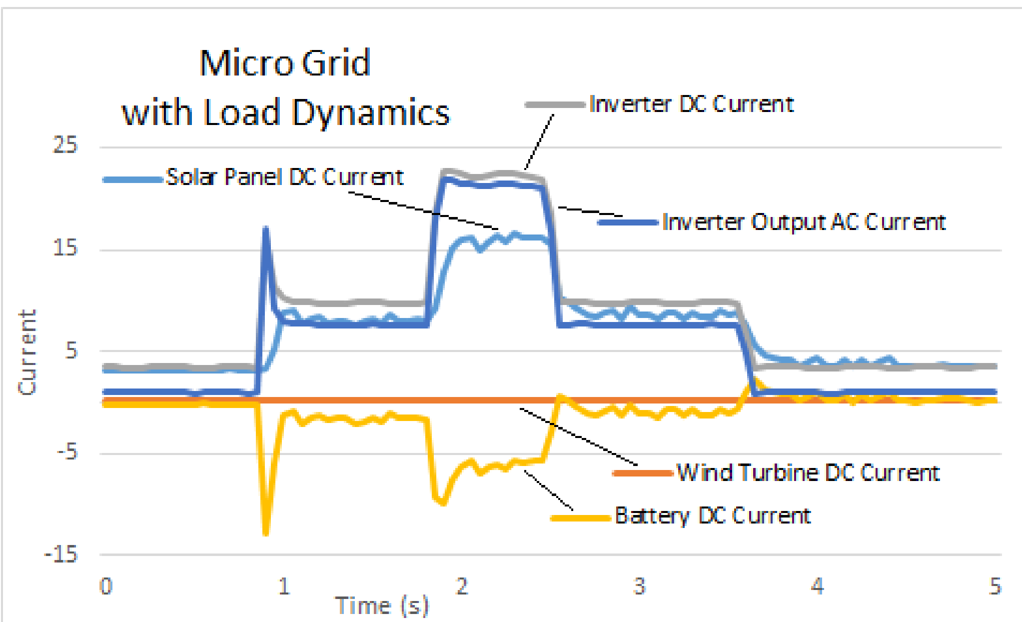


Figure 5-5 - Microgrid showing Load Dynamics

Figure 5-6 shows longer term dynamics in the system (about one minute of time along the x-axis). In this plot, there is a constant AC load on the AC bus indicated by the steady draw of DC current by the inverter and the steady AC current output of the inverter. This particular plot was captured to show the effect of clouds going past the solar panels. For about the first 80% of this plot the current from the solar panels it not sufficient to supply the inverter due to clouds passing over the panels. During this time period, the battery provides the additional current needed to keep the DC current into the inverter constant. The sharp dip in the middle of the graph is the location where the solar panel charge controller switches modes from normal operation to a “stand-by” mode where the controller focuses on supporting the voltage level of the DC bus while the battery supports the extra current needed to keep the system running. The sharp upward spike of the solar panel current near the end of the plot is the point where the clouds have passed. The controller returns to normal operation supplying current to the inverter and charging the battery.

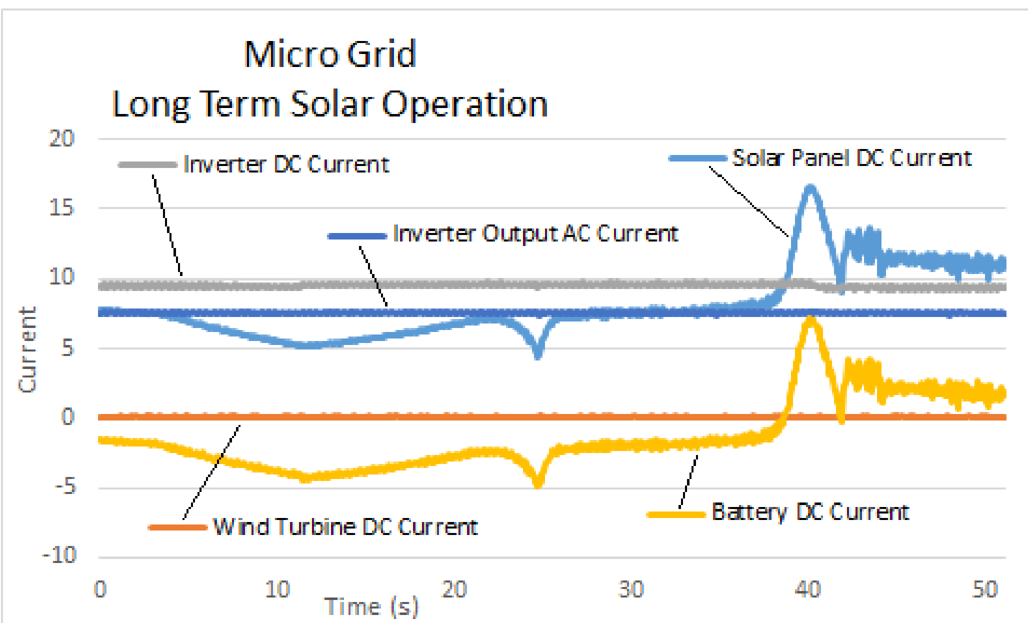


Figure 5-6 - Microgrid Long Term Solar Operation

The previous plots show that the batteries do a fine job of providing current any time that the renewable sources are unable to provide enough current to the inverter. This is exactly the type of operation needed for an islanded grid operation. But what about the dynamic performance of the grid when power needs to come from outside the grid. Figure 5-7 shows the performance of the system when the grid experiences an unexpected outage of the renewable resources. The application has been designed to sense the outage and switch to an alternate source. In this case, power is connected into the AC input of the inverter from the ring bus which is being supplied from another Microgrid. The inverter is designed to automatically switch to its AC input when the DC supply drops. A small delay occurs while the inverter syncs to the new source. This delay is an adjustable setting in the inverter itself. For this example, it has been set to its minimum value and results in the delay shown of about 12 seconds. The plot shows the initial outage of the solar panel at about the 12 second mark and the subsequent response of the batteries to support the inverter current. The complete loss of power from the solar panels triggers the inverter to switch from the DC bus (gray line) to the alternate source (green line). The transition is seamless to the AC bus of the Microgrid and service is uninterrupted (dark blue line). In this case the power is coming from an adjacent Microgrid which happens to have excess power to spare. As power from the solar panel is restored, the system makes a transition back to using power from the DC bus.

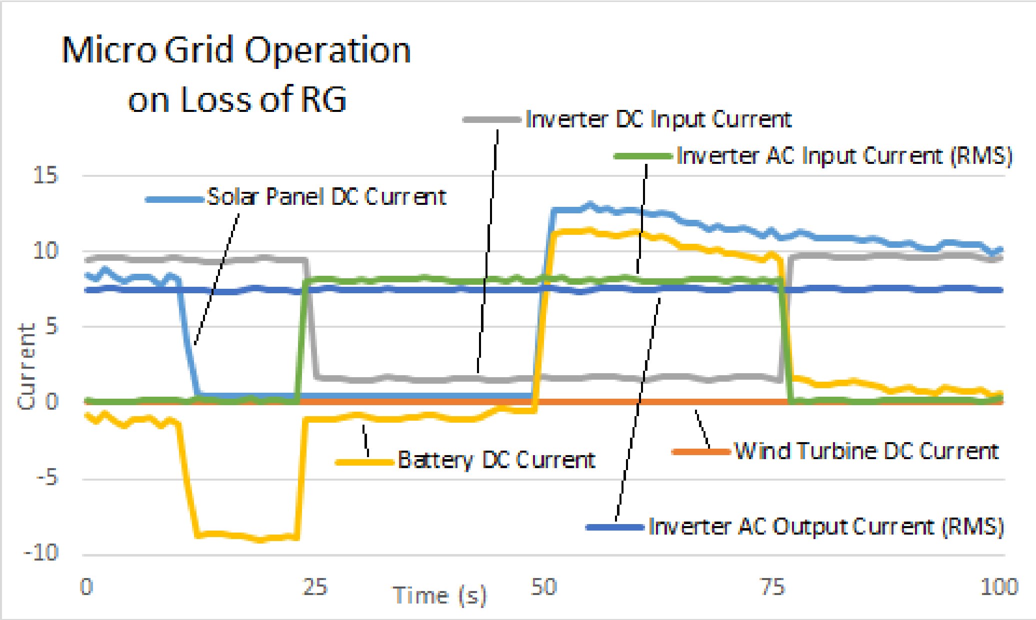


Figure 5-7 - Microgrid Operation on Loss of Solar Power

Chapter 6

Conclusions

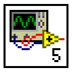
This dissertation takes the formalized methods for state machine design used to build digital circuits and applies it in a unique way that gets around the exceptions and violations that would otherwise hinder the design of a digital circuit. This is mainly because the method is being applied to software rather than hardware. Applications in any embedded system could probably benefit, but Smart Microgrid solutions seem to be well suited due to the fact that they tend to enter a state and stay for relatively long periods of time before making further transitions. This design method is also flexible and scalable. New use cases can be quickly implemented by following a simple set of guidelines that promote fast software prototyping.

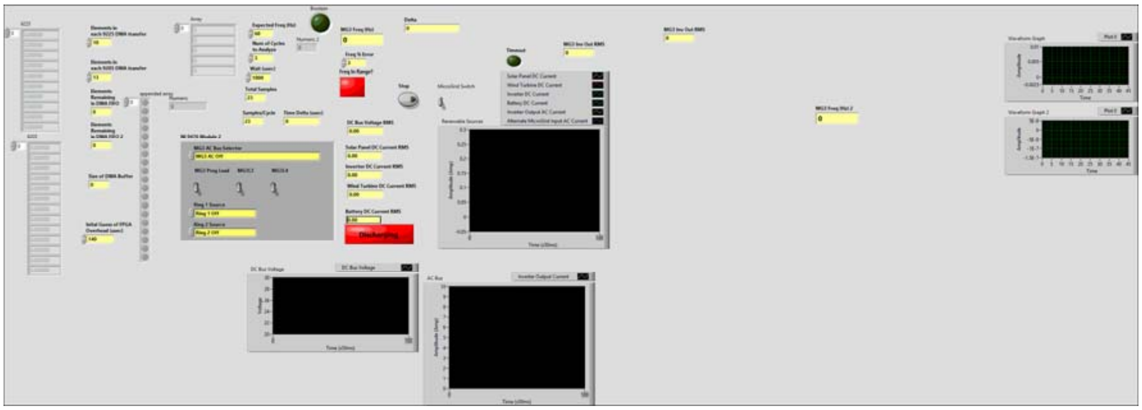
Further research areas may include an extended suite of testing situations. There is a need to create test suites for real life abnormal situations. These should include the energizing of large loads with high starting currents, switching of heavy loads, starting of large motors, and other variety of system faults. Also needed are some very deterministic ways to measure the performance of these state machine based systems as opposed to more traditional programming methods which are more procedural in nature.

Appendix A

Source Code Listing

MG3_cRIO_Main.vi





Stop

Elements in each 9225 DMA transfer

Wait (usec)

Freq % Error

Num of Cycles to Analyze

Inital Guess of FPGA Overhead (usec)

Expected Freq (Hz)

NI 9476 Module 2

- **MG3 AC Bus Selector**
- **MG3 Prog Load**
- **MG3L3**
- **MG3L4**
- **Ring 1 Source**
- **Ring 2 Source**

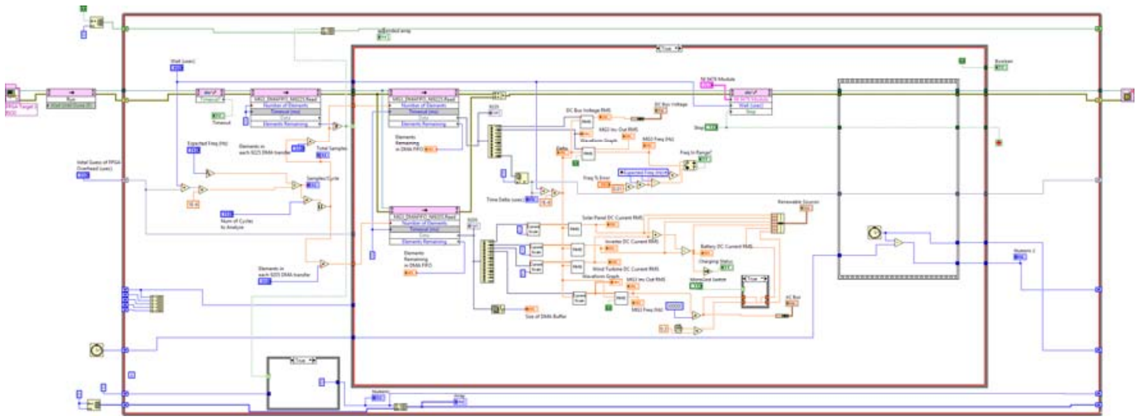
Elements in each 9205 DMA transfer

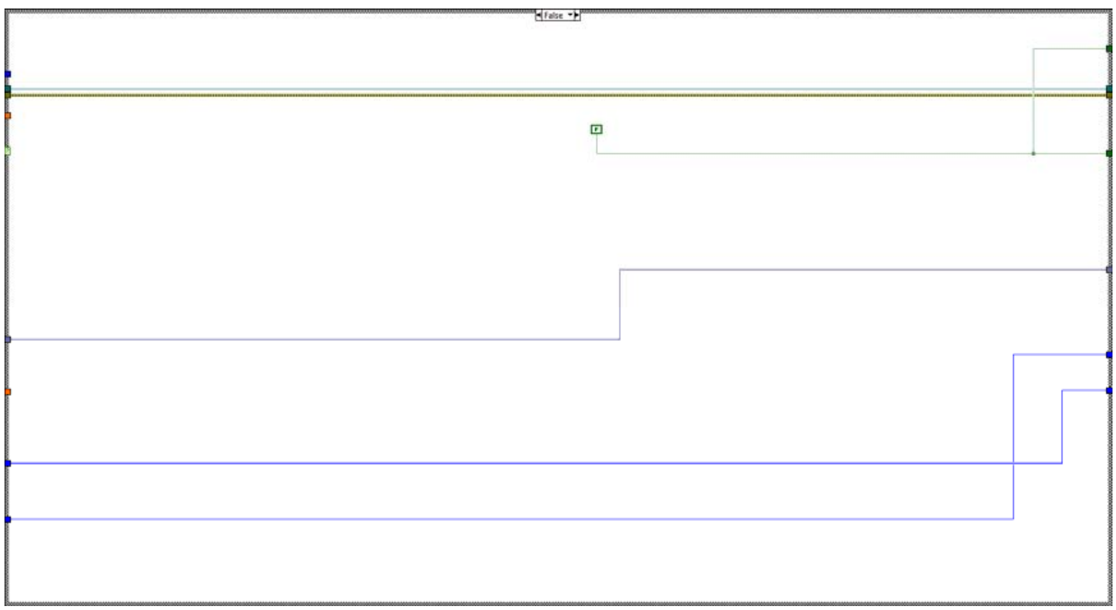
MicroGrid Switch

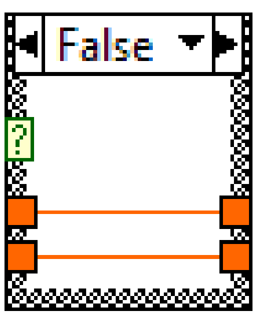
Elements Remaining in DMA FIFO

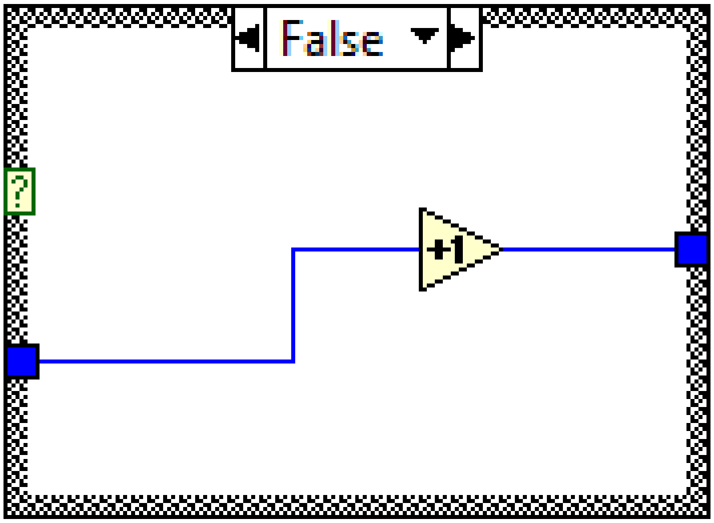
Size of DMA Buffer

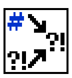
- TF Timeout
- DBL MG3 Freq (Hz)
- TF Freq In Range?
- DBL MG3 Inv Out RMS
- U32 Time Delta (usec)
- I32 Total Samples
- I32 Samples/Cycle
- DBL DC Bus Voltage RMS
- [DBL] Waveform Graph
- [FXP] 9205
    - FXP Numeric
- DBL Elements Remaining in DMA FIFO 2
- DBL Delta
- [FXP] 9225
    - FXP Numeric
- DBL Solar Panel DC Current RMS
- [DBL] Waveform Graph 2
- DBL MG3 Inv Out RMS
- DBL MG3 Freq (Hz) 2
- DBL Inverter DC Current RMS
- DBL Wind Turbine DC Current RMS
- DBL Battery DC Current RMS
- TF Charging Status
- 906 Renewable Sources
- I32 Numeric
- [I32] Array
    - I32 millisecond timer value
- TF Boolean
- U32 Numeric 2
- [TF] appended array
    - TF
- 906 AC Bus
- 906 DC Bus Voltage



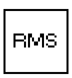

False


False



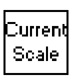

Error Cluster From Error Code.vi

C:\Program Files (x86)\National Instruments\LabVIEW 2012\vi.lib\Utility\error.llb\Error Cluster From Error Code.vi



CalculateRMS.vi

C:\Users\Greg\Documents\LabVIEW Data\gregT\SmartGrid\CalculateRMS.vi



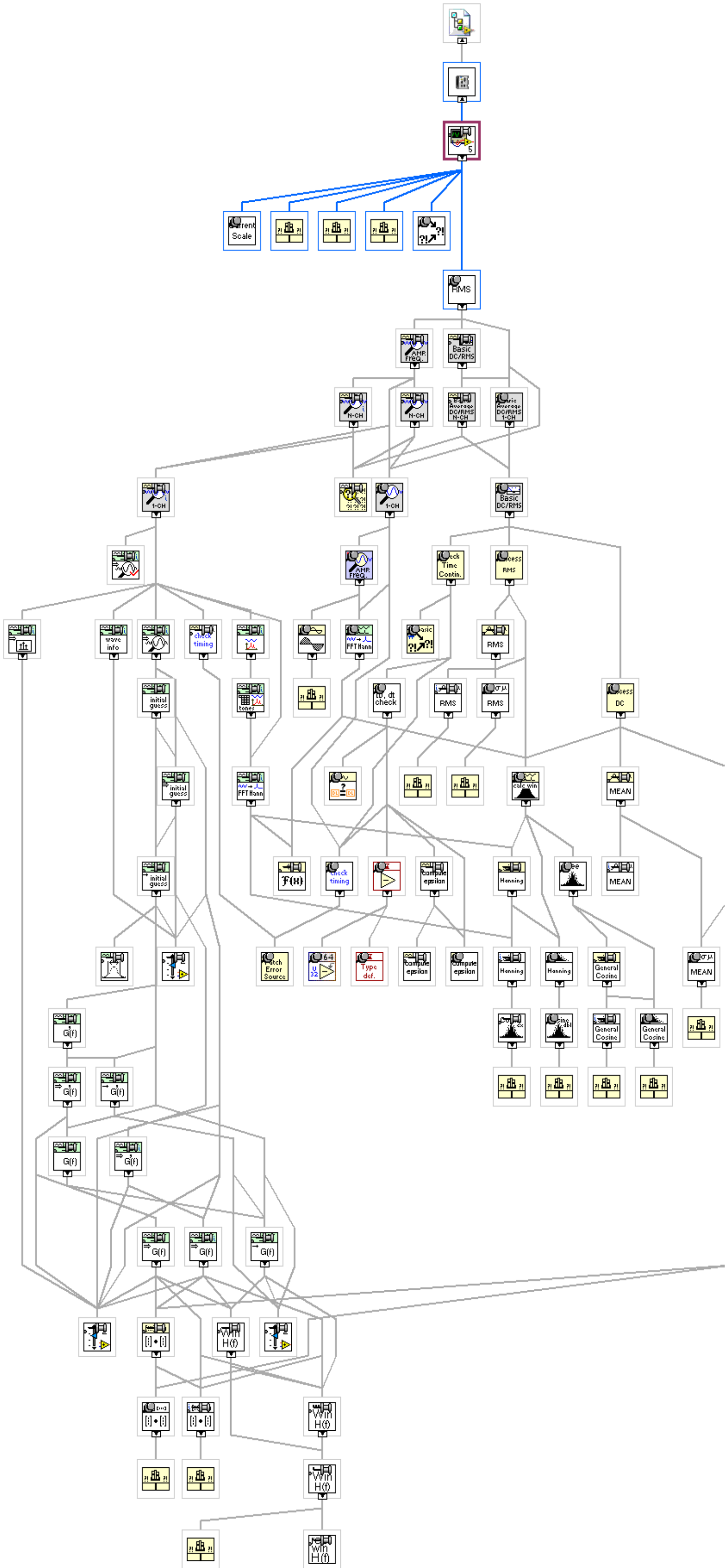
Current Scale.vi

C:\Users\Greg\Documents\LabVIEW Data\gregT\SmartGrid\Current Scale.vi

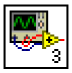
"MG3_cRIO_Main.vi History"

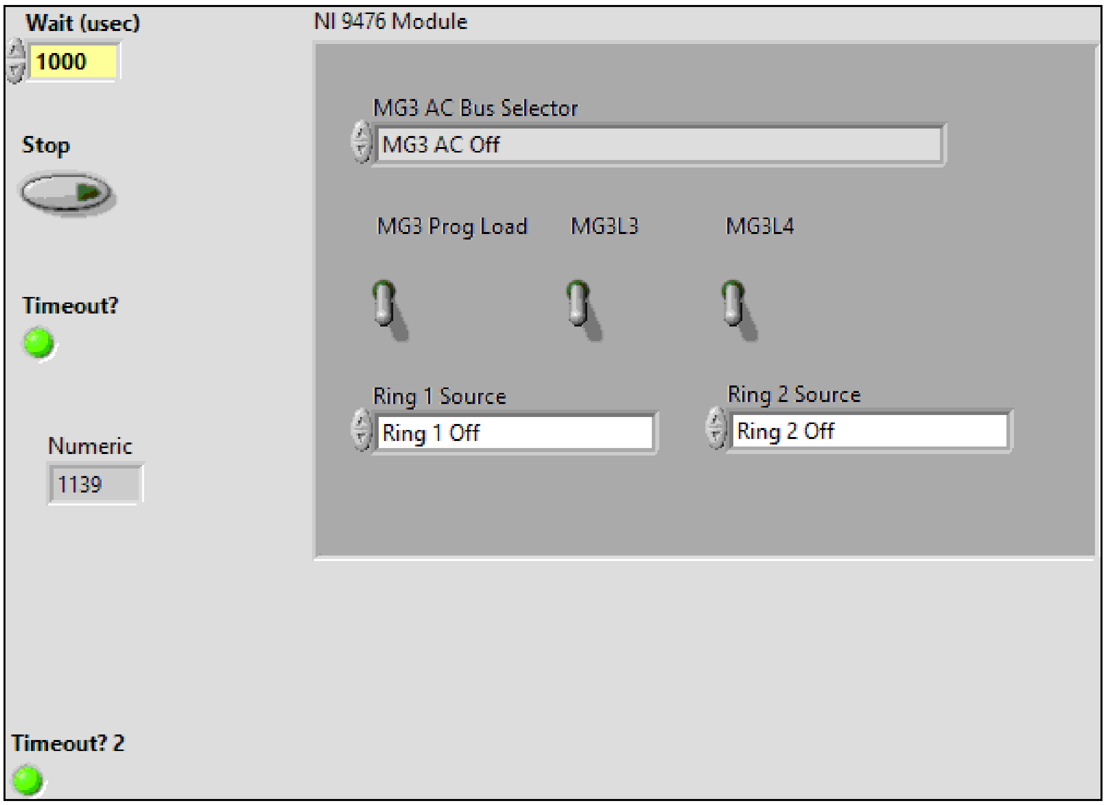
Current Revision: 372

Position in Hierarchy



MG3_FPGA_Main.vi





TF **Stop**

U32 **Wait (usec)**

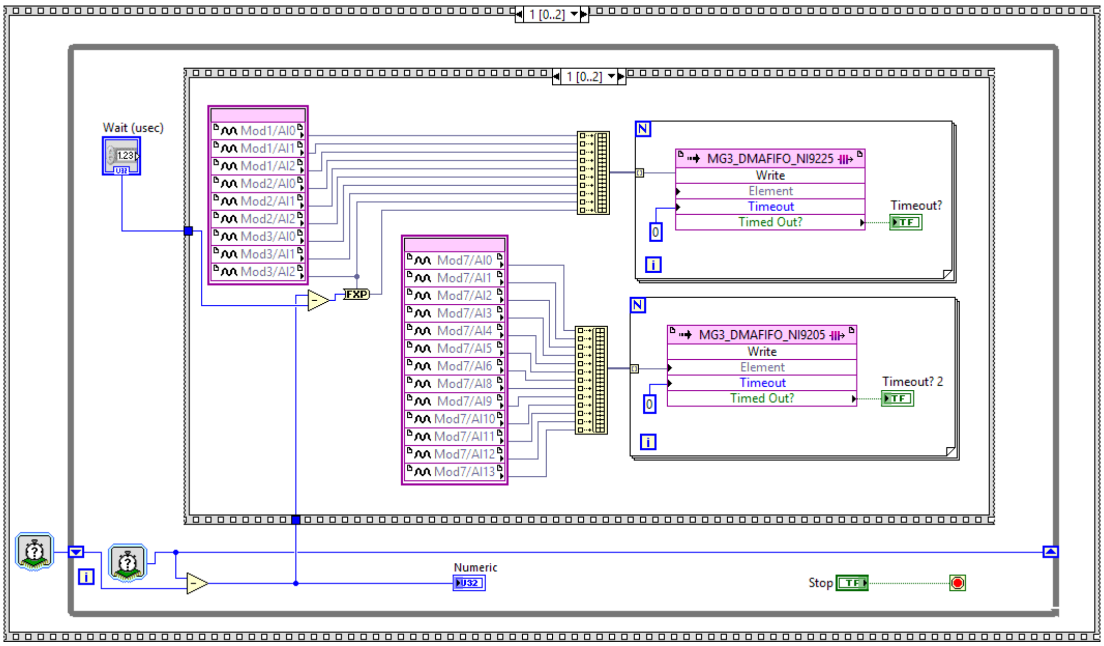
NI 9476 Module

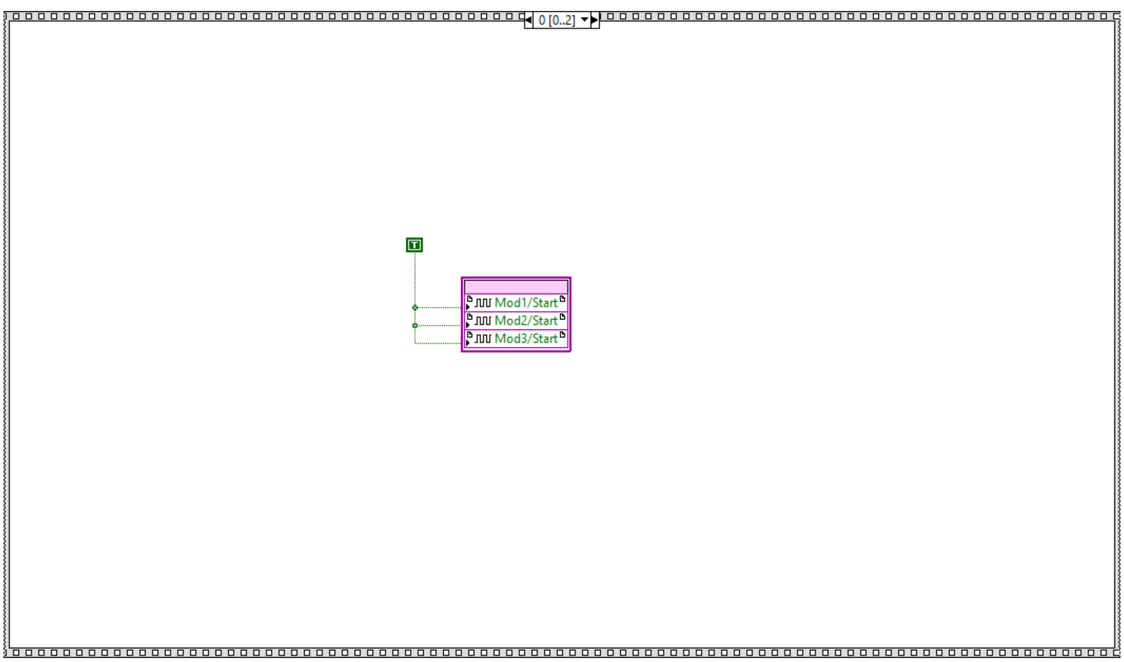
- **MG3 AC Bus Selector**
- **TF** **MG3 Prog Load**
- **TF** **MG3L3**
- **TF** **MG3L4**
- **Ring 1 Source**
- **Ring 2 Source**

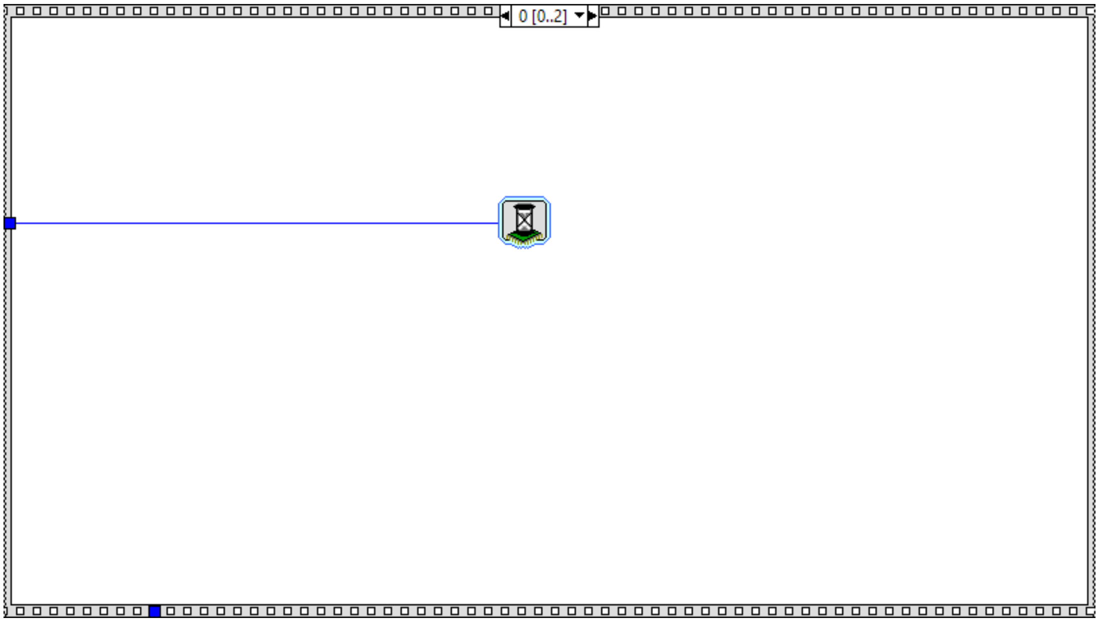
TF **Timeout?**

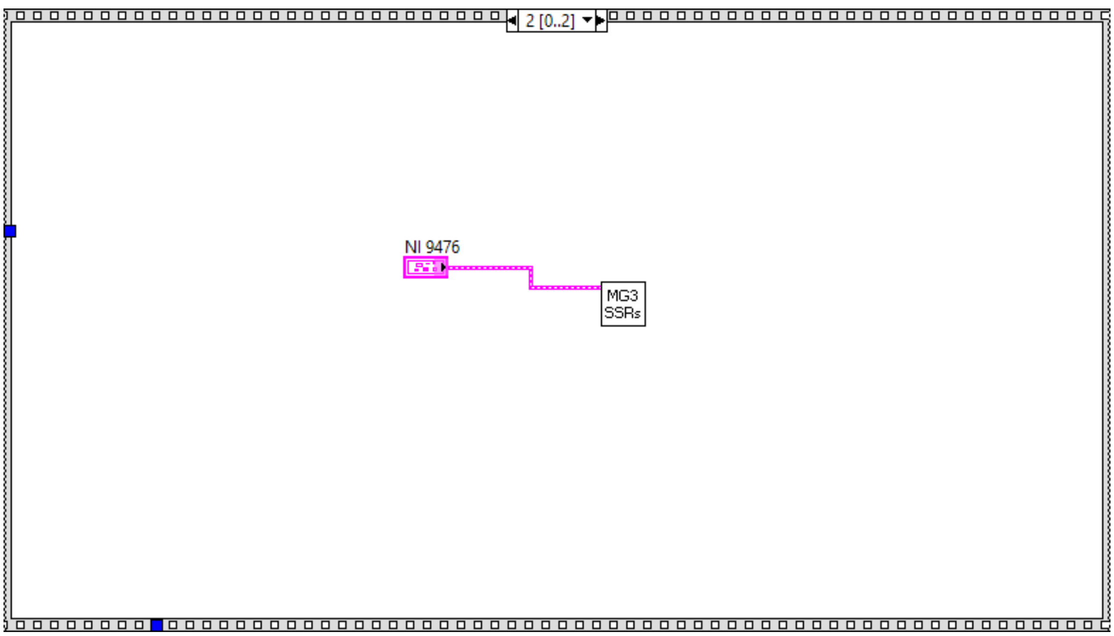
Timeout? 2

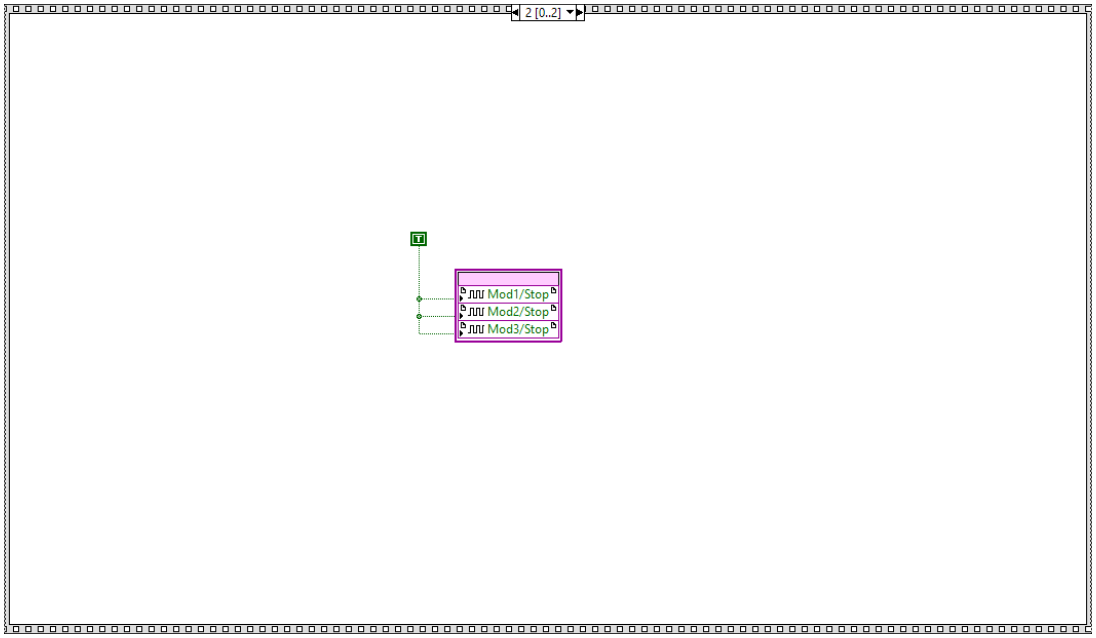
Numeric






0 [0..2]


2 [0..2]
NI 9476
MG3
SSRs



MG3_SSR.vi

C:\Users\Greg\Documents\LabVIEW Data\gregT\SmartGrid\MG3_SSR.vi

Wait

Wait

Delays for a certain time interval before the output data dependence becomes valid.

Tick Count

Tick Count

Returns the value of a free running counter in the units specified. The output and internal counter are both of the configured width.

Tick Count2

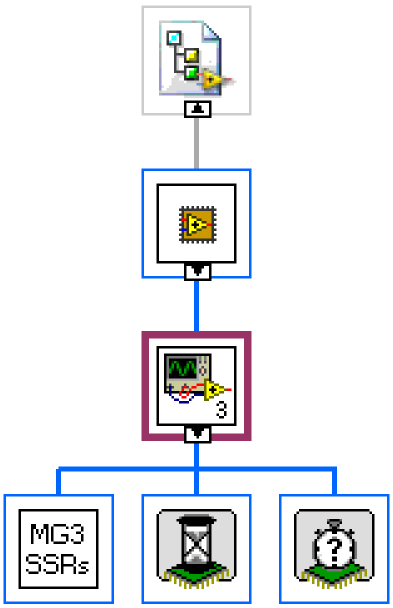
Tick Count

Returns the value of a free running counter in the units specified. The output and internal counter are both of the configured width.

"MG3_FPGA_Main.vi History"

Current Revision: 92

Position in Hierarchy



.

References

[1] M. El-Sharkawi, Electric Energy: An Introduction, Third ed., CRC Press, 2012, p. 606.

[2] J. E. Chadwick, "How a smarter grid could have prevented the 2003 U.S. cascading blackout," in *Power and Energy Conference*, Champaign, IL, 2013.

[3] D. Fessler, "Going Big on the Texas Power Problem," Investment U, 13 September 2011. [Online]. Available: http://www.investmentu.com/2011/September/going-big-texas-power-problem.html. [Accessed 1 January 2014].

[4] United States Department of Energy, "Smart Grid," [Online]. Available: http://energy.gov/oe/technology-development/smart-grid. [Accessed 31 January 2014].

[5] F. P. Sioshansi, Smart Grid, Academic Press, 2011, p. 568.

[6] A. Davoudi and A. Bidram, "Hierarchical Structure of Microgrids Control System," *IEEE TRANSACTIONS ON SMART GRID,* vol. 3, no. 4, p. 14, December 2012.

[7] 109th U.S. Congress, "Energy Policy Act of 2005," U.S. Government Printing Office, Washington, D.C., 2005.

[8] 110th U.S. Congress, "Energy Independence and Security Act," U.S. Government Printing Office, Washington, D.C., 2007.

[9] United States Department of Energy, "Smart Grid Investment Grant Program Progress Report," U.S. DoE, 2012.

[10] S. Bossart, "Success Stories in DoE's ARRA Smart Grid Program," 24 October 2013. [Online]. Available: http://www.netl.doe.gov/smartgrid/referenceshelf/articles/Success%20Stories%20in%20DOE.pdf. [Accessed 1 December 2013].

[11] Consortium for Electric Reliability Technology Solutions, "CERTS Microgrid Test Bed Demonstration with American Electric Power," 1 July 2013. [Online]. Available: http://certs.lbl.gov/certs-derkey-mgtb.html. [Accessed 1 December 2013].

[12] R. H. Lasseter and e. al., "CERTS Microgrid Laboratory Test Bed," *IEEE TRANSACTIONS ON POWER DELIVERY,* vol. 26, no. 1, p. 8, January 2011.

[13] Santa Clara University, "Sustainability at SCU," 1 July 2013. [Online]. Available: http://www.scu.edu/sustainability/stewardship/renewable.cfm. [Accessed 1 December 2013].

[14] Illinois Institute of Technology, "Microgrid at IIT," 1 Jul 2013. [Online]. Available: http://www.iitmicrogrid.net/. [Accessed 1 December 2013].

[15] U.S. Department of Energy, "National Energy Technology Laboratory," 1 July 2013. [Online]. Available: http://www.netl.doe.gov/. [Accessed 1 December 2013].

[16] IEEE, "IEEE Xplore Digital Library," Institute of Electrical and Electronics Engineers, 1 March 2104. [Online]. Available: http://ieeexplore.ieee.org/Xplore/home.jsp. [Accessed 1 March 2014].

[17] Safari Books Online, "Safari Books Online," O'Reilly Media, 1 March 2014. [Online]. Available: http://www.safaribooksonline.com/. [Accessed 1 March 2014].

[18] Z. Dongmei, Z. Nan and L. Yanhua, "Micro-grid connected/islanding operation based on wind and PV hybrid power system," in *Innovative Smart Grid Technologies*, Tianjin, 2012.

[19] P. van de Ven, N. Hegde, L. Massoulie and T. Salonidis, "Optimal Control of End-User Energy Storage," *Smart Grid, IEEE Transactions on,* vol. 4, no. 2, pp. pp.789,797, 2013.

[20] Z. Dan, W. Tong, W. Zhi, S. ZuYong and G. Liang, "An advanced control method of inverters in Microgrid," in *Advanced Power System Automation and Protection (APAP), 2011 International Conference on*, 2011.

[21] Z. Zhao, W. C. Lee, Y. Shin and K.-B. Song, "An Optimal Power Scheduling Method for Demand Response in Home Energy Management System," *Smart Grid, IEEE Transactions on* , vol. 4, no. 3, pp. pp.1391,1400, 2013.

[22] M. Ambia, A. Al-Durra and S. M. Muyeen, "Centralized power control strategy for AC-DC hybrid micro-grid system using multi-converter scheme," in *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society* , 2011.

[23] E. Bakirtzis and C. Demoulias, "Control of a micro grid supplied by renewable energy sources and storage batteries," in *Electrical Machines (ICEM), 2012 XXth International Conference on* , 2012.

[24] H. Dagdougui and R. Sacile, "Decentralized Control of the Power Flows in a Network of Smart Microgrids Modeled as a Team of Cooperative Agents," *Control Systems Technology, IEEE Transactions on,* vol. 22, no. 2, pp. pp.510,519, 2014.

[25] A. Bidram, A. Davoudi, F. Lewis and J. Guerrero, "Distributed Cooperative Secondary Control of Microgrids Using Feedback Linearization," *Power Systems, IEEE Transactions on,* vol. 28, no. 3, pp. pp.3462,3470, 2013.

[26] D. Koss, D. Bytschkow, P. Gupta, B. Schatz, F. Sellmayr and S. Bauereiss, "Establishing a smart grid node architecture and demonstrator in an office environment using the SOA approach," in *Software Engineering for the Smart Grid (SE4SG), 2012 International Workshop on*, 2012.

[27] T. Strasser, M. Stifter, F. Andren, D. Burnier de Castro and W. Hribernik, "Applying open standards and open source software for smart grid applications: Simulation of distributed intelligent control of power systems," in *Power and Energy Society General Meeting, 2011 IEEE*, 2011.

[28] B. Jurisic, N. Holjevac and B. Morvaj, "Framework for designing a smart grid testbed," in *Information & Communication Technology Electronics & Microelectronics (MIPRO), 2013 36th International Convention on*, 2013.

[29] M. Khalgui, O. Mosbahi, Z. Li and H. -M. Hanisch, "Reconfigurable Multiagent Embedded Control Systems: From Modeling to Implementation," *Computers, IEEE Transactions on,* vol. 60, no. 4, pp. pp.538,551, 2011.

[30] I. Gorton, Y. Liu and J. Yin, "GridOPTICS(TM): A design for plug-and-play smart grid software architecture," in *Software Engineering for the Smart Grid (SE4SG), 2012 International Workshop on* , Zurich, 2012.

[31] M. Khan and M. Iravani, "Supervisory Hybrid Control of a Micro Grid System," in *Electrical Power Conference, 2007. EPC 2007. IEEE Canada*, Montreal, Que, 2007.

[32] G. Hackenberg, M. Irlbeck, V. Koutsoumpas and D. Bytschkow, "Applying formal software engineering techniques to smart grids," in *Software Engineering for the Smart Grid (SE4SG), 2012 International Workshop on*, Zurich, 2012.

[33] M. Yin, Y. Xu and X. Li, "Design of Virtual Instrument Software System Based on Nested State Machine," in *Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on*, Wuhan, 2009.

[34] J.-P. Knauss, C. Warren and D. Kearns, "An innovative approach to smart automation testing at National Grid," in *Transmission and Distribution Conference and Exposition (T&D), 2012 IEEE PES* , Orlando, FL, 2012.

[35] A. Etemadi, E. Davison and R. Iravani, "A Decentralized Robust Control Strategy for Multi-DER Microgrids—Part I: Fundamental Concepts," *Power Delivery, IEEE Transactions on,* vol. 27, no. 4, pp. pp.1843,1853, 2012.

[36] A. Etemadi, E. Davison and R. Iravani, "A Decentralized Robust Control Strategy for Multi-DER Microgrids—Part II: Performance Evaluation,"

Power Delivery, IEEE Transactions on , vol. 27, no. 4, pp. pp.1854,1861, 2012.

[37] C. Marnay, N. DeForest and J. Lai, "A green prison: The Santa Rita Jail campus microgrid," in *Power and Energy Society General Meeting, 2012 IEEE*, San Diego, CA, 2012.

[38] A. Mathur and R. Krishnan, "A novel approach to micro-grid control using DAQ interface," in *Energy Efficient Technologies for Sustainability (ICEETS), 2013 International Conference on* , Nagercoil, 2013.

[39] H. Keshtkar, J. Solanki and S. Solanki, "Application of PHEV in load frequency problem of a hybrid microgrid," in *North American Power Symposium (NAPS)*, Champaign, IL, 2012.

[40] C. Hill, M. Such, D. Chen, J. Gonzalez and W. Grady, "Battery Energy Storage for Enabling Integration of Distributed Solar Power Generation," *Smart Grid, IEEE Transactions on* , vol. 3, no. 2, pp. pp.850,857, 2012.

[41] A. Flueck and C. Nguyen, "Integrating Renewable and Distributed resources - IIT Perfect Power Smart Grid Prototype," in *Power and Energy Society General Meeting, 2010 IEEE*, Minneapolis, MN, 2010.

[42] R. Tinder, Asynchronous Sequential Machine Design and Analysis, Morgan & Claypool Publishers, 2009.

[43] M. Mozgovoy, Algorithms, Languages, Automata, & Compilers: A Practical Approach, Jones & Bartlett Learning, 2009, p. 345.

[44] S. Kandar, Automata Theory and Formal Languages, Pearson Education, 2012, p. 376.

[45] P. Vasant, Meta-Heuristics Optimization Algorithms in Engineering, Business, Economics, and Finance, IGI Global, 2012, p. 734.

[46] M. Anvaari, D. Cruzes and R. Conradi, "Smart Grid software applications as an ultra-large-scale system: Challenges for evolution," in *Innovative Smart Grid Technologies (ISGT) 2012 IEEE PES*, Washington, D.C., 2012.

Biographical Information

Greg Turner is a graduate of the University of Texas at Arlington (UTA) Electrical Engineering program. After receiving a Bachelor's degree in Electrical Engineering, he went to work for the Cellular Infrastructure Division of Motorola Inc. As a working engineering he established a career in software development for numerous companies including National Semiconductor, Peavey and Texas Instruments. He has specific expertise in developing automated test software and embedded system software. In his most recent position at Texas Instruments he was part of the applications team developing security and cryptography software for the Linux software development kit that enables the Sitara™ line of ARM system-on-a-chip microprocessors.

Combining his software development skills with a desire to work in the field of renewable energy, he began working for the Microgrid lab at UTA. With the hardware for the Microgrid in place he came in at a time when a software control application was needed to get the Microgrid working efficiently.