Low-cost SCADA platforms for a solar energy system

By

Ibrahim Allafi

A Thesis submitted to the School of Graduate Studies in partial fulfillment

of the requirements for the degree of MASTER OF ENGINEERING

Faculty of Engineering and Applied Science

Memorial University of Newfoundland

November 2017

St. John's

Newfoundland

Canada

Abstract

Conventional energy has been used from ancient times. Coal, natural gas, oil, uranium, and firewood are example of conventional source. These sources are consider the main source of air pollution. So, it is desired to use clean energy sources such as solar power, wind turbine, and hydro-electric power. We need to monitor and control parameters of clean energy sources when they are working as a power source.

In this research, few low-cost Supervisory Control and Data Acquisition (SCADA) systems have been designed for a small photovoltaic system to save data in a text file and show it on monitor screen as a number or graph. The proposed system is designed using four methods. The first method, is based on low cost sensors, Arduino Uno, and Raspberry pi. These components are connected together to send data to web server on the internet (www.ubidots.com). Second method, is based on three things: low cost sensors, Arduino Uno, and free Reliance SCADA software. In this method, the data is sent to user interface -which are created by Reliance Software- from Arduino by using mini USB. Third method, is based on a low-cost ESP32, SD card reader, Wi-Fi, and sensors. The purpose of SD card is to save data as text file which come from sensors by ESP32, and it saves on a web page as well. Wi Fi of ESP32 is used to access to the file of data using a computer, a tablet, or cell phone and it could be download easily. Last method, is based on low cost sensors and Arduino LoRa. In this method, the data of photovoltaic system has been sent to a control room which is located 1km from photovoltaic system location. LoRa is used to send the data instead of mini USB and Wi Fi. This thesis, which describes the details of all the four SCADA system for a PV system at MUN, presents a comparison among them.

ACKNOWLEDGEMENTS

The author would like to express profound gratitude to his supervisor Dr. Tariq Iqbal for his supervision, persistent support and encouragement throughout the research work. The author is grateful to him for sharing his vast experience and knowledge over past two years. His moral support and guidance enabled the author to complete the research successfully.

I owe a very important debt to my family, especially my parents, and my brothers and sisters for their encouragement and both mental and financial support during my graduate studies.

The authors also would like to thank the Libyan government for providing graduate student funding for this research, and Elmergib University in Al Al Khums, Libya that nominated me to get master degree in Electrical Engineering.

Contents

| Abstract | ii |
|--|-----------|
| ACKNOWLEDGEMENTS | iii |
| List of Tables | ix |
| List of Figures | X |
| Chapter 1: Introduction and literature review | 1 |
| 1.1 Introduction to PV System and the SCADA Platform | 1 |
| 1.2 Importance of Automating PV Systems | 4 |
| 1.3 SCADA Integration Methods | 6 |
| 1.4 SCADA Software, Algorithms, and Hardware | 14 |
| 1.5 Cost Efficiency of SCADA-Based Monitoring and Control Systems | 21 |
| 1.6 Thesis Objectives | 24 |
| 1.7 Thesis Organization | 26 |
| 1.8 References | 28 |
| Chapter 2: Design an Open low-cost SCADA system for Monitoring a Sol | ar Energy |
| Installation | 34 |
| 2.1 Introduction | 35 |
| 2.2 Literature Review | 36 |
| 2.3 System Architecture | |

| 2.4 Hardware Design | 40 |
|--|-----------|
| 2.4.1 Arduino Uno Microcontroller | 41 |
| 2.4.2 Raspberry-pi | 42 |
| 2.4.3 Current Sensor | 44 |
| 2.4.4 Voltage Sensor | 45 |
| 2.5 Software | 46 |
| 2.6 Hardware Setup | 47 |
| 2.7 Results and Discussion | 48 |
| 2.8 Conclusion | 53 |
| 2.9 Acknowledgment | 53 |
| 2.10 References | 54 |
| Chapter 3: Low-Cost SCADA System Using Arduino and Reliance SC. | ADA for a |
| Standalone Photovoltaic System | 56 |
| 3.1 Introduction | 57 |
| 3.2 Hardware Design | 59 |
| 3.2.1 Arduino Uno Microcontroller | 61 |
| 3.2.2 Current Sensors | 63 |
| 3.2.3 Voltage Sensors | 64 |
| 3.3 Hardware Setup | 64 |

| 3.4 Software Requirements | |
|---|--------------------------|
| 3.4.1 Arduino IDE | 66 |
| 3.4.2 Reliance SCADA | 71 |
| 3.5 Communication System | 72 |
| 3.6 Cost of the SCADA System | 72 |
| 3.7 Result and Discussion | 73 |
| 3.8 Conclusion | 78 |
| 3.9 Acknowledgment | |
| 3.10 References | |
| Chapter 4: Design and Implementation of a Low Cost We | b Server Using ESP32 for |
| Real-Time Photovoltaic System Monitoring | 80 |
| | |
| Abstract | 80 |
| Abstract | |
| Abstract 4.1 Introduction 4.2 PV System Description | |
| Abstract 4.1 Introduction 4.2 PV System Description 4.3 System Design | |
| Abstract 4.1 Introduction 4.2 PV System Description 4.3 System Design 4.3.1 Sparkfun ESP32 Thing | |
| Abstract 4.1 Introduction 4.2 PV System Description 4.3 System Design 4.3.1 Sparkfun ESP32 Thing 4.3.2 SD CARD Reader | |
| Abstract 4.1 Introduction 4.2 PV System Description 4.3 System Design 4.3 System Design 4.3.1 Sparkfun ESP32 Thing 4.3.2 SD CARD Reader 4.3.3 Wi-Fi | |

| 4.4 Hardware Setup | 91 |
|--|---|
| 4.5 Software Used | 92 |
| 4.5.1 Arduino IDE | 92 |
| 4.5.2 HTML Script | 93 |
| 4.6 Implementation and Result | 93 |
| 4.8 Power Needs for Design | 98 |
| 4.9 Conclusion | |
| 4.10 Acknowledgment | |
| 4.11 References | |
| | |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f | or Small |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa | or Small 102 |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa | or Small 102 103 |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa | or Small 102 103 105 |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa 5.1 Introduction 5.2 PV System 5.3 Hardware Configuration | or Small 102 103 105 107 |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa 5.1 Introduction 5.2 PV System 5.3 Hardware Configuration 5.4 Transition Station design | or Small 102 103 105 107 110 |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa 5.1 Introduction 5.2 PV System 5.3 Hardware Configuration 5.4 Transition Station design 5.4.1 Arduino Lora Transmitter (ALT) | or Small 102 103 105 107 110 110 |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa 5.1 Introduction 5.2 PV System 5.3 Hardware Configuration 5.4 Transition Station design 5.4.1 Arduino Lora Transmitter (ALT) 5.4.2 Sensors | or Small 102 103 103 105 107 110 110 112 |
| Chapter 5: A Low-cost Monitoring and Data Acquisition System f Photovoltaic System Based on Arduino LoRa 5.1 Introduction 5.2 PV System 5.3 Hardware Configuration 5.4 Transition Station design 5.4.1 Arduino Lora Transmitter (ALT) 5.4.2 Sensors 5.5 Hardware Setup | or Small 102 103 103 105 107 110 110 112 113 |

| | 5.7 Powering Needs for This Design | 114 |
|---|---|-----|
| | 5.8 Software Used | 115 |
| | 5.8.1 Arduino IDE | 115 |
| | 5.8.2 PUTTY | 116 |
| | 5.9 Range Testing | 116 |
| | 5.10 Cost Design | 118 |
| | 5.11 Result and Discussion | 120 |
| | 5.12 Conclusion | 124 |
| | 5.13 Acknowledgment | 124 |
| | 5.14 References | 125 |
| (| Chapter 6: Conclusions and Future Works | 126 |
| | 6.1 Conclusion | 126 |
| | 6.2 Future Works | 127 |

List of Tables

| Table 1. 1: Remote Site Hardware Specifications | |
|---|-----|
| Table 1. 2: Data Center Hardware Specifications | 20 |
| Table 1. 3: Brief Descriptions of Software Used | |
| Table 2. 1: Specifications of Arduino board | 42 |
| Table 2. 2 MEASURMENT ERROR OF SENSORS SYSTEM | 49 |
| Table 3. 1: Specifications of Arduino board | 62 |
| | |
| Table 3. 2: Allocation of MODBUS Address for MODBUS RTU | 72 |
| Table 3. 3: Allocation of MODBUS Address for MODBUS RTU | 73 |
| Table 3. 4: MEASUREMENT ERRORS OF SENSOR SYSTEM | 74 |
| | |
| Table 4. 1: SPECIFICATIONS OF ESP32 | 86 |
| Table 4. 2: COST design web server | |
| Table 4. 3: CURRENT AND VOLTAGE DRAWN | |
| | |
| Table 5. 1: CURRENT AND VOLTAGE DRAWN | 115 |
| Table 5. 2: Coordinates of ALT and ALR | 117 |
| Table 5. 3: Cost Monitoring and Data Acquisition System | 119 |

List of Figures

| Figure 1.1: Figure 2An illustration of a SCADA platform in relation to the elements of |
|--|
| the PV system from |
| Figure 1.2: An illustration an automated PV system for fish cage cooling |
| Figure 1.3: An illustration of a user interface of a SCADA used by the PV Plant7 |
| Figure 1.4: Illustration of an architectural design for an automated PV system which |
| utilizes a SCADA platform and a master controller devise10 |
| Figure 1.5: A Closed-loop control system schema12 |
| Figure 1.6: Schema of the implementation of a closed-loop control system14 |
| Figure 1.7: Basic algorithm for a PV system utilizing SCADA and HMI as main data |
| processing elements [14]17 |
| Figure 1. 8: Illustration of a low-cost SCADA system for small PV system22 |
| Figure 1.9: Block diagram of the pins used of the AVR ATmega853523 |
| Figure 2. 1: SCADA Architecture |
| Figure 2. 2: Hardware Configuration of SCADA System |
| Figure 2. 3: Arduino Uno41 |
| Figure 2. 4: Raspberry-pi43 |
| Figure 2. 5: Connection Drawing |
| Figure 2. 6: Voltage Divider for voltage of PV45 |
| Figure 2. 7: Voltage Divider for voltage of Battery46 |
| Figure 2. 8: Some Hardware of the system |

| Figure 2. 9: Photo of Communication of Hardware and Software | 48 |
|--|------------|
| Figure 2. 10: Front Screen of SCADA System | 50 |
| Figure 2. 11: Efficiency of Solar energy system | 50 |
| Figure 2. 12: Current of PV against Time | 51 |
| Figure 2. 13: Voltage of PV against Time | 51 |
| Figure 2. 14: Current of Battery against Time | 52 |
| Figure 2. 15:Voltage of Battery against Time | 52 |
| Figure 2. 16: Power before MPPT against Time | 53 |
| Figure 3. 1: Solar panels on the roof of engineering building | 50 |
| Figure 3. 2: Hardware configuration of SCADA system | 51 |
| Figure 3. 3: Arduino Uno R3 | 52 |
| Figure 3. 4: Connection drawing of current sensor | 53 |
| Figure 3. 5 Connection Drawing of voltage sensor | <u></u> 54 |
| Figure 3. 6 Hardware setup of SCADA system | <u> </u> |
| Figure 3. 7: Define Address MODBUS on Reliance Software | 70 |
| Figure 3. 8: Shows writing addresses registers on Reliance SCADA | 70 |
| Figure 3. 9: User interface of SCADA system | 71 |
| Figure 3. 10: User interface of SCADA while running | 75 |
| Figure 3. 11 Data saved in Excel file | 77 |
| Figure 3. 12: Efficiency of MPPT | 77 |
| Figure 4. 1: Server architecture | 33 |

| Figure 4. 2: Proposed PV system connection | 84 |
|---|-----|
| Figure 4. 3: ESP32 sparkfun thing | 85 |
| Figure 4. 4: ESP32 connected to SD card reader | 87 |
| Figure 4. 5: SD card and ESP32 connection | 87 |
| Figure 4. 6 :Voltage divider connection | 89 |
| Figure 4. 7: ACS 712 current sensor and pull-down resistor connection | 90 |
| Figure 4. 8: Hardware setup connection | 91 |
| Figure 4. 9: ESP32, SD card and sensor connections | 92 |
| Figure 4. 10: Serial monitor of IDE a test run | 94 |
| Figure 4. 11: Different Internet browsers showing web page | 95 |
| Figure 4. 12: Main web page for data download | 95 |
| Figure 4. 13: Downloading text file containing data | 96 |
| Figure 4. 14: Text in the data file4.7 Cost the Web Server | 97 |
| Figure 5. 1: Connection diagram | 106 |
| Figure 5. 2: PV system installation | 107 |
| Figure 5. 3: Connection diagram of proposed system | 108 |
| Figure 5. 4: PV location and control room on map (1Km distance) | 109 |
| Figure 5. 5: Arduino Lora Transmitter | 111 |
| Figure 5. 6: Hardware setup connection | 113 |
| Figure 5. 7: Paths between ALT at Signal Hill and ALR at different points | 116 |
| Figure 5. 8: Putty settings | 121 |

| Figure 5. 9 Serial monitor with data | 122 |
|--------------------------------------|-----|
| Figure 5. 10: PV data in text file | 123 |

Chapter 1

Introduction and literature review

1.1 Introduction to PV System and the SCADA Platform

Solar energy is one of the most reliable alternative energy sources due to its many advantages. Such advantages include simple installation of the photo voltaic (PV) system, high reliability, low maintenance cost, and lack of noise because of the absence continuously moving parts in a PV system [1] [2]. The reliability of solar energy is the reason why diverse institutions all over the world have engaged in the manufacture and installation of PV systems. It was estimated in 2010 that approximately 310 GW of annual energy production will be reached by 2016; current estimates are even higher. In line with the growing demand for PV systems, efforts to further increase their efficiency in terms of energy storage and distribution are being conducted. The current PV systems have diverse limitations that inventors and scientists continuously try to address [3]. PV system manufacturers are exerting efforts to optimize the systems electricity production and process reliability. Such efforts require PV plants to increase their electricity production rate, availability & lifespan of PV system elements, and reduce their maintenance & operating cost [4].

For large and small scale production of electricity from solar energy, one of the most promising improvements in the PV system that is being widely developed is the automation of the entire process, which includes energy acquisition, tracking, and distribution. Such PV systems are called "PV on-time systems" since they require ontime monitoring and control [3] [5]. PV on-time systems require diverse components or elements. One of the most important components is the Supervisory Control and Data Acquisition (SCADA) platform [3].

It should be noted that there are other platforms that are currently being used for PV systems, which include cloud-based or web-based monitoring platforms. In [6], it was shown that a cloud-based SCADA system can also be used for storing, processing, and real-time publishing of PV systems data. The said system has all its algorithms and software rented from a cloud service provider [6]. There are also some studies that integrated satellite tracking technologies in their SCADA system; one example of such study was conducted [7]. In [8], it was explained that, for PV systems that have more than 10MW capacity, the best platform to use is SCADA and not web-based platforms. The reason for this is that SCADA is specifically built for real-time monitoring and control, while web-based platforms are usually built for monitoring purposes only [8].

As its name suggests, the SCADA platform is responsible for acquiring information about all the elements of the PV system. The SCADA platform, together with a substation controller, is responsible for the monitoring and control of all PV elements. These functions of the SCADA platform are illustrated in figure 1.1. It should be observed in the figure that SCADA is at the monitoring level, which is the heart of PV on-time systems. SCADA allows the gathering and processing of information from all the elements of a PV system. Based on the processed information, the SCADA system sends commands to the different parts of the PV system, especially when errors are detected. Such actions include the automated movement of the solar panels towards the direction of the sun to optimize sunlight exposure [3].



Figure 1.2:An illustration of a SCADA platform in relation to the elements of the PV system from [3]

The elements shown in figure 1.2 include: DC - AC inverter, SPA – Smart Power Quality Analyzer, RSYN - Synchronism-check, QVTR - Voltage Transient; QVVR - Voltage Variation, QITR - Current Transient, QFVR - Frequency Variation, MMXU -Measurement, MMTR - Metering 3 Phase, MMET - Meteorological Information, MMDC - DC Measurement, DPVM - PV Module ratings, DPVA - PV Array characteristics, and SCADA - Supervisory Control and Data Acquisition [3].

1.2 Importance of Automating PV Systems

The use of SCADA platform in a PV system is usually for automation and data logging purposes. Automation is very important in improving the efficiency and overall lifetime of a PV system. The current small PV systems usually use dyes and other metal alloys gather energy from the sun. The light coming from the sun is composed of diverse electromagnetic energies of different wavelengths and frequencies, which means that they also have different effects to the dye and the alloy of the PV system [9].

In general, dyes and alloys lose their capacity to gather energy as temperature increases. Ideally, the energy that comes from sunlight is all converted to electrical energy upon hitting the solar panel surface. In reality, however, a significant amount of the occurrence sunlight is converted to heat. Some of this heat is dissipated into the atmosphere – hence, producing an overall increase in local environmental temperature – while some of the heat stays in the solar panels [9]. At a certain temperature, the dye in the solar panels disintegrates and the alloys widen their band-gap, which ultimately results in the disruption of the entire energy conversion process. Depending on the dye and alloys that are used, the solar panels could acquire permanent damage due to accumulated heat [10]. Due to this problem with heat accumulation, modern day PV systems are equipped with temperature monitoring devices [11].

SCADA platforms have also been used, not only to monitor the temperature of the solar panels, but also in controlling the release of stored electrical energy. Note that all PV systems contain batteries to which the converted electrical energy is stored. The release of this stored energy is usually done manually or continuously in older PV systems. The introduction of automation through SCADA applications has allowed the automated release of electric energy. One of the industries that use automated electrical energy release is the fishing industry. In [12] an automated PV system was used in maintain the temperature of fish cages. The PV system that they developed included a SCADA application that utilizes a Programmable Logic Controller (PLC) device and Human Machine Interface (HMI) software [12].

In [12] PV system, the SCADA application was not used to monitor the temperature of the solar panels, but the temperature of water in the fish cages. According to the study, the fish cages should be maintained at 17^{0} C. Due to the heat from the sun and body heat of the fish cultures, the temperature of the water within the fish cages could reach method above 17^{0} C. Hence, the PV system had its temperature detection node immersed in the fish cages to monitor water temperature. Whenever the node registers a certain level of temperature that is higher than 17^{0} C, the PLC controller activates the release of electricity from the PV system into the water pump, which then draws cold water from a well into the fish cages to return the temperature to 17^{0} C. An illustration of the automated PV system is shown in the figure 1.3. It can be observed in the illustration that the center piece of the automated PV system is the PLC, which was made to run by the HMI software [12].



Figure 1.3: An illustration an automated PV system for fish cage cooling [12]

The illustration of the small PV system in figure 1.3 is relatively simple compared to other types of PV systems. Other PV systems integrate wireless technologies, and other web-based applications. Nevertheless, although the integration of SCADA devices and software vary for different PV systems, the basic principle of PV automation through SCADA applications is shown in the said figure.

1.3 SCADA Integration Methods

There are different methods of integrating the SCADA platform in a PV on-time system. One of these methods is shown in the figure 1.3. The method of integration involves integrating the SCADA platform through cRIO programmable controllers, which include those that are used for time synchronization. A cRIO programmable controller may be installed in the Transformer Center (TC) to monitor energy and energy production. In figure 1.4, the SCADA is able to monitor the energy distribution and production information. SCADA also expresses such information in terms of percentage of the nominal power of the PV modules that are directly connected to the DC (AC converters). The SCADA can also be connected to the GPS system and weather monitoring devices [3]. Note that the cRIO programmable controllers can also be integrated to web-based applications for long distance monitoring and data acquisition [13].



Figure 1.4: An illustration of a user interface of a SCADA used by the PV Plant [3]

The SCADA user interface in figure 1.4 is set at the "Diagram" tab. The display shows that the SCADA platform regularly analyzes the status of the PV system's elements or devices. In this particular tab, the status of each element is indicated by color codes: green means working or executing element, yellow for loss of synchronization or alarm, red for the loss of communication, and gray for disconnected elements. Such SCADA platforms are also applicable for small PV systems [3].

Another method of integrating SCADA platform into a PV system is by using it as a center for processing information. In [14], it was explained that one of the challenges of PV systems is when they need to be integrated into existing electrical networks. This challenge is common in the United States (US), because diverse state governments have mandated small and large PV power plants to simply integrate their own PV systems to the existing electrical networks rather than creating new ones. This means that the PV power plants must be able to generate electricity that will meet the parameters for the efficient distribution of power across the existing electrical networks [14]. It should be noted that the mandate of the US government to integrate PV systems to existing electrical networks is perceived to continue despite the challenges faced by PV system owners because the US is aiming at creating a smart grid system that would also integrate other renewable energy sources [15].

In [14] further noted that the fluctuating amount of electricity that is generated through PV systems poses a major problem to efficient power distribution. There are sub-systems in the existing electrical networks. Each of these sub-systems can only function properly

with a steady flow of electricity from the main PV systems. The automated PV system, therefore, must be able to solve the problem with the fluctuating amount of electricity. One efficient method of doing this is to have a PV system that is architecturally designed to have a specialized master or central control system other than the SCADA. In such an architectural design, the SCADA is used as a rallying point for the data generated from the existing sub-systems [14].

The master control system is the one that decides or the one that interprets the data processed by the SCADA platform. This architectural design is shown in the figure 1.4. Based from the architectural design, the SCADA platform, together with HMI, is directly linked to the master controller. The master controller, on the other hand, is connected to diverse devices and chips. Some of these devices are the sensing devices which include meters, input/output (I/O) devices, microprocessor-based protective relays, alarm systems, and solar trackers. The master controller is also connected to equipment controllers, which include inverters and solar trackers that could support bi-directional communications. Other systems and devices that must be connected to the master forecast systems, text messaging & emailing systems, power & energy management systems, and generators [14].



Figure 1.5: Illustration of an architectural design for an automated PV system which utilizes a SCADA platform and a master controller devise [14]

It should be noted that SCADA and HMI are both involved in the control function of the architectural design. Since present day PV systems can be used either in tandem with preexisting electrical networks or customized electrical networks, it is important for engineers make sure that both SCADA and HMI are capable of coordinated and local control functions. These functions are possible if the PV system has an open-loop control. Based from the architectural design, the set point can be moved from local SCADA and HMI system to centralized SCADA and HMI system within an open-loop schema. The algorithm for this schema receives the signals for the set-point changes or movements that fall within an acceptable range, and then sends the control variable or the output control signal to the target device through specified communication protocols. In doing this, the PV system must make sure that the SCADA and HMI system is in control in accordance with the remote or local switch indicator. The switch makes it possible to make sure that will be no conflict between the SCADA and HMI; it also makes sure that the entire PV system is in safety mode whenever there are personnel that are working on the PV plant site. There are several open-loop controls, which include the limiters for inverters, power factor setters for inverters, and locators or positioning equipment at certain angles – such locators are essential elements of the solar tracker. Note that open-loop controls often use low-speed communication protocols over high speed communication protocols because the response time has less significance [14].

A SCADA and HMI system can also be integrated into a PV system through a closedcontrol scheme as shown in figure 1.5. In the said figure, the system contains three major elements which include: the process, which can be the PV plant; the sensing device, which provides the information processing capability; and the controller, which controls the entire system and to which all other elements integrate [16]. The main role of the master controller is to compare the desired set point to the measured process variable. In other words, it compares the actual measured variables with a standard value. The controller also calculates a third variable in accordance to the difference between the standard and the actual values. The third variable is then sent to an end device which performs a function. Note that, the difference between the standard and the actual value is referred to as the error; what the end device does, therefore, is to correct this error [14]. The closed-loop system has several advantages compared to the open-loop system. Some of these advantages include increased response speed, minimization of noises from disturbance, reduced sensitivity caused by modeling errors, and reduced error. Such SCADA and HMI system integration schema is best suited for PV systems that are highly affected by wind speeds. Note that one of the main factors that should be considered in the construction of PV systems is the availability of sunlight. Interestingly, due to high levels of sunlight on the majority of selected areas, there is also a relatively higher wind velocity in them. Hence, diverse PV systems integrate a wind speed monitoring device. For such PV systems it is important to integrate or use closed-loop controls in monitoring wind speed and in positioning the PV modules of solar panels to minimize wind disturbances while maintaining a steady flow of current at the point of interconnection (POI) between the PV system and the pre-existing electrical network [14].



Figure 1.6: A Closed-loop control system schema [14]

A typical closed-loop control system can be constructed from the components listed in Table 1. Figure 1.6 shows how the closed-loop control system in figure 1.5 is implemented. Note that the schema can be applied for small PV systems as well as big PV systems that utilize more than 40, 000 solar panels. In the schema, the master controller has numerous ports. One of these ports is connected to the PV system's inverters and other port is connected to a protective relay, which can provide the entire PV system a power factor. In [14] the relay interface used was EIA-232, while the communication networks between the inverters were constructed using multi-drop EIA-485. The SCADA that was used was connected physically with the master controller through a DSL modem. The SCADA and HMI also communicate though this DSL modem. The communication protocol used was a proprietary communications protocol, which was primarily used for the master controller-protective relay communication process. The communication of the protocol between SCADA and HMI systems is through Ethernet ModBus TCP/IP. Lastly, the protocol between the inverters and the master controller is through the serial ModBus RTU as shown in figure 6 [14].



Figure 1.2: Schema of the implementation of a closed-loop control system

1.4 SCADA Software, Algorithms, and Hardware

There are different commercial SCADA software that are available at present. Two examples of commercially available SCADA software include Winlog pro SCADA Software and Ignition SCADA [17] [18]. There are also free softwares that are downloadable on the internet. Two of these free softwares are Reliance SCADA and Easy Java Simulations [19]. It should be noted, however, that most of the available free softwares need to be reconfigured so that they can be properly integrated with PV systems. In [20] the SCADA software that was used was ARC Informatique's PcValue software. This software was configured so that it can be used to control different PV system components and operations such as solar trackers, substation systems, meters, and inverters. The study utilized a PV system that was capable of delivering 5MW power and other multi-tenant, multi-site PV systems of larger power outputs. The configured algorithm was capable of monitoring the entire PV system's power generation and delivery performance. The alrogirthm utilized a mathematical model that was initialized during the PV system installation with plant design data. The variables which were considered include string length, number of strings, electric parameters, inverter specifications, solar panel's peak power values, etc. The scheme for the architectural design, however, did not include an automated weather status determination. This information had to be manually feed to the master controller through the user interface and not at the algorithm or program level [20].

In [14] they have utilized an algorithm for their SCADA and HMI system. The algorithm considered a sequential step process for the PV system as shown in figure 1.7. The algorithm in the said figure was for a 20 MW PV system located on a single site. The PV system was integrated into a pre-existing electrical network as described in the previous discussions; two of these networks also have their independent electric power system. The solar panels or modules were built upon movable, single-axis solar trackers. The PV system has a generator on-site [14].

With regards to the control requirements, the PV system has to enable all its solar trackers when strong winds occur. The solar panels or modules have to be positioned at a specified angle for optimum sunlight exposure. The solar modules would also have to move to a certain angle for cleaning purposes. The control system should be able to turn-off the PV inverter as well as put all solar trackers to a stowed position upon power

outage. This means that the entire power plant would have to be switched to a certain state so that the controller would recognize a power outage. Ones the inverters are turned off, they can only be turned-on manually through the SCADA and HMI system. The corresponding algorithm for all these functions is shown in the figure 1.7. Note that, depending on the desired functions and the number and purposes of the elements of the PV system, the algorithm may change. Nevertheless, for a small PV system, the algorithm in figure 1.7 would suffice. Note further, the algorithm takes into consideration that the functions or commands that are inputted automatically and manually into the system would follow a certain sequence. In other words, the algorithm follows a sequential-loop control system [14].



Figure 1.3: Basic algorithm for a PV system utilizing SCADA and HMI as main data processing elements [14]

In [20] they further discussed the basic communication protocols of the SCADA to the rest of a PV system. In his study the communication protocol was done through the ModBus at the DC level. The DC data were fed to the controller through string combinator boxes or string probe units that are able to read DC current voltage values as well as power values. The string probe units are interfaced with the RS-485 port. The study further notes that Remote Terminal Units (RTUs) have been installed at the location of the PV system. These RTUs are connected to multiple string junction boxes of the RS-485 multi-drop loops. With regards to the communication protocol at the AC level, the inverters expose RS-485 so as to allow easy connection. For real time acquisition, processing, storage, alarming, display, and reporting, the communication drivers of PcValue collect AC are time-stamped data from RTUs and control boxes. The other SCADA functionalities were also used for the monitoring of energy meters, grid protection relays, low tension and high tension control panels, weather monitoring/sensors, transformers, DC switches, etc. The SCADA application also allowed dynamic configuration, redundancy for data protection, client and stand-alone server configurations, real-time trend & historical analysis, and advanced alarm or forecast management. It was further noted in the study that IEC 61850 and DNP3 can be used to support the above-mentioned communication protocols in case there is a need to communicate to different substations [20]. Further a study, it was explained that IEC 61850 can be utilized for effective utility automation [19].

Aside from the aforementioned software and algorithms, a SCADA algorithm can also be made using NI cRIO. In [21] a PV system with a central processing unit or system to monitor an entire PV plant was constructed. The system was able to take or read different sensor readings, transfer data from a centralized location for report generation & storage purposes, and generate remote site alarms. The built-in centralized system was able to transmit data that were collected from remote plants to the centralized data storage device through a GPRS modem. The study noted that ADSL can be used instead of GPRS. To do this, all the device server was interfaced with all the inverters using RS-485, RS-232, and RS-422. It was also noted in the study that the centralized control system was able to response to other types of inverters as long as the software is configured accordingly. Moreover, all digital meters and instruments can be connected to the central control device through standard serial interfaces, which winlcude RS-485, RS-422, and RS-232. The rest of the system's instruments communicated to the Remote I/O units through digital signaling and 4-20mA standard [21]. The remote site hardwares, data center hardwares, and the general descriptions of the software are provided in the following tables (see table 1.1, table 1.2 and table 1.3) [21].

| Hardware | Description |
|--------------------------|--|
| Device server – 6 sets | Ethernet to serial server RS485, RS422, and RS232, 2 ports |
| GPRS modem | EDGE/GPS modem by Signet |
| Industrial PC | Industrial PC by Advantech |
| Media converter - 5 sets | Multimode F/O industrial type by LINK |
| Pyranometer | KIPP & ZONEN |
| Switch 5-ports – 5 sets | Industrial Switch 5-ports |
| Switch 8-ports – 1 set | Industrial Switch 8-ports |
| Temperature Sensor | 4-20 mA output PT100 |

Table 1. 1: Remote Site Hardware Specifications

Table 1. 2: Data Center Hardware Specifications

| Hardware | Description |
|------------------|---------------------|
| SERVER | Server by HP |
| Operating System | Windows server 2008 |
| Database Server | SQL Server |

| Software | Description |
|-------------|---|
| Remote Site | (1) Software that can execute commands for |
| | real time data collection and transmission to |
| | the DATA center through NI LabView |
| | (2) Industrial PC software that can be used |
| | for local database back up and re- |
| | transmittance through LabView |
| | |
| Data Center | (1) Web-based service data collecting |
| | software to colected data from the remote |
| | site |
| | (2) Web-based service software for server |
| | to client log-in and SCADA |
| | (3) Database management software |

Table 1. 3: Brief Descriptions of Software Used

1.5 Cost Efficiency of SCADA-Based Monitoring and Control Systems

In [22] it was explained that the overall efficiency of PV systems is improved when SCADA-based monitoring and control systems are used. This efficiency is attributed to the minimization of cost for overall automation of PV systems. It was further explained in the study that different efforts have been done by researchers to further decrease the cost of PV system monitoring and control through SCADA systems. In order to determine how the cost of such systems can be further decreased. Performed a cost benefit analysis of voltage control, SCADA systems used in PV systems. The results of the study shown that local or stand-alone PV systems have cheaper SCADA systems compared to connected PV systems. The main reason for this difference in cost, is not necessarily the hardware that was used by each system, but the level of complexity of

their respective algorithms. Therefore, conclude that the cost of SCADA systems can be further decreased by the parameterization of their algorithms [22].

In a [23] it was shown that for very small PV systems, a web-based SCADA system can be used to minimize the cost. The cost can be further decreased by using a cheap central control device. In their study, they have used an AVR ATMEGA8535 microcontroller device, which is approximately \$2 to \$5 in price. The schema of the SCADA system that they used is shown in figure 1.8. The researchers also used low cost software called the IntegraXor software, which can be used in the web [23].



Figure 1. 4: Illustration of a low-cost SCADA system for small PV system [23]

The pins that were used in the AVR ATMEGA8535 microcontroller are shown in the figure 1.9. Based from the figure, the microcontroller was used to obtain the following information: intensity of solar irradiation, wind speed & direction, and temperature. All this information was fed into the microcontroller device using an analog to digital converter [23][24].



Figure 1.5: Block diagram of the pins used of the AVR ATmega8535 [24]
There are diverse studies that show that Arduino Uno and Arduino-like boards can be used for constructing SCADA systems for automated monitoring and control purposes. The basic fuction of such boards is to serve as the central control that can perform error monitoring functions, and send commands to other parts of a PV system [25]. In [26] an Arduino EMGA board was used for fault detection of solar panels. The board has been proven to improve the overall performance of the entire PV system[26]. It should be noted that such SCADA systems have also been proven to be efficient in executing, monitoring and control functions of hybrid PV systems [27]. The main component of such boards which allow them to be utilized in SCADA systems is the microcontroller chips, instead of an entire Arduino board. One of such studies was conducted; which utilized a PIC18F4550 microcontroller [28].

1.6 Thesis Objectives

In these days, a small PV systems are using wildly. Beside the PV system, SCADA system shall install to do the following functions

- Control industrial processes locally or at remote locations
- Monitor, gather, and process real-time data
- Record events into a log file

In markets, there are many commercial SCADA systems but they are very expensive. Also, there are papers describe commercial SCADA systems with non-open source software, and they applied on different system such as Photovoltaic, wind turbine, different factories ...etc, such as in papers [29]]30].

However, this research contains four diverse methods to design a SCDA system for a small PV system with low cost and lowest consumption power, and whole methods have been applied in the lab in Engineering Building at Memorial University of Newfoundland and Labrador. Following are the main research objectives:

- To design a low cost SCADA system based on Arduino Uno, and Raspberry-pi with web server on the internet (www.ubidots.com). In this case, Wi-Fi and mini USB have been used. Arduino Uno connects with Raspberry pi by using mini USB. The pi sends data to a web server by using Wi-Fi.
- To design a low cost SCADA system for a small PV system based on Arduino Uno and free software. In this case, Wi Fi has not used. The software, which are used to designs user interface, is Reliance SCADA. In addition, the power consumption has been calculated.
- To design a low cost SCADA system for a small PV system based on ESP32, SD card reader, and Wi-Fi. The SD card reader connected with ESP 32 by using SPI pins of ESP32. The function of SD card is to save data as text file and web server page. When ESP32 connects with internet, ESP32 gets an IP. In order to download the file of data, the IP of ESP32 shall write in an internet browser either

in a computer or a tablet. On the web page, a link will appear, which describes how download the file of data. In addition, the power consumption has been calculated.

• To design a low cost monitoring and data acquisition system for a small PV system based on Arduino LoRa. The Arduino LoRa is two devices. One (ALT) shall locate at PV system and other (ALR) shall locate at a control room. ALT is connects with sensors which are measure parameters of PV system and send data to other side where ALR is installed. ALR is connected with a computer by using mini USB. Whole parameters are saved in computer as text file, and shown in monitor screen as well. In addition, the distance between ALR and ALT has been tested in St. John's, Newfoundland and Labrador, Canada.

1.7 Thesis Organization

According to the school of graduate studies of Memorial University of Newfoundland guidelines, this thesis has been written in the manuscript format. As the papers are dealing with closely related subjects, there exists a certain amount of overlap of the introductory material among the chapters.

In chapter two, an open low-cost SCADA system has been designed for monitoring a solar energy system installation. In this design, Arduino Uno and Raspberry pi have been used to implement the SCADA system. In addition, a server on the internet

(www.ubidots.com) is employed. The proposed system has been applied on PV system to monitor parameters of the PV system include: voltage, current of PV and battery, and efficiency. This chapter was accepted and published in the conference proceedings and presented in IEEE Newfoundland Electrical and Computer Engineering Conference (NECEC) 2016, St. John's, Newfoundland and Labrador, Canada.

The work of Chapter 3 explains design low-cost SCADA system for a small PV system, however without using webserver as mention in chapter 2. In this work, the SCADA system was based on low cost sensors, Arduino UNO and free Reliance SCADA software. This design has been applied on a small PV system in the lab in Engineering building at Memorial University of Newfoundland and Labrador to monitor current, voltage of PV and battery, and efficiency as well. The Arduino collects data from sensors, and it is connected with computer by a USB. The Arduino is programmed to send the data to Reliance SCADA on a computer. This chapter has been accepted for publication in International Journal of Solar Energy (Hindaw) Accepted 27 July 2017.

In chapter 4, a low cost Web Server has been designed for real-time photovoltaic system monitoring. This design was based on low cost sensors, a microcontroller ESP32, Wi-Fi, and SD card reader. ESP32 has been programmed to collect data from sensors and save it on SD card as text file. Period of saving file was about a week and then it will delet and creat new text file to save new data. Morover, file data could be downloded by using Wi-Fi. This design was applied in lab in Engineering bulding at Memorial university of Newfoundland and Labrador. This chapter was accepted and published in the conference proceedings and presented in IEEE Electrical Power and Energy Conference, 2017. St. Saskatoon, Saskatchewan, Canada.

In chapter 5, a monitoring and data logging system has been designed for a small photovoltaic systems. The data include: voltage, current of PV and battery. This design is based on low cost sensors, and Arduino Lora. In this work, we did not use Wi-Fi or mini USB to transfer data to computer. In this technic, LoRa antenna has been used to send data from PV location to room location. Moreover, Arduino LoRa has been tested in many locations in St. John's, Newfoundland and Labrador, Canada. The longest distance founded to receive excellent data about 15Km. This chapter was accepted and published in the conference proceedings and presented in IEEE Newfoundland Electrical and Computer Engineering Conference (NECEC) 2017, St. John's, Newfoundland and Labrador, Canada.

Summary of the research work, research contributions and future work have been included in chapter 6.

1.8 References

 [1] Chine, Willard, et al. "Fault detection method for grid-connected photovoltaic plants." Renewable Energy, vol.66 no. 1, 2014, pp.99–110.

- [2] Baziliana, Morgan, et al. "Re-considering the economics of photovoltaic power". Renewable Energy, vol. 53, no. 1, 2013, pp. 1-21
- [3] Moreno-Garcia, Isabel M. et al. "Real-Time Monitoring System for a Utility-Scale Photovoltaic Power Plant." Ed. Albert M. K. Cheng. Sensors (Basel, Switzerland) vol. 16, no.6, 2016, pp.1-25.
- [4] Zhang, Peng, et al. "Reliability assessment of photovoltaic power systems: Review of current status and future perspectives." Applied Energy, vol. 104, no.1, 2013, pp. 822-833.
- [5] Karoly, Ronay and Christina Dragos Dumitri. "Management of a Power System Based Renewable Energy." Procedia Technology, vol.12, no.1, 2014, pp. 693 – 697.
- [6] Manzano, Santiago, et al. "An Overview Of Remote Monitoring PV systems: Acquisition, Storages, Processing And Publication Of Real-Time Data Based On Cloud Computing." Academia. 2015.
- [7] Prinsloo, Gerro and Robert Dobson. Solar Tracking. Solar Books. 2015.
- [8] Schreck, Tom and Petit Agarwal. "SCADA Versus Web-Based Monitoring." Ausenco. 2014.

http://www.ausenco.com/uploads/news/1405647839-solarpro-5-pg12-qa.pdf

- [9] Barron-Gafford, Greg A., et al. "The Photovoltaic Heat Island Effect: Larger solar power plants increase local temperatures." Scientific reports 6 (2016): 35070.
- [10] "Effect of Temperature." PV Education. 2017.

http://www.pveducation.org/pvcdrom/effect-of-temperature. Accessed 3 Sept 2017

- [11] ABB, Inc. "Industrial Automation." ABB. 2014.
 https://library.e.abb.com/public/26715ccc7867052ec1257c22002e8f60/1TXH000213
 B0201.pdf. Accessed 4 Sept 2017.
- [12] Byarak, Gokay and Mehmet Cebeci. "A PV Based Automation System for Fish Farms: An Application Study ." Firat University. 2016.
- [13] Trillo-Montero, D., et al. "Development of a software application to evaluate the performance and energy losses of grid-connected photovoltaic systems." Energy Conversion and Management 81 (2014): 144-159.
- [14] Mills-Price, Michael and Kei Hao. "The Importance of Coordinated Control Systems in Solar Generation Plants." 1st Annual PAC World Americas Conference. 2014.

https://cdn.selinc.com/assets/Literature/Publications/Technical%20Papers/6658_Impo rtanceCoordinated_KH_20140729_Web.pdf?v=20170306-105901. Accessed 3 Sept 2017.

- [15] Wang, Yu. "Algorithms for Optimal Energy Management in the Smart Grid." Auburn University. 2015.
 https://etd.auburn.edu/bitstream/handle/10415/4674/YuWangPhdDisserAUETD.pdf? sequence=2. . Accessed 7 Sept 2017.
- [16] Mills-Price, Michael, Michael Rourke, and Darrin Kite. "Adaptive Control Strategies and Communications for Utility Integration of Photovoltaic Solar

Sites." proceedings of the 2014 Power and Energy Automation Conference, Spokane, WA. 2014.

- [17] SIELCO Sistemi. "Winlog pro SCADA System Software." SIELCO Sistemi.2017. https://www.sielcosistemi.com/
- [18] Inductive Automation. "Ignition: The New SCADA." Ignition SCADA Software.2017. https://inductiveautomation.com/scada-software/
- [19] Gonzalez-Redondo, Miguel J., et al. "Influence of data-related factors on the use of IEC 61850 for power utility automation." Electric Power Systems Research 133 (2016): 269-280.
- [20] Kareli, Guatam. "Scada & Monitoring Systems Used In Solar Power Plants In India, What Is Scada For Solar Power Plants?" 2015

https://www.linkedin.com/pulse/scada-monitoring-systems-used-solar-power-plantsindia-gautam-karelia

- [21] Satitpanyapan, Ranon. "SCADA and Monitoring for Solar Energy Plant." Graphical Achievement Design Achievement Award. 2017. ftp://ftp.ni.com/pub/branches/asean/gsdaa13_finalist08.pdf
- [22] Stetz, Thomas, et al. "Technical and economical assessment of voltage control strategies in distribution grids." Progress in Photovoltaics: Research and Applications 21.6 (2013): 1292-1307.

- [23] Soetedjo, Aryuanto, et al. "Web-SCADA for Monitoring and Controlling Hybrid Wind-PV Power System." TELKOMNIKA (Telecommunication Computing Electronics and Control) 12.2 (2014): 305-314.
- [24] Kashyap, Nisha. "Design of Low Cost Multi Channel Data Acquisition System for Meteorological Application." Department of Electronics and Communication Engineering National Institute of Technology. 2015.
- [25] Barnicha, Fatima Ezzahra, Ahmed Khallaayoun and Rachid Lghoul. "Smart Home Energy Management System Monitoring and Control of Appliances Using an Arduino Based Network in the context of a Micro-grid." Al Akhamethodan University. 2015.

http://www.aui.ma/sse-capstone-repository/pdf/Smart-Home-Energy-Management-System-Monitoring-and-Control-of-appliances-Using-an-Arduino-Based-Network-inthe%20context-of-a-Micro-grid.pdf

 [26] Dahoud, Ali Al and Mohamed Fezari. "Improving Monitoring and Fault Detection of Solar Panels Using Arduino Mega in WSN." Research Gate, vol. 13, no. 3, 2015, pp. 1072-1078.

http://www.zuj.edu.jo/wp-content/uploads/2014/05/with-ali-fezari.pdf

[27] Puigt, Matthieu. "Using Arduino to Monitor an Home-made Energy Autonomous Platform." TEKTOS. 2014. <u>http://www-lisic.univ-littoral.fr/~puigt/LectureNotes/Arduino/ArduinoDay_vfinal.pdf</u>

- [28] Segundo, Alan Kardek Rêgo, et al. "Low cost SCADA system for education." Global Engineering Education Conference (EDUCON), 2015 IEEE. IEEE, 2015.
- [29] Ayompe, L. M., et al. "Measured performance of a 1.72 kW rooftop grid connected photovoltaic system in Ireland." Energy conversion and management 52.2 (2011): 816-825.
- [30] Pietruszko, S. M., and M. Gradzki. "Performance of a grid connected small PV system in Poland." *Applied energy* 74.1 (2003): 177-184.

Chapter 2

Design an Open low-cost SCADA system for Monitoring a Solar Energy Installation

Preface

A version of this manuscript has been published in the conference proceedings of IEEE Newfoundland Electrical and Computer Engineering Conference 2016, St. John's, Newfoundland and Labrador, Canada. This paper has been presented in that conference. The co-author Dr. Tariq Iqbal supervised the first author Ibrahim Allafi to develop the research and helped him to use the best possible way to implement the idea using existing techniques. Allafi connected and implemented the experiment in the lab, analyzed the results and wrote the paper while Dr. Iqbal reviewed the results, manuscript, and provided necessary suggestions.

Abstract

This paper demonstrates a system to monitor photovoltaic current, voltage, and the battery remotely via internet using a web browser. Designed system uses low-cost sensors, a microcontroller Arduino Uno, and a Raspberry pi to implement the SCADA system. Server on the internet (www.ubidots.com) is employed as Web-SCADA to show the parameters of the solar energy system. The Arduino Uno collects data from sensors, and it communicates with Raspberry pi by USB cable, which has been programmed to transmit data to the server on the internet. Arduino is used as a slave while Raspberry pi is used as a master in this work. The results of the experiments demonstrate that the SCADA Website works in real time, and can be effectively used for monitoring a solar energy system.

Index Terms: Arduino Uno Microcontroller, Raspberry pi, ubidots.com and SCADA, Sensors.

2.1 Introduction

For several hundreds of years, fossil fuels on the earth are consumed, and they have rabidly gone down in recent years. People now understand the importance of renewable energy and have dedicated research, expansion, and deployment of such important energy. Photovoltaic (PV) is an example of renewable energy, however PVs depend on some factors to generate power, and they are longitude, latitude, and weather. In addition, power generation by PV is limited to daytime [1].Recent developments in renewable energy resource, PV system have been used significantly. The PV systems are located separately in rural area. To monitor them remotely, the Supervisory Control and Data Acquisition (SCADA) system were utilized in [2].

The SCADA is a process control system that qualifies a site operator to observe and control processes that are distributed between different distant sites. So, we can eliminate the need for service personnel to visit each site for inspection, data collection or make adjustments. Result of elimination is to save time and money. In addition, we can find SCADA system in many of industries such as electric distribution systems, facility security alarms, and food processing as well.

2.2 Literature Review

A previous study has shown that researchers could monitor and control a hybrid power system by Web-SCADA. In addition, they have used sensor systems and RTUs to collect data of the system, all data is transmitted to Web-SCADA server, PLC is used to control the system. The PLC relies on control command that come from Web-SCADA [2].

Another study has shown that, we can use Zigbee-based, microcontroller PIC18F553, and web-application for online monitoring of a grid-connected photovoltaic system. The parameters, which are monitored, are temperature, irradiation, PV power output, and grid inverter power output, as an example. The result was, people could monitor the system remotely, and they could access the system from any place and any time via internet [3]. Another study was done by a microcontroller (PIC16) and LABVIEW-based SCADA system to monitor and control parameters of a greenhouse system which are temperature, humidity, soil moisture and intensity of light [4].

The SCADA system also has been used in monitoring stations for checking air quality level in civilized environments. The experiment has implemented, and an internet technology and large-scale sensor networks are one of technic which was used in the experiment [5].

When we would like to control small industries such as water treatment stations; electric power stations, we can use a SCADA with programmable Logic Control (PLC). So, a past study has shown PLC and SCADA system to control in oil and gas refineries instead of using Distributed Control System (DCS) which oil and gas refineries depend on it to provide whole process and devices control functions. SMIATIC WinCC FLEXIBLE 2008 has been used to design Graphical User Interface (GUI) of main units of the system; it has four units. Which are used to connect between operators and the system (oil refinery field). The advantage of using SCADA and PLC system are to monitor the system, detect the problems, and execute actions much faster than DCS, and the DCS is not cheap. The SCADA system has ability to save data for a long time, and the SCADA screens are more realistic than DCS screens for the operators [6].

SCADA system has been used in complex system; it was used in power network in Gaza Strip. The power network has been feeding from three places: Israeli Electric Company, Egyptian Power Company, and the Gaza Power Station. In addition, in this system, the SCADA relies on optimal power flow (OPF). So, the SCADA system was employed to gather data, observe and control performance of operation system. This idea introduces a solution for power distribution in Gaza Strip [7].

According to one of previous studies, researchers have applied SCADA system based on wireless communication for offshore wind farm. They have used Profibus fieldbus to communicate between lower computer and sensor; which is located in the wind turbine while between upper computer and SCADA system of control room is connected by wireless communication. They have achieved monitoring and management offshore wind farm by SCADA system [8].

In this paper, the SCADA system is employed for monitoring parameters of solar energy system (photovoltaic) in real-time which consist of solar module, MPPT, and Batteries. The Parameters are current and voltage of the photovoltaic, and current and voltage of the battery. Data acquisition system is by Raspberry-pi, Arduino controller, and sensors with server on internet (www.ubidots.com) that allows users and operators to monitor parameters of PV system. The equipment which are used to design this work, are cheap.

2.3 System Architecture



Figure 2. 1: SCADA Architecture

Figure 2.1 shows the architecture of the proposed SCADA system, and the graphical user interface is designed by server "www.ubidots.com" to receive data from Raspberry-pi by dongle Wi-Fi. Raspberry-pi connects with Arduino by USB Cable, and some sensors have installed on inputs analog of Arduino to gather data from solar energy system. The purpose of Arduino UNO, it can read analog sensors that is why it connected with

Raspberry pi. Users and operators can access all data via internet either by computers or application download on their phones. Design of SCADA system consists of hardware, and software.

2.4 Hardware Design

The system was designed to be low-cost and all operations are under software control, so we can expand or modify the system without the need for major hardware changes in future. Basic elements of the design are as shown in Figure 2.2 they are Arduino Uno controller, Raspberry-pi, and sensors.



Figure 2. 2: Hardware Configuration of SCADA System

2.4.1 Arduino Uno Microcontroller

Arduino Uno is open source hardware, and it is easy to use. Figure 2.3 shows Arduino Uno, and Table. 2.1 shows specification of Arduino. The licence gives permeation anyone to improve, build or expand Arduino. The original Arduino and its enhancement environment was founded in 2005 in Italy in the Smart Project Company. Moreover, it has 14 digital input/output pins. Six of them can be used as analog input/output [9]. They are used in this project as shown in Figure 2.2.



Figure 2. 3: Arduino Uno

The languages of Arduino are very easy to use, and one of language is IDE (Integrate Development Environment) software. In this work, The Arduino is programmed to change its operation to work as slave with Raspberry-pi because Raspberry-pi just has digital inputs, and we need analog inputs to connect with sensors

Table 2. 1: Specifications of Arduino board

| Specifications | | | | |
|-------------------------|-------------------|--|--|--|
| Microcontroller | ATmega328 | | | |
| Operating Voltage | 5V | | | |
| Input Voltage | 7-12V | | | |
| Input Voltage (limits) | 6-20V | | | |
| Digital I/O Pins | 14 (6PWM outputs) | | | |
| Analog Input Pins | 6 | | | |
| DC current per I/O Pin | 40mA | | | |
| DC current for 3.3V Pin | 50mA | | | |
| Flash Memory | 32 KB(ATmega328) | | | |
| SRAM | 2 KB(ATmega328) | | | |
| EEPROM | 1KB(ATmega328) | | | |
| Clock Speed | 16 MHz | | | |

2.4.2 Raspberry-pi

To design this work, we use Raspberry-pi Model B with Linux operating system, and Figure 2.4 shows Raspberry-pi. The Raspberry-pi is a credit-card size computer, and it was evolved in the Laboratory of University Cambridge in UK and lunched by raspberrypi Foundation in 2012 [10]. Raspberry-pi is characterized by a 700MHZ ARM11 coprocessor, Broadcom Video Core IV graphics, and 512 MB RAM on model B and 256 MB RAM on model A [11]. It has four USB ports, and one Ethernet port. In addition, it has 26 bins including 8 General purpose Input/output (GPIO), one SPI bus, one I2C bus, one UART bus and 3.3V, GND and 5V [12]. Further, Raspberry-pi has enormous feature, it has open-source operation system such as Linux (Raspbian). On this system, you can run many operating systems like Android, Arch Linux ARM, Firefox operation system, Google chromium [13]. However, we need Secure Digital (SD) card to store its operation system and all the user data, and the SD have to 4GB at least. There is programming language including with operating system of Raspberry-pi is called Python; it has two version python2 and Python3. It is used to write code of the raspberry-pi. Also, you can use Nano to write program of Raspberry-pi.



Figure 2. 4: Raspberry-pi

2.4.3 Current Sensor

Current sensors for DC current must be able to measure range of current of PV and battery, and they are between 0 A and 20 A. In this work, the CR5210-50 has been used for sensing the current. It is designed to provide a DC signal which is proportional to a DC sensed current. Moreover, it uses for direct current only, and the CR5210-50 can measure between 0 to 50 ADC. Its voltage supply is 24 volts while it has output voltage between 0 and 5 volt; however this voltage is appropriate for Arduino.



Figure 2. 5: Connection Drawing

Figure 2.5 demonstrates how it connects in electrical circuit with Arduino Uno. In this paper, we need two current sensors. One is installed before MPPT, and another is installed after Maximum Power Point Tracking (MPPT) as shown in Figure 2.2.

2.4.4 Voltage Sensor

We need to drop a voltage of PV and battery to 5 volts because Arduino Uno analog inputs are capable to receive data as voltage; it is between 0 and 5V. Depend on theory of voltage divider; we have designed two voltage dividers to measure or sense voltage of PV and voltage of battery. They are installed as shown in Figure 2.2.



Figure 2. 6: Voltage Divider for voltage of PV



Figure 2. 7: Voltage Divider for voltage of Battery

The voltage divider is composed from a set of two resistors; they are connected in series. Figure 2.6 illustrations values of resistors which are appropriate to sense voltage of PV. Figure 2.7 demonstrates voltage divider to sense voltage of battery.

2.5 Software

There are two types of SCADA to make graphical user interface: proprietary or open. Some companies have developed proprietary software to communicate with their hardware, however it has enormous problem; it dependence on the supplier of the system. Open software systems have gained Popularity because of the interoperability they bring to the system. Human Machine Interface (HMI) is software to make graphical user interface to monitor and control whole the system. And also, WinTr software; it is advanced SCADA software for monitoring and save data of any system, and it has many advantages [14]. However, in this work, www.ubidots.com is used to design user interface to collect data from our system (solar energy system). Ubidots.com provides the graphical tools, and save data for long time. Moreover, you can download all data as excel file, and other features it has.

2.6 Hardware Setup

The proposed SCADA system is designed to monitor the solar energy system developed on Department of Electrical Engineering, Memorial university of Newfoundland and Labrador. Figure 2.8 shows hardware of the SCADA for solar energy system while figure 2.8 shows the whole system after the designed system was connected.



Figure 2. 8: Some Hardware of the system



Figure 2. 9: Photo of Communication of Hardware and Software

2.7 Results and Discussion

The proposed SCADA is implemented in real solar energy system and several experiments are conducted. The experiment covers the measurement errors of sensor system which are installed to measure parameters of the system, and, efficiency of the system. The sensor systems, which are used in this work, have errors. So, these errors are calculated with calibrated instruments as listed in Table 2.2. As shown in the table 2.2, the measurement error of current sensors is highest among the others. Error percentage of PV-current sensor and Battery-current sensor are about 11.3% and 9.04% respectively.

While error percentage of both voltage sensors (voltage dividers) is low and they are closer to calibrated instrument.

| No | Sensor Module | Value | Calibrated Instrument | Value | Measurem ent Error % |
|----|-------------------------------|-------|--------------------------|--------|----------------------------|
| 1 | PV-Voltage Sensor | 13.0V | Standard Voltmeter | 13.67V | 1.71% |
| 2 | Battery- Voltage Sensor | 13.2V | Standard Voltmeter | 13.39V | 0.75% |
| 3 | PV-Current Sensor | 2.19V | Standard Ampermeter | 2.47V | 11.3% |
| 4 | Battery Current Sensor | 2.17V | Standard Ampermeter | 1.99V | 9.04% |
| | | | | | Avr=5.7% |

 Table 2. 2 MEASURMENT ERROR OF SENSORS SYSTEM

After we run the experiment, go to www.ubidots.com, we got values of parameters of our system and efficiency of the system as shown in Figure 2.10. The voltage and current of photovoltaic are 15.98 V and 6.60A respectively while current and voltage of battery are 13.8 V and 6.38 A respectively, and the efficiency of the system in that time was 0.83 or 83%, and it represents output power of MPPT over input power to MPPT. Figure. 2.11 shows efficiency of the system period of the time.

| | | | Dashboard | | terative |
|--|---|--|---|--|---------------|
| | | | | | Achd Variable |
| ⊕ 15.98Volt | • • ന 6.60Amps | @ 13.80Valt | € 6.38Amps | @ 0.83 | |
| Vultage-FV Last Action, a New Incompliant | Current-RV Last Activity a few seconds age | Voltage-Battisty Last Actum, a New excentionage | Current-Battery sati Attriji, alter accosti agri | Power Last Athlije a tex temprats egt | |
| O Hotes | O hrop | C Seriegi | Q Notings | O no nee | |
| Add variable | | | | | |
| Add variable | | | | | |
| Add variable | | | | | |
| Add variable | | | | | |
| Add variable | | | | | |
| Add variable | | | | | Des |

Figure 2. 10: Front Screen of SCADA System





If we click on one of icons of main screen on ubidots.com, we will get values of sensor as chart. For example, Figure. 2.12 shows values of current sensor of PV. Also, another icons give us chart between values of sensor and time when click on it.

| ubidots | | | |
|--|--|---------------------------------------|------------|
| Back to Solar System (PV) | | | |
| urrent-Pv | | | |
| 0.0 | | | |
| | Statistic • 2016 07-16 17:38 - 2016 07-16 17:49 • | | 00 |
| | 31 | | Current-Pv |
| unnent-PV att acturity: in a fex seconds | man man and a show a show when the show | Annum and Backing a second | |
| No tags | 45 | Marin Mary Marin Marine | www.w |
| | | | |
| me: | 4.0 | | |
| rent-Py | 3.5 | | |
| bel: | 3.1 | | |
| in in in its second | 17:39 17:40 17:41 17:42 17:43 17:44 17: | 45 17:46 17:47 17:48 17:49 | 17:50 |
| R | Sæt 16 Sæt | 16 Sat 16 Sat 16 Sat 16 Sat 16 Sat 16 | 6 Sat 16 |
| scription: | | | |
| | Baw data | | |
| gs: | | | |
| tag | Date Value | | |
| eranla7625420eR0858er5 | 2016-07-16 17:50:18 -0230 4.6432067 | | |
| 1 Endneintr | 2016-07-16 17:50:17:0230 4:59433084 | | |
| n, Entopount. ps://things.ubidots.com/api/v1.ti/variabi | 2016-07-16 17:50:16-0230 4:59433084 | | |
| | 2016-07-16 17:50:14-0230 4-54545498 | | |
| | 2016-07-16 17:50:13 -0230 4:59433084 | | |
| | 2016-07-16 17:50:12-0230 4:54545498 | | |
| | 2016-07-16 17:50:10 -0230 4:59433084 | | |
| | | | |

Figure 2. 12: Current of PV against Time

| bidots Data × M Inbox (53) - Ra686@gma × | 🥵 My Drive - Google Drive 🛛 🗙 🏹 🐼 Voltage Divider Calculator 🗴 🏹 🌺 HDW-TO use the ARDUD | 🗙 💘 Connect Raspberry Pi an 🛛 🗙 🏹 🚺 Dr. Monk's I | DIV Electronics × | |
|--|---|--|---------------------------|---------------------|
| | /es/#/detail/5/7ec9ae7625427810609ec8/variable/577ecaD87625420b18d11445 | | Dathboard | Duents Brahim - |
| 0010013 | | | | |
| Back to Solar System (PV) | | | | |
| oltahe-Pv | | | | |
| • • | | | | |
| ♠ 15.38Volt | Statistic • 2016-07-16 17:38 - 2016-07-16 17:51 • | | | 00 |
| Voltahe-Pv | 21 | | | Construction of the |
| Last activity: in a few seconds | 20 | | | |
| O No tags | 19 | | | |
| | 18 | | | |
| Name: | 17 | | | |
| | | | | |
| Label: rotahe-pi | 15 | | | |
| Joit: | 17599 17540 17541 17542 | 1/343 1/344 1/345 1/346 | 5 17/42 17/48 17/49 17/50 | 12:31 12:34 |
| 68. | | Sat 16 Sat 16 | Sat 16 Sat 16 Sat 16 | Sat 16 |
| Description: | | | | |
| | Paw data | | | |
| Tags: | Norr Goto | | | |
| dd teg | Date | | | |
| D: | 2016-07-16 17:52:19-0230 | 15.376345556 | | |
| 77ecab87625420b/8d/1445 | 2016-07-16 17:52:17 -0230 | 15.376345556 | | |
| JRL Endpoint: ttos://those.ubidots.com/api/v1.6/veriebl | | | | |
| rs/577ecab87625420b/8dr1445 | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Figure 2. 13: Voltage of PV against Time

| C https://app.ubidots.com/ubi/datasourc | ces/#/detail/577ec9ae7625427a1d609ec8/variable/577ecb097625421090a5 | ice5c | | | | |
|---|---|---------------------------|---------------------|-------------|-------------|--------|
| ubidots | | | | | | |
| Back to Solar System (PV) | | | | | | |
| urrent-Battery | | | | | | |
| @ (1) | | | | | | |
| 🔂 0. 78Amps | Statistic • 2016-07-16 17:38 - 2016-07-16 17:53 • | | | | | 00 |
| 0.70711105 | 08. | | | | • current-B | sttery |
| current-Battery Last activity: in a few seconds | 0.8 | | | | | |
| No tags | 0.8 | | | | | |
| | | | | | | |
| ame: | 0.0 | | | | | |
| rrent-Battery | 0.8 | | | | | |
| abel: | 0.7 | | | | | |
| ait | 17:39 17:40 17:41 17:42 17: | 43 17:44 17:45 17:46 17:4 | 7 17:48 17:49 17:50 | 17:51 17:52 | 17:53 17:54 | 17:55 |
| 190 | Sat 16 Sat 16 | Sat 16 Sat 16 Sat 16 | Sat 16 Sat 16 | Sat 16 | Sat 16 | Sat 16 |
| escription: | | | | | | |
| | Paw data | | | | | |
| ags: | Norr Gala | | | | | |
| ld tag | Date | | | | | |
|): 77a-19-0076-75-6311000-90-45- | 2016-07-16 17:55:06 -0230 | 0.78201376 | | | | |
| | 2016-07-16 17:55:05 -0230 | 0.78201376 | | | | |
| RL Endpoint: tps://things.ubidots.com/api/v1.6/variabl | 2016-07-16 17:55:03 -0230 | 0.78201376 | | | | |
| /577ecb097625421090a9ce5c | 2016-07-16 17:55:02 -0230 | 0.78201376 | | | | |
| | 2016-07-16 17:55:01 -0230 | 0.78201376 | | | | |
| | 2016-07-16 17:54:59 -0230 | 0.78201376 | | | | |
| | 2016-07-16 17:54:57 -0230 | 0.78201376 | | | | |
| | | | | | | |



| C https://app.ubidots.com/ubi/datasourd | ces/#/detail/577ec9ae7625427a1d609ec8/variable/577ecaee7625420f7892f | 03a | S LAT ERCOON: A | Card County |
|---|--|--|-------------------------|-------------------------------------|
| ubidots | | | Dashboard | Sources Events Ibrahim - |
| Back to Solar System (PV) | | | | |
| ltage Battery | | | | |
| 0 @ | | | | |
| € 14.38Volt | Statistic • 2016-07-16 17:38 - 2016-07-16 17:52 • | | | 00 |
| | 15 | | | Voltage Battery |
| ottage Battery madvity in a few seconds | 15 | | | |
| No taga | 15 | . Jundlin | | |
| | 14 | 11 | Ild It Warm | |
| me: | 14 | | | |
| tage Battery | 14 | | | |
| bel: aze-battery | 14 | | | |
| dr | 17:39 17:40 17:41 17:42 | 17:43 17:44 17:45 17:46 | 17:47 17:48 17:49 17:50 | 17:51 17:52 17:53 |
| | Sat 16 Sat 16 | Sat 16 Sat 16 | Sat 16 Sat 16 Sat 16 | Sat 16 |
| scription: | | | | |
| | Raw data | | | |
| gs: | 1011 0010 | | | |
| tag | Date | | | |
| araaa763542067802403a | 2016-07-16 17:53:03 -0230 | 14.384165598000001 | | |
| Production Control of | 2016-07-16.17:53:02 -0230 | 14.384165598000001 | | |
| L Endpoint: ss://things.ubidots.com/api/v1.6/variabl | 2016-07-16 17:53:00 -0230 | 14.384165598000001 | | |
| 577ecaee7625420f7892f03a | 2016-07-16 17:52:59 -0230 | 14,384165598000001 | | |
| | 2016-07-16 17:52:56 -0230 | 14.384165598000001 | | |
| | | | | |
| | 2016-07-16 17:52:55 -0230 | 14.384165598000001 | | |
| | 2016-07-16 17:52:55 -0230 2016-07-16 17:52:54 -0230 | 14.384165598000001 14.384165598000001 | | |

Figure 2. 15:Voltage of Battery against Time

| And the state of the second state of the | | | | |
|--|---|---|---------------------------------------|-----------------------|
| ubidots | | | | |
| Back to Solar System (PV) | | | | |
| ower | | | | |
| 0 0 | | | | |
| ♠ 64.56Watts. | Statistic - 2016-07-16 17:38 - 2016-07-16 1 | 17:55 * | | 00 |
| | 110 | | | Power |
| Power Last activity in a few seconds | 100 | | | |
| No tags | 90 | | | |
| | 2016/07/16 17:39:22 | | | |
| Inner | Power 79.195718271325 | man man have up and a sub will | my wanter and a second and the second | |
| GUIVE. | 10 | | | any man way when when |
| abel: | 60 | | | |
| ewet | 49 17:39 17:40 17:41 17:42 | 17:43 17:44 17:45 17:46 17:47 | 17:48 17:49 17:50 17:51 17:52 | 17:53 17:54 17:55 |
| Jnit: | | ~ ^ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ | | |
| | Sat 16 Sat 16 | Sat 16 Sat 16 Sat 16 | Sat 16 Sat 16 Sat 16 | Sat 16 Sat 16 |
| Pescription: | | | | |
| ne | Raw data | | | |
| ags. | | | | |
| 50 tag | Date | Value | Context | |
| 0: 77ecber7625431896co4881 | 2016-07-16 17:55:52 -0230 | 64.56028731002739 | | |
| | 2016-07-16 17:55:51 -0230 | 64.56028731002739 | | |
| HL Endpoint: tps//things.ubidots.com/api/v1.6/veriabl | 2016-07-16 17:55:50 -0230 | 65.30235957795875 | | |
| 2/577ecbef7625421fb/co48/1 | 2016-07-16 17:55:48-0230 | 65.30235957795875 | | |
| | 2016-07-16 17:55:47-0230 | 66.0444318458901 | | |
| | 2016-07-16 17:55:45 -02:30 | 65.30235957795875 | | |
| | 2016-07-16 17:55:44 -0230 | 64.56028731002739 | | |
| | | | | |

Figure 2. 16: Power before MPPT against Time

2.8 Conclusion

This paper emphasizes on the low cost SCADA to monitor electrical parameters of the solar energy system. Arduino controller is developed to collect data of the sensors, and Raspberry-pi receives data from Arduino to transfer them to server on internet (www.ubidots.com), and it is used to design a basic user interface of SCADA system.

We have achieved to monitor the solar energy system remotely via the web browsers or app on cell phone. Data is also stored on SD card for local use.

2.9 Acknowledgment

Authors would like to thank the Ministry of Higher Education and Scientific Research of Libya for funding this research.

2.10 References

 Dumitru, Cristian-Dragoş, and Adrian Gligor. "SCADA based software for renewable energy management System." Procedia Economics and Finance 3 (2012): 262-267.

 [2] Soetedjo, Aryuanto, et al. "Web-SCADA for Monitoring and Controlling Hybrid
 Wind-PV Power System."TELKOMNIKA(Telecommunication Computing Electronics and Control) 12.2 (2014): 305-314.

 [3] Shariff, Farihah, Nasrudin Abd Rahim, and Wooi Ping Hew. "Zigbee-Based Data Acquisition System for Online Monitoring of Grid-Connected Photovoltaic System."
 Expert Systems with Applications 42.3 (2015): 1730- 1742.

[4] Gajipara, Nishantkumar D., and Prashant L. Ahire. "Design of SCADA for Real Time System with LabVIEW and Microcontroller." International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN (2014): 2349-2163.

[5] Tomić, Josif, et al. "Smart SCADA System for Urban Air Pollution Monitoring." Measurement 58 (2014): 138-146.

[6] Morsi, Iman, and Loay Mohy El-Din. "SCADA System for Oil Refinery Control." Measurement 47 (2014): 5-13.

[7] Elaydi, Hatem, and Hasan Abu Meteir. "Using SCADA System to Improve Electricity in Gaza Strip." (2012). [8] Meng, Yanjing, and Wenzhan Gong. "Design of SCADA System Based on Wireless Communication for Offshore Wind Farm." Recent Advances in Computer Science and Information Engineering. Springer Berlin Heidelberg, 2012. 347-352.

[9] Vařacha, P. A. V. E. L., et al. "Technical Devices for Supervising of a Household via Interned Based on Arduino Microcontroller." 12th WSEAS International conference on Applied Informatics and Communications, Istanbul, Turkey. 2012.

[10] Brock, J. Dean, Rebecca F. Bruce, and Marietta E. Cameron. "Changing the world with a Raspberry Pi." Journal of Computing Sciences in Colleges29.2 (2013): 151-153.

[11] Anwaar, Waqas, and Munam Ali Shah. "Energy Efficient Computing: A Comparison of Raspberry PI with Modern Devices." Energy 4.02 (2015).

[12] Manasa, J., et al. "Real Time Object Counting using Raspberry pi."

[13] Abrahamsson, Pekka, et al. "Affordable and energy-efficient cloud computing clusters: The Bolzano Raspberry Pi cloud cluster experiment."Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on. Vol. 2. IEEE, 2013.

[14] http://www.scadasoftware.net/systems/scadaraspberrypi.ht ml

Chapter 3

Low-Cost SCADA System Using Arduino and Reliance SCADA for a Standalone Photovoltaic System

Preface

A version of this manuscript has been accepted for publication in the International Journal of Solar Energy (Hindaw). The co-author Dr. Tariq Iqbal supervised the first author Ibrahim Allafi to develop the research and helped him to use the best possible way to implement the idea using existing techniques. Allafi connected and implemented the experiment in the lab, analyzed the results and wrote the paper while Dr. Iqbal reviewed the results, manuscript, and provided necessary suggestions.

Abstract

SCADA (supervisory control and data acquisition) systems are currently employed in many applications, such as home automation, greenhouse automation, hybrid power systems, etc. Commercial SCADA systems are costly to setup and maintain therefore those are not used for small renewable energy systems. This paper demonstrates applying Reliance SCADA and Arduino Uno on a small photovoltaic (PV) power system to monitor the PV current, voltage, and battery, and efficiency as well. The designed system uses low-cost sensors, a microcontroller Arduino Uno, and free Reliance SCADA software. The Arduino Uno collects data from sensors and communicates with a computer through a USB cable. Uno has been programmed to transmit data to Reliance SCADA on PC. In addition, Modbus library has been uploaded on Arduino to allow communication between the Arduino connected to our SCADA system by the MODBUS RTU protocol. The results of the experiments demonstrate that SCADA works in real-time and can be effectively used in monitoring a solar energy system

Keywords— Arduino Uno, Modbus RTU, Sensor, PV Power system SCADA systems.

3.1 Introduction

For several hundred years, fossil fuels have been consumed as the main source of energy on Earth. As a result, they are now experiencing rapid depletion. Researchers and scientists who understand the importance of renewable energy have dedicated their efforts to the research, expansion, and deployment of new energy sources to replace fossil fuels.

Photovoltaics (PV) are an important renewable energy sources. Also called solar cells, PV are electronic devices which can convert sunlight directly into electricity. The modern forms of PV were developed at Bell Telephone Laboratories in 1954 [1]. Despite their promising performance, PV have some limitations, such as depending on factors like longitude, latitude and weather, and being limited to daytime hours to generate power [2]. The SCADA system is software that has been installed in several sites to monitor and control processes, and it is called telemetry importance [3][4]. SCADA can monitor realtime electrical data measurements of solar module and batteries and collect data from wind turbines, such as the condition of the gearbox, blades, electric system, etc. [5] [6]. Moreover, the sun-tracker system has also used the SCADA system to observe the solar insolation and movement of the sun [6].

These days, commercial companies are widespread for monitoring a system such as photovoltaic system. However, those are quite expensive. For example, SMA Company; it is a Germany Company, and it is founded in 1981. It has many products. Some of them are related to monitor and control, for example, Sunny View. It can show all of your system data in good condition, and we can read all data clearly. However the major problem is device costly; it is about CA \$793[7] [8].

In this paper, we look for to design SCADA system lower price compared with commercial devices, and it works the same performance. In order to test this work, the SCADA system is employed for monitoring the parameters of solar energy systems (photovoltaic) in real-time, which consists of a solar module, MPPT, and batteries. The parameters are the current and voltage of the photovoltaic (PV), and the current and voltage of the battery. Data acquisition system is by Arduino controller and sensors. All data are sent to a PC and are shown on a user interface designed by Reliance SCADA.

The data are saved on a computer as an Excel file as well. This allows users and operators to monitor the parameters of the PV system in real-time. The components of the SCADA system in this paper consist of two parts: hardware and software.

3.2 Hardware Design

The proposed Reliance SCADA is designed to monitor the parameters of a small PV system. It is installed at the Department of Electrical Engineering, Memorial University, St. John's, Canada. Figure 3.1 shows 12 solar panels up to 130 watts and 7.6 amps. Two solar modules are connected in parallel. Therefore system shown in Figure 3.1 consist of 6 sets of 260 watts each.


Figure 3. 1: Solar panels on the roof of engineering building

The Reliance SCADA system was designed to be low-cost and can be expanded or modified without the need for major hardware changes in the future. Basic elements of the design are an Arduino Uno controller and sensors, as shown in Figure 3.2.



Figure 3. 2: Hardware configuration of SCADA system

3.2.1 Arduino Uno Microcontroller

Arduino Uno is an open-source hardware that is relatively easy to use. Figure 3.3 shows Arduino Uno, while Table 3.1 shows specifications for the hardware. The license gives permission to anyone to improve, build, or expand Arduino. The original Arduino and its enhancement environment were founded in 2005 in Italy at the Smart Project Company. It has 14 digital input/output pins, 6 of which can be used as analog input/output [9].



| Table 3. | 1: | Specifications | of | Arduino | board |
|----------|----|----------------|----|---------|-------|
|----------|----|----------------|----|---------|-------|

| Specifications | | | | | |
|----------------------------|-------------------|--|--|--|--|
| Microcontroller | ATmega328 | | | | |
| Operating Voltage | 5V | | | | |
| Input Voltage | 7-12V | | | | |
| Input Voltage (limits) | 6-20V | | | | |
| Digital I/O Pins | 14 (6PWM outputs) | | | | |
| Analog Input Pins | 6 | | | | |
| DC Current per I/O Pin | 40mA | | | | |
| DC Current for 3.3V Pin | 50mA | | | | |
| Flash Memory | 32 KB(ATmega328) | | | | |
| SRAM | 2 KB(ATmega328) | | | | |
| EEPROM | 1KB(ATmega328) | | | | |
| Clock Speed | 16 MHz | | | | |

3.2.2 Current Sensors

Current sensors for DC currents must be able to measure a range of currents for PV and batteries between 0 A and 20 A. In this work, an ACS 712 sensor is used for sensing the current. It is designed to be easily used with any microcontroller, such as Arduino. The sensors are based on the Allegro AC712ELC chip. The scale value of ACS 712, which is used in this design, is 20 amps, which is appropriate for sensing current. Two sensors are installed: one is installed before MPPT to measure the PV current, and the other is installed after MPPT to measure the battery current. Figure 3.4 demonstrates how it connects in an electrical circuit with Arduino Uno.



Figure 3. 4: Connection drawing of current sensor

3.2.3 Voltage Sensors

The function of voltage sensors, which are based on just two resistors, is to turn a large voltage into a small one. In this work, the voltage sensor is a 25V-Sensor with two resistors of $30K\Omega$ and 7.3 K Ω . The maximum voltage of either PV or battery is 25V, so this sensor is appropriate. The output of the voltage sensor is between 0V to 5V. This scale is suitable to the Arduino analog inputs. In this experiment, we need two voltage sensors – one is installed before MPPT to measure the PV voltage, and the other is installed after MPPT to measure the battery voltage. Figure 3.5 demonstrates how it connects in an electrical circuit with Arduino Uno



Figure 3. 5 Connection Drawing of voltage sensor

3.3 Hardware Setup

Figure 3.6 shows the hardware setup designed for the SCADA system.



Figure 3. 6 Hardware setup of SCADA system

3.4 Software Requirements

3.4.1 Arduino IDE

IDE is open-source software which features easy-to-write code that can be uploaded to any board. In this work, we needed to upload a new library on IDE to make a configuration between Arduino Uno and SCADA software by MODBUS RTU protocol. The following flow chart shows how the system working, and the code which has been burned on Arduino Uno.

A. Flow Chart :



A. Code:

The code has some main functions such as setup() – it is called once when the sketch starts- and loop() - it is called over and over and is heart of sketch. The most important in the code libraries which mention in the first code, regBank.setId() command , regBank.add(), and regBank.set(). The purpose of libraries is to connect between Arduino Uno and Reliance SCADA software by MODBUS RTU protocol. regBan.setId() uses to define MODBUS to work as slave. For example, in this code Id was (regBank.setId(10)), so in Reliance software has to be 10 as well as shown in Figure 3.7. regBank.add() command is used to define addresses of registers which are used to send data to Reliance SCADA on computer. In this work, the addresses were from 30001 to 30005 as mention in the code. While reBank.set() command is used to write data on the previous addresses. Addresses of registers on Reliance software has to write as shown in Figure 3.8.

#include <modbus.h>

#include <modbusDevice.h>

#include <modbusRegBank.h>

#include <modbusSlave.h>

modbusDevice regBank;

modbusSlave slave;

float out1 = 0.0; float out2 = 0.0; float out3 = 0.0; float out4 = 0.0; float AlO; float Al1; float Al2; int Al3; float vpv = 0.0; float vbatt = 0.0; float eff = 0.0; float p1 = 0.0; float p2 = 0.0; int mVperAmp = 100; int ACSoffset = 2500; double Voltage2 = 0; double Voltage3 = 0; double Ampspv = 0; double Ampsbatt = 0; float R1 = 30000.0; float R2 = 7500.0;

void setup(){

Serial.begin(9600);

regBank.setId(10); ///Set Slave ID

regBank.add(30001);regBank.add(30002);regBank.add(30003);regBank.add(30004);regBank.add(30005);

```
slave._device = &regBank;
```

slave.setBaud(9600);

```
pinMode(0,INPUT);pinMode(1,INPUT);pinMode(2,INPUT);pinMode(3,INPUT); }
```

void loop(){

while(1){

int AIO = analogRead(0); int AI1 = analogRead(1);

int AI2 = analogRead(2); int AI3 = analogRead(3);

vpv = (AIO * 5.0) / 1024.0;

out1 = vpv / (R2/(R1+R2));

vbatt = (AI2 * 5.0) / 1024.0;

out2 = vbatt / (R2/(R1+R2));

Voltage2 = (AI1 / 1024.0) * 5000;

Ampspv = ((Voltage2 - ACSoffset) / mVperAmp);

Voltage3 = (AI3 / 1024.0) * 5000;

Ampsbatt = ((Voltage3 - ACSoffset) / mVperAmp);

p1 = vpv * Ampspv;

p2 = vbatt * Ampsbatt;

eff = p2/p1;

Ampspv=abs(Ampspv);

Ampsbatt=abs(Ampsbatt);

out1=out1*100;

Ampspv= Ampspv*(100);

out2=out2*100;

Ampsbatt=Ampsbatt*(100);

eff=eff*100;

regBank.set(30001, (word) out1);

regBank.set(30002,(word) Ampspv);

regBank.set(30003,(word) out2);

regBank.set(30004,(word) Ampsbatt);

regBank.set(30005, (word)eff);

delay(60000)

slave.run();}}



Figure 3. 7: Define Address MODBUS on Reliance Software

| 🚱 De | vice Manager | | | | | - | 100.000 | | | U | كار ال | | ^ |
|------|--|---|---------|-----------------|------------|---|---|------------------|-------|----------------------------|----------|--------|---|
| 1 | 🛍 🐁 📾 🐝 🐮 | 8 & 8 | 1 | AN 22 1 - 19 Ra | × 10. 44 × | | Scripts | Security | Sh | aring | Infor | nation | |
| | Current of Current of Current of Current of Current of Voltage of voltage of Somunications | Battery PV PV Battery PV D Zones | | | | | Basic Name Voltage of Alias General | Advanced Battery | Range | Correcti | on | Limits | 8 |
| 1 | Alams/Events | | | | | | Eng. name | | | | Eng. | unita | |
| | System | - | 1.20 | | | * | | | | | Volt | | • |
| Name | | Туре | Address | | | - | Tag kind | | | Update | interval | (ms) | |
| | Current of Battery | Word (16 | IR:3 | | | | Physical | | | 60,000 |) | E | * |
| | Current of PV | Word (16 | IR:1 | | | | Tag data ty | pe | | | | | |
| | Emciensy-PV | Word (16 | 18:4 | | | | Word (16) | 1 | | | | | - |
| | voltage of Battery | Word (16 | IR.2 | | | | 100(100) | | | | | - | |
| | | 1000 (10 | | | | | Mata structu | | | | | | |
| | | | | | | | Alow us | e at runtime | | | | | |
| | | | | | | | Alow re | ading | E A | low writing | | | |
| | | | | | | | Modbus | | | | | | |
| | | | | | | | Register typ | | > | | | | |
| | | | | | | | Input Regi | stors | | | | | - |
| | | | | | | | File number | Address 2 | | A V | | | |

Figure 3. 8: Shows writing addresses registers on Reliance SCADA

3.4.2 Reliance SCADA

Reliance software is employed in numerous technologies for monitoring and controlling systems. It can also be used for connecting to a smartphone or the web. Reliance is used in many colleges and universities around the world for education or scientific research purposes [10]. Figure 3.9 shows a user interface designed by Reliance SCADA software to monitor the parameters of the photovoltaic system.

The user interface has four real-time trends and four display icons to show values as digital numbers. In addition, it has two buttons and a container. These features are discussed in the results and discussion sections.



Figure 3. 9: User interface of SCADA system

3.5 Communication System

MODBUS library is added to Arduino Uno to allow communication with Reliance SCADA via a USB cable using MODBUS RTU protocol. Table 2 shows the allocation of MODBUS address for MODBUS RTU on Reliance SCADA software, with the MODBUS address for Arduino Uno mentioned in the Arduino Code.

| No | Variable Name | MODBUS RTU Address |
|----|-------------------------|-----------------------|
| 1 | Voltage of Photovoltaic | 0 |
| 2 | Current of Photovoltaic | 1 |
| 3 | Voltage of Battery | 2 |
| 4 | Current of Battery | 3 |
| 5 | Efficiency of MPPT | 4 |

Table 3. 2: Allocation of MODBUS Address for MODBUS RTU

3.6 Cost of the SCADA System

Most factories that use several systems are looking for a low cost SCADA system to monitor and control their systems remotely. In this paper, the components used are a quiet cheap. Table 3.3 shows the price (CA Dollar) for whole components according to the amazon.ca website.

| NO | Item | Quantity | Price |
|----|---|--------------|---------|
| 1 | Current Sensor | 2 | 6.8 |
| 2 | Voltage Sensor | 2 | 7.3 |
| 3 | Arduino Uno | 1 | 7.9 |
| 4 | Software (IDE) | _ | Free |
| 5 | Software (Reliance) | _ | Free |
| 6 | Old Small Computer (PC) With Windows 7 | 1 | 60 |
| | | Total amount | CA \$82 |

Table 3. 3: Allocation of MODBUS Address for MODBUS RTU

According Table 3.3 we found whole price of SCADA system was CA\$82. This price seems cheap to design SCADA system for monitoring parameters of our system.

3.7 Result and Discussion

In this work, the proposed SCADA monitors a solar energy system and several experiments are carried out. The experiments cover the measurement error of the sensor systems which are installed to measure PV current and voltage, battery current and voltage, MMPT efficiency, and SCADA features.

The sensors which are used contain errors, so these errors are calculated with calibrated instruments, as listed in Table 3.4.

| No | Sensor Module | Value | Calibrated Instrument | Value | Measurement Error |
|----|-------------------------------|-------|--------------------------|-------|----------------------|
| 1 | PV-Voltage Sensor | 15.3V | Voltmeter- Ammeter | 15.5V | 1.31% |
| 2 | Battery- Voltage Sensor | 13.84 | Voltmeter- Ammeter | 14.33 | 3.42% |
| 3 | PV-Current Sensor | 3.22A | Voltmeter- Ammeter | 2.77A | 1.62% |
| 4 | Battery- Current Sensor | 3.66A | Voltmeter- Ammeter | 3.55A | 3.10% |

 Table 3. 4:
 MEASUREMENT ERRORS OF SENSOR SYSTEM

As can be seen in Table 3.4, the measurement error of current sensors was highest. The error percentage of the PV current sensor and the battery current sensor are about 3.42% and 3.10%, respectively. Although the error percentages of both voltage sensors were quite low, they were closer to the calibrated instrument.



Figure 3. 10: User interface of SCADA while running

The monitoring tasks are displayed on the PC. They include the PV parameters as a graph and digital numbers, and the MPPT efficiency as digital numbers. Figure 3.10 shows the user interface of SCADA after the system was operational.

The SCADA system is designed to make an update every minute. As shown in Figure 3.10, there are four figures: two of them observe the PV voltage and current, and the other two monitor the battery voltage and current. The figure also shows that the SCADA system makes updates every minute.

The user interface of SCADA shows five icons displaying values of parameters as digital numbers, and they also make automatic updates every minute.

Our SCADA system has the feature of enabling all data to be easily saved on a computer as an Excel file. To save the data, the user just has to hit the Export Data icon, and then hit the Save Data icon. These icons are programmed by script to save the data on a PC as an Excel file. Figure 3.11 shows a screenshot of data saved in Excel.

Also, user inter face has a container which show details. The Arduino connects with SCADA, and it gives warning if there is any error in connection.

The efficiency of MPPT was also monitored. It represents the output power of MPPT over the input power to MPPT. Figure 3.12 presents MPPT efficiency for various periods of time, with efficiency ranging between 1 and 0.8.

| - | А | B | С | D | E | F | G | Н | 1 | J |
|----|---|------------|---|------------|---|------------|---|------------|---|-----------|
| 1 | | Voltage DV | | Current DV | | Voltage Di | | Current Di | | Efficance |
| 2 | | 17 12 | | 11 ce | | Voltage-B: | | Land | | encency. |
| 0 | | 17.13 | | 11.00 | | 15.56 | | 12.01 | | 0.92 |
| 4 | | 17.13 | | 11.00 | | 15.30 | | 12.01 | | 0.92 |
| 5 | | 17.13 | | 11.00 | | 15.56 | | 11.42 | | 0.92 |
| 7 | | 16.89 | | 11.27 | | 15.10 | | 11.42 | | 0.91 |
| 0 | | 16.89 | | 11.27 | | 15.10 | | 11.42 | | 0.91 |
| 0 | | 16.89 | | 11.27 | | 15.10 | | 11.42 | | 0.91 |
| 10 | | 17.13 | | 11.27 | | 15.10 | | 12.01 | | 0.91 |
| 11 | | 17.15 | | 11.00 | | 15.50 | | 12.01 | | 0.92 |
| 12 | | 17.13 | | 11.00 | | 15.38 | | 12.01 | | 0.92 |
| 12 | | 17.15 | | 11.00 | | 15.56 | | 12.01 | | 0.92 |
| 13 | | 16.99 | | 11.27 | | 15.35 | | 11.71 | | 0.93 |
| 14 | | 16.99 | | 11.27 | | 15.35 | | 11.71 | | 0.93 |
| 15 | | 16.99 | | 11.27 | | 15.35 | | 11.71 | | 0.93 |
| 10 | | 16.99 | | 11.27 | | 15.35 | | 11.71 | | 0.93 |
| 10 | | 16.99 | | 11.27 | | 15.35 | | 11.71 | | 0.93 |
| 18 | | 16.99 | | 11.27 | | 15.35 | | 11./1 | | 0.93 |
| 19 | | 16.99 | | 11.27 | | 15.35 | | 11.71 | | 0.93 |
| 20 | | 16.99 | | 11.2/ | | 15.35 | | 11./1 | | 0.93 |
| 21 | | 16.77 | | 10.64 | | 15.13 | | 10.79 | | 0.91 |
| 22 | | 16.77 | | 10.64 | | 15.13 | | 10.79 | | 0.91 |
| 23 | | 16.77 | | 10.64 | | 15.13 | | 10.79 | | 0.91 |
| 24 | | 16.77 | | 10.64 | | 15.13 | | 10.79 | | 0.91 |
| 25 | | 10.77 | | 10.64 | | 15.13 | | 10.79 | | 0.91 |
| 26 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 21 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 28 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 29 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 30 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 31 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 32 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 33 | | 17.4 | | 12.15 | | 15.33 | | 11.91 | | 0.86 |
| 34 | | 16.77 | | 10.64 | | 15.18 | | 10.79 | | 0.91 |
| 35 | | 16.77 | | 10.64 | | 15.18 | | 10.79 | | 0.91 |
| 36 | | 16.77 | | 10.64 | | 15.18 | | 10.79 | | 0.91 |
| 37 | | 16.77 | | 10.64 | | 15.18 | | 10.79 | | 0.91 |
| 38 | | 16.77 | | 10.64 | | 15.18 | | 10.79 | | 0.91 |
| 39 | | 16.77 | | 10.64 | | 15.18 | | 10.79 | | 0.91 |
| 40 | | 17.13 | | 11.66 | | 15.38 | | 12.01 | | 0.92 |
| 41 | | 17.13 | | 11.66 | | 15.38 | | 12.01 | | 0.92 |
| 42 | | 17.13 | | 11.66 | | 15.38 | | 12.01 | | 0.92 |
| 43 | | 17.13 | | 11.66 | | 15.38 | | 12.01 | | 0.92 |
| 44 | | 17.13 | | 11.66 | | 15.38 | | 12.01 | | 0.92 |
| 45 | | 17.13 | | 11.66 | | 15.38 | | 12.01 | | 0.92 |

Figure 3. 11 Data saved in Excel file



Figure 3. 12: Efficiency of MPPT

3.8 Conclusion

In this paper, a low-cost SCADA system was designed and built with Reliance SCADA software and Arduino Uno. The SCADA system was applied to a stand-alone photovoltaic system to monitor the current and voltage of PV and batteries. The results of the experiments demonstrate that SCADA works in real-time and can be effectively used in monitoring a solar energy system. The developed system costs less than \$100 and can be modified easily for a different PV system.

3.9 Acknowledgment

Authors would like to thank the Ministry of Higher Education and Scientific Research of Libya for funding this research.

3.10 References

- [31] IRENA International Renewable Energy Agency. "RENEWABLE ENERGY TECHNOLOGIES: COST ANALYSIS SERIES", June, 2012
- [32] Dumitru, Cristian-Dragoş, and Adrian Gligor. "SCADA based software for renewable energy management System." Procedia Economics and Finance 3 (2012): 262-267.

- [33] Kim, Kyusung, et al. "Use of SCADA data for failure detection in wind turbines." ASME 2011 5th International Conference on Energy Sustainability. American Society of Mechanical Engineers, 2011.
- [34] Wang, Li, and Kuo-Hua Liu. "Implementation of a web-based real-time monitoring and control system for a hybrid wind-PV-battery renewable energy system." Intelligent Systems Applications to Power Systems, 2007. ISAP 2007. International Conference on. IEEE, 2007.
- [35] Figueiredo, João MG, J. M. G. Sa da Costa, and R. R. Ramalho. "Intelligent suntracking system for efficiency maximization of

International Conference On Renewable Energy and Power Quality. Santader. 2008.

- [36] Figueiredo, João MG, J. M. G. Sa da Costa, and R. R. Ramalho. "Intelligent suntracking system for efficiency maximization of photovoltaic energy production." Proceedings of International Conference On Renewable Energy and Power Quality. Santader. 2008.
- [37] http://www.sma.de/en/products/monitoring-control/sunny-view.html
- [38] http://www.ebay.com.au/itm/SMA-Sunny-View-Colour-Touchscreen-Solar-Monitor-Screen-/251743364128
- [39] Vařacha, P. A. V. E. L., et al. "Technical Devices for Supervising of a Household via Interned Based on Arduino Microcontroller." 12th WSEAS International conference on Applied Informatics and Communications, Istanbul, Turkey. 2012
- [40] https://www.reliance-scada.com/en/main

Chapter 4

Design and Implementation of a Low Cost Web Server Using ESP32 for Real-Time Photovoltaic System Monitoring

Preface

A version of this manuscript has been published in the conference proceedings of IEEE Electrical Power and Energy Conference, 2017. Saskatoon, Saskatchewan, Canada. This paper has been presented in that conference. The co-author Dr. Tariq Iqbal supervised the first author Ibrahim Allafi to develop the research and helped him to use the best possible way to implement the idea using existing techniques. Allafi connected and implemented the experiment in the lab, analyzed the results and wrote the paper while Dr. Iqbal reviewed the results, manuscript, and provided necessary suggestions.

Abstract

This paper presents a method for applying a web server based on ESP32 to a small photovoltaic (PV) power system to monitor and/or collect the PV and the battery current, voltage data. The designed system uses low-cost sensors, a microcontroller ESP32, Wi-Fi, and an SD-card reader. The ESP32 collects data from sensors. All of the data are

saved in a text file on an SD-card, which is connected with ESP32 SPI pins. The text file is saved on the SD card for a week or longer, after which the system deletes all the data and starts saving new data. The web page file is saved on an SD card as well, so the ESP32 is programmed to access the web page by using the Wi-Fi via a laptop, cellphone, or tablet. Moreover, the web page has a link that allows users to download the data text file simply by clicking on the link. This can also be done remotely. The results of the experiments demonstrate that the web server works in real-time and can be effectively used for monitoring small solar energy systems.

Index Terms —ESP32, PV Monitoring, Data Logging, PV Power System, Low-Cost Web Serve.

4.1 Introduction

In most photovoltaic (PV) power stations, a monitoring and control system oversees operations of equipment such as PV array strings, storage batteries bank, controller (which achieves maximum power point such as MPPT) and electrical load [1]. In order to analyze, examine or make decisions regarding PV stations, a monitoring system is needed [2-3]. For example, if either the voltage or battery current goes up, the loads could be damaged. Monitoring system is also used to determine total energy produced.

A web site can be hosted by a web server and provides requested services for clients. Web servers consist of an operating system, web pages, and a vast amount of memory. Special hardware is also sometimes needed [4]. Clients can access a web server through a LAN router and the Internet. Figure 4.1 shows typical client-server architecture. For example, if a client wants to access the server, a request is sent to the server. This request requires a router that is connected to the Internet such design could be used for large PV installation. For small PV installation, a low-cost solution is required that can storage data locally and data should be accessible through electronics like cellphone, tablet or laptop. Low-cost web server for small PV installations should not consume any significant power. Internet is not required for data download. Data could be download using a cellphone by connecting to a local webserver through local Wi-Fi.

In this paper, a low-cost ESP32 is used to collect the parameters of a PV power system. The parameters are the current and voltage of the photovoltaic (PV), and the current and voltage of the battery. All data are saved on an SD card as a text file, with two files being saved on a card. One is an image that will appear when the web page has been opened as the background for a page. Other one is web page file with an HTML extension. The ESP32 is running a web server code via Wi-Fi, so the ESP32 has an IP address. A user could type the IP in any browser on a laptop or phone can access the web page, and data which is saved on an SD card. The page has a link to download all the data saved on the card.



Figure 4. 1: Server architecture

4.2 PV System Description

The test photovoltaic system consists of 12 solar modules. Each module is 130W and two are connected in parallel. Two parallel modules can generate up to 260 watts and 15 amps. The system is connected to a battery bank, a battery-operated control unit (MPPT), and a load. Figure 4.2 shows the proposed PV system connection.

Sensors are placed in various locations in the system. There are two current sensors, one of which is installed in front of the MPPT to measure the PV current, and the other which is placed behind the MPPT to measure the load current or battery. There are also two

voltage sensors: One is installed in front of the MPPT to monitor the PV voltage, and a sensor is placed after the MPPT to measure the battery voltage or load.



Figure 4. 2: Proposed PV system connection

4.3 System Design

4.3.1 Sparkfun ESP32 Thing

An ESP32 is an open-source microcontroller based on an input/output board. The board is designed to support Wi-Fi and Bluetooth. ESP32 has 36 GPIO, 14 of which are Analog to Digital Converter (ADC) that could be connected to the sensors. When a 3.3 volt is applied to one of the ADC ports, the value is 4095. This value is called the reference

voltage to the ESP32. In addition, the ESP32 has ISP pins which are used to connect the ESP32 with an SD card reader. The range VCC supplied to the ESP32 is 2.2V to 3.6V. Either the ESP32Thing connects to a micro USB to upload the program and power, or it connects to a 3.7V battery for power. Figure 4.3 shows an ESP32 SparkFun Thing connected to a micro USB, while Table 4.1 provides details on some of the ESP 32 specifications



Figure 4. 3: ESP32 sparkfun thing

Table 4. 1: SPECIFICATIONS OF ESP32

| Specifications | | | | | |
|-------------------|---------------|--|--|--|--|
| Operating Voltage | 2.2V to 3.6V | | | | |
| GPIO | 36 ports | | | | |
| ADC | 14 ports | | | | |
| DAC | 2 ports | | | | |
| Flash memory | 16 Mbyte | | | | |
| SRAM | 250 Kbyte | | | | |
| Clock Speed | Up to 240 MHZ | | | | |
| Wi-Fi | 2.4 GHz | | | | |
| Sleep Current | 2.5 μΑ | | | | |

4.3.2 SD CARD Reader

In order to save data, an SD card is used in this paper. Internal storage of ESP32 can also be used for limited storage. However, because an ESP32 does not have a place to insert an SD card, SPI pins are used to connect the SD card reader. Figure 4.4 shows how an SD card reader connecting with an ESP32 by SPI pins.

| | 3V3 | VCC | |
|-------|-------------|-----|----------|
| | MISO(Pin19) | DO | SD C |
| ESP3: | MOSI(pin23) | DI | ard R |
| 2 | SCK(Pin18) | SCK | leade |
| | CS(Pin2) | CS | , |
| | GND | GND | |
| | | | I |
| | | | |

Figure 4. 4: ESP32 connected to SD card reader



Figure 4. 5: SD card and ESP32 connection

The voltage operation of an SD card is 3.3V, while the maximum voltage on the SPI pins of an ESP32 is 3.3V as well. We do not need to connect either the pull-up or pull-down resistors between the pins of an SD card and those of an ESP32 to balance the voltage or protect the SD card from damage. In this study, we need an SD card for three main reasons: to save data (parameters of PV system power) as a text file; to save files that represent web pages and html extension; and to save the file as an image that will appear as the web page background. Figure 4.5 shows the SD card reader and the ESP32 connection on a breadboard in a lab.

4.3.3 Wi-Fi

An ESP32 can implement TCP/IP, full 802.11 b/g/n/e/i WLAN MAC protocol, and Wi-Fi Direct specifications [5]. The frequency of the transmitter or receiver is about 2.4GHZ, which is suitable for a web server to download data from an SD card using any cellphone or tablet with Wi-Fi.

4.3.4 Sensors

In this paper, four sensors are used: two are voltage sensors, and two are current sensors. The voltage sensors have been installed in an electric circuit to measure the voltage of the PV and battery, acting as a voltage divider. As mentioned previously, the operating voltage of an ESP32 is 3.3V, so the voltage divider is designed to drop the voltage of the PV and battery to 0V - 3.3V. Figure 4.6 shows an illustration of the values of resistors that are appropriate for measuring the voltage of a PV and battery.



Figure 4. 6 : Voltage divider connection

The voltage divider is composed of a set of two resistors connected in series. Based on the voltage of the PV and battery, the values of the resistors are calculated in the voltage divider circuit. In the current sensors, the maximum current of the PV and battery is measured using a voltmeter. In our experiments, the maximum current was 20A. Based on this value, the ACS 712 current sensor was installed in the PV power system. The sensor is based on the Allegro AC712ELC chip. Because its operating voltage is 5V, it can be powered directly by the ESP32. The output voltage of the sensor is between 0V to 5V, but this voltage is not appropriate for ADC ports of the ESP32. In order to resolve this issue, a pull-down resistor is placed between the current sensor ESP32 output and input. The purpose of the pull-down resistor is to drop the maximum voltage from 5V to 3.3V maximum. Figure 4.7 is shows an ACS 712 current sensor and pull-down resistor connection.



Figure 4. 7: ACS 712 current sensor and pull-down resistor connection

4.4 Hardware Setup

All the required components were installed in the lab at the Department of Electrical Engineering, Memorial University, St. John's, Canada. Figure 4.8 shows the entire system connection, while Figure 4.9 shows the hardware setup in a close-up photo.



Figure 4. 8: Hardware setup connection



Figure 4. 9: ESP32, SD card and sensor connections

4.5 Software Used

4.5.1 Arduino IDE

IDE is open-source Arduino software (IDE), which makes it easy to write code and upload it to the board. It can be run on Windows, Mac OS X, or Linux. In the present study, three factors were taken into consideration before writing the program for the ESP32. First, the package-ESP32 had to be added to IDE. Second, because the Wi-Fi library is same as the Arduino IDE library, we did not need to add another library for the Wi-Fi ESP32. Finally, the new SD card library was uploaded to IDE.

4.5.2 HTML Script

Earlier in this paper, we mentioned that a file had been saved on an SD card for the web page. The file was programmed in HTML (Hyper Text Markup Language), which is a format that tells a computer how to display a web page. The following code is for the web page.

<html>

```
<body background="image.jpg">
```

<center>Web Server to Collect Data of PV Power

System</center>

<center>Click HERE to DownLoad

Data</center>

</body>

</html>

4.6 Implementation and Result

To implement this project, we needed to open a serial monitor of Arduino IDE, as shown in Figure 4.10.



Figure 4. 10: Serial monitor of IDE a test run

As can be seen in the serial monitor, an IP address has appeared. We need to open the web page, so the IP address has been copied from the serial monitor and entered into an Internet browser either on a computer or a cellphone. In this project, we used Google Chrome and Firefox on two different laptops at the same time and reached the web page on both devices. Figure 4.11 shows two laptops accessed to the web page. One is opened via Google Chrome and the other via Firefox.



Figure 4. 11: Different Internet browsers showing web page



Figure 4. 12: Main web page for data download
Figure 4.12 shows a snapshot of the main web page that includes a link to download data. When the blue link on the main web page is hit, the ESP32 will receive a command to start downloading a text file of the data from the SD card to the computer. Figure 4.13 shows when the text file has started to download.



Figure 4. 13: Downloading text file containing data

The downloading operation typically takes a few minutes, depending on Internet speed and the size of the text file. Figure 4.14 shows a text file of the data after being opened. The data, which are well-organized and easy to read, refer to voltage, current PV and battery, and efficiency as well.

| text(4).txt - Notepad | - | | | - | | | - | | - 0 × |
|-----------------------|---------------|------------|------------|------------|--------------------|---|---|------|------------|
| File Edit Format View | Help | | | | | | | | |
| | | | | | | | | | ^ |
| Voltage-PV: | Current-PV: | Voltage-B: | Current-B: | Efficency: | | | | | |
| 17.01 | 11.37 | 15.3 | 11.52 | 0.91 | | | | | |
| 17.01 | 11.37 | 15.3 | 11.52 | 0.91 | | | | | |
| 17.01 | 11.37 | 15.3 | 11.52 | 0.91 | | | | | - |
| 17.01 16.06 | 11.3/ 8.49 | 15.3 | 11.52 | 0.91 | | | | | - |
| 16.06 | 8.49 | 14.81 | 8.54 | 0.92 | | | | | |
| 16.67 | 10.59 | 15.11 | 10.83 | 0.92 | | | | | |
| 16.67 | 10.59 | 15.11 | 10.83 | 0.92 | | | | | |
| 15.03 | 5.46 | 14.25 | 5.56 | 0.96 | | | | | |
| 15.03 | 5.46 | 14.25 | 5.56 | 0.96 | | | | | |
| 15.03 | 5.46 | 14.25 | 5.56 | 0.96 | | | | | |
| 17.01 | 11.32 | 15.13 | 10.35 | 0.81 | | | | | |
| 17.01 | 11.32 | 15.13 | 10.35 | 0.81 | | | | | |
| 16.99 | 11.27 | 15.28 | 11.42 | 0.91 | | | | | |
| 16.99 | 11.27 | 15.28 | 11.42 | 0.91 | | | | | |
| 17.21 | 11.91 | 15.45 | 11.91 | 0.89 | | | | | |
| 17.21 | 11.91 | 15.45 | 11.91 | 0.89 | | | | | |
| 16.87 | 11.03 | 15.23 | 11.32 | 0.92 | | | | | |
| 16.87 | 11.03 | 15.23 | 11.32 | 0.92 | | | | | |
| 16.87 | 11.03 | 15.23 | 11.32 | 0.92 | | | | | |
| 16.8/ | 11.03 | 15.23 | 11.32 | 0.92 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 17.11 | 11.62 | 15.35 | 11.76 | 0.9 | | | | | |
| 4 | 11.02 | 13.33 | 11.70 | 0.9 | | | | | |
| | | | | | | - | | EN 🚝 | 12:13 PM |
| | | | | | line of the second | | | | 30/05/2017 |

Figure 4. 14: Text in the data file4.7 Cost the Web Server

In these day, small and large systems PV need to be monitored and data is collected. Commercial systems achieve this objective use very expensive equipment. So, we have concentrated a how to design lowest cost web server in this paper. Table 4.2 presents main requirement of the design with the prices. The prices are according to amazon.ca website, and in Canadian Dollar (CA Dollar).

| NO | Item | Quantity | Price |
|----|-------------------|-----------------|------------|
| 1 | Current Sensor | 2 | 6.80 |
| 2 | Resistors | Package(600PCS) | 7.12 |
| 3 | ESP32 | 1 | 87.84 |
| 4 | Software(IDE) | | Free |
| 5 | SD-Card Reader | 1 | 5.80 |
| 6 | SD card (4GB) | 1 | 6.58 |
| 7 | Micro USB | 1 | 7.00 |
| | | Total cos | t \$121.14 |

Table 4. 2: COST design web server

Total cost is about CA\$122, and this price is very appropriate to design a web server in order monitor and protect small PV systems. In addition, this price might be changed in future and most liked it will go down. The price also depends on quality components.

4.8 Power Needs for Design

In this work, the power consumption which represents ESP32, SD-card, and current sensors have been calculated. Table 4.3 shows currents and voltages which have been measured by multimeter for each one unit.

| Device | Voltage Drawn | Current Drawn | Power | | |
|----------------------------------|------------------|------------------|---------|--|--|
| ESP32 | 3.3V | 150mA | 500 mW | | |
| Current Sensor | 5V | 12.5mA | 62.5 mW | | |
| Current Sensor | 5V | 12.5mA | 62.5 mW | | |
| SD-Card | 3.3V | 0.15mA | 0.5 mW | | |
| Total Consumption Power 625.5 mW | | | | | |

From the table 4.3, we found the overall power consumption is about 0.6255 Watts. Only this much power is required to run a web server, and collect PV system data.

4.9 Conclusion

The main goal of this paper was to design and implement a web server using an ESP32 controller, an SD card, and Wi-Fi. This design was applied on a PV power system to

collect data, showing the voltage and current of the PV and battery. The ESP32 collects data and saves it on an SD card as a text file. Users can then access this data via the Internet and download it either using a laptop or a cellphone. The data on the SD card can be saved for a week or longer, after which the system will remove the old data and start saving new data.

The total system cost was less than CA\$122 that includes ESP32, SD-card, current sensors, USB cable, and resistors. This is the lowest cost possible system that could be used for logging small PV system data and monitoring. Total power consumption by ESP32, two current sensors and SD card was about 0.6255 watts, and this power is needed to design web server for small PV power system.

4.10 Acknowledgment

Authors would like to thank Libyan government for providing graduate student funding for this research.

4.11 References

[1] Zahran, Mohamed, et al. "Wired and wireless remote control of PVsystem." WSEAS Transactions on Systems and Control 5.8 (2010): 656-666. [2] Mingo, Laurent, and P. Eng. "A Cost-effective Solution for Photovoltaic Panels Performance Monitoring Using NI LabVIEW." Laurent Mingo at Blue System Integration Ltd., www.bluesystem. ca.

[3] Hossain, Mohammad Zakir, AKM Sadrul Islam, and D. G. Infield. "PVwind Hybrid System Modelling for Remote Rural Application." ISESCO Science and Technology Vision 3.4 (2007): 59-64.

[4] Roy, V. Billy Rakesh, Sanket Dessai, and SG Shiva Prasad Yadav. "Design and development of ARM processor based webserver." International Journal of Recent Trends in Engineering 1.4 (2009): 94-98.

[5] https://cdn.sparkfun.com/datasheets/IoT/esp32_datasheet_en.pdf

Chapter 5

A Low-cost Monitoring and Data Acquisition System for Small Photovoltaic System Based on Arduino LoRa

Preface

A version of this manuscript has been published in the conference proceedings of IEEE Newfoundland Electrical and Computer Engineering Conference 2017, St. John's, Newfoundland and Labrador, Canada. This paper has been presented in that conference. The co-author Dr. Tariq Iqbal supervised the first author Ibrahim Allafi to develop the research and helped him to use the best possible way to implement the idea using existing techniques. Allafi connected and implemented the experiment in the lab, analyzed the results and wrote the paper while Dr. Iqbal reviewed the results, manuscript, and provided necessary suggestions.

Abstract

In this research a monitoring and data logging system is designed and developed for small photovoltaic systems. Designed system can monitor various parameters, along with PV and battery voltages and currents. Different sensor modules are connected to an Arduino LoRa Transmitter (ALT) to measure and collect system parameters, while ALT sends the parameters to another (ALR) working a receiver. By using Putty software, the system operator can show the sensing data on a monitoring screen and save data that text file on a computer. Additionally, the distance between ALT and ALR has been tested at different points in St. John's, NL, Canada. The results of the experiments demonstrate that the designed monitoring and data acquisition system can transmit up to 15Km in

real-time using only 1.125W

Index Terms—PV Power System, Low-Cost Monitoring System, Arduino based LoRa, Renewable Energy

5.1 Introduction

Over the past few decades, renewable energy sources have been developing rapidly, and as a result many renewable energy power systems are being installed around the world. However, the installation cost for these systems is still too high [1]. At the same time, there is a growing awareness among the general public that many natural resources now being used for energy, such as oil, natural gas and nuclear materials, are finite in nature and generate massive amounts of pollution that is damaging the environment and contributing to global warming. In response to these issues, much of the recent energy-related research is focusing on upgrading the efficiency of solar modules. Thus far, using the latest technology, efficiency has improved up to 40% [2]. There are currently several types of solar photovoltaic systems in the market, each with its own characteristics and price-point. The one commonality in all of the systems, however, is that they convert solar radiation into electricity. In order to define the characteristics and behavior of solar photovoltaic systems, monitoring and evaluation systems are needed [3] [4].

Monitoring is a useful approach for obtaining information about photovoltaic systems. For instance, through monitoring, we can determine system performance and locate any deficiencies that are hindering the effective collection of data [5]. Data acquisition systems are imperative for renewable energy resource applications in order to monitor and evaluate the performance of the installed system [6]. Several different data acquisition systems have evolved for gathering and processing. It is worth noting that observing the performance of renewable energy systems while they are under operation provides an optimal opportunity to evaluate them [7][8].

In this paper, a low-cost Arduino LoRa Transmitter (ALT) has been installed near a photovoltaic system. ALT is used to collect the photovoltaic system's data, which mainly consists of the current and voltage of the photovoltaic (PV), and the current and voltage of the battery. In order to collect the entire data, ALT is connected to some sensors (e.g.,

current sensor and voltage sensor) and a voltage supply of 5Volt. ALT has a pin 5Volt which can be used as a power supply.

The observation room is located about 1 km from the PV system. ALT has been installed in this room and is connected to a computer via a mini USB. Putty, which is software that can easily be added to a computer, is used to show serial COM-related data on the monitoring screen. The software includes the option to save data as a text file on a computer. In this paper, the farthest possible distances between ALT and ALR were tested in several places throughout St. John's, NL, Canada.

5.2 PV System

The PV system was installed on the rooftop of the Department of Electrical Engineering, Memorial University, in St. John's, Canada. The system consists of twelve modules covering a total area of about 14 m2. Two modules are connected in parallel, with up to 260 W and 14A.tilt angle is 490 fixed. A charge controller was connected between the modules and batteries as shown in the figure 5.1.



Figure 5. 1: Connection diagram

The positive side of the controller connects with the positive battery terminal, and the negative side of the controller connects to the negative battery terminal. All types of 12 volt rechargeable batteries, such as lead-acid automotive batteries, traction batteries and stationary batteries, can be recharged. Figure 5.1 shows a connection diagram of the battery and charge controller, while Figure 5.2 shows a photo of the PV system installation.



Figure 5. 2: PV system installation

5.3 Hardware Configuration

As mentioned in the previous section, ALT has been mounted in a photovoltaic system location. Two current sensors and two voltage sensors have also been installed. The main purpose of current sensors is to measure the PV and battery currents, while the main purpose of voltage sensors is to collect voltage data on the PVs and batteries. Additionally, ALR has been installed in the control room; its purpose is to show and save data on the computer using Putty software. The data saves as a text file, and is wellorganized and easy to read. Figure 5.3 illustrates a connection diagram of the proposed system.



Figure 5. 3: Connection diagram of proposed system

We used Google Earth to measure the distance between the PV system and control room. Figure 5.4 shows a snapshot which illustrates both locations. ALT was installed in the side building (Engineering Building), while ALR was installed inside the control room. Although either or both could have been installed outside the buildings, we needed to protect them from damage such as rain.



Figure 5. 4: PV location and control room on map (1Km distance)

5.4 Transition Station design

The transmission station consists of two functional sections: the sensors, and the microcontroller Arduino Lora Transmitter.

5.4.1 Arduino Lora Transmitter (ALT)

An Arduino Lora Transmitter (ALT) is an open-source microcontroller based on Atmeg328P. The board is designed to support a Lora antenna transmitter. ALT has 30 GPIO, of which 5 are Analog Input that can be connected to the sensors. ALT also supports SPI pins to connect with the data interface. In addition, the VCC range supplied to ALT is 1.8 - 3.6V, and ALT either connects to a mini USB to upload and power the program or it connects to a 3.6V battery for power. Figure 5.5 shows ALT and the names of pins.

| | ~ | л 1 | J2 | PB0 |
|-----------------------------|--|--|--|--|
| ADC7 | X | 2 PINI | PIN12 | PC4 |
| | ~ | 3 PINZ | PIN2 B | PC5 |
| PD3 | | 4 PINS | PIN3 4 | PC2 |
| PB1 | | S PIN4 | pin4 5 | PC3 |
| GND VCC GND VCC_5V | X X 1 3V3 1 1 X 1 X 1 X 1 1 X 1 1 | 6 PIN6 9 PIN7 9 PIN8 9 PIN9 1 PIN10 1 PIN11 2 PIN12 9 PIN13 9 PIN14 9 PIN15 | PIN6 PIN7 PIN7 PIN8 PIN9 PIN9 PIN10 PIN10 11 PIN11 PIN12 12 PIN13 14 PIN15 | Here A GND PD6 PD4 PD5 PD7 GND GND |



Figure 5. 5: Arduino Lora Transmitter

5.4.2 Sensors

Four sensors are used in this work – two voltage sensors and two current sensors. The voltage sensors serve as a voltage divider, which consists of two resistors in series whose values are $10K\Omega$ and $1.5K\Omega$. These values have been chosen depending on the maximum voltage expected from the PV. The function of the voltage divider is to drop the voltage of the PV or battery to 0V -3.3V to synch with the operation voltage of ALT, which is 3.3V. One of the sensors has been installed in the PV system to measure PV voltage, while the other sensor (which also has been installed in the PV system) measures the battery voltage.

The current sensors and the maximum current of the PV and battery are all determined by using a voltmeter. Based on the obtained value of 20A, we chose the ACS712 current sensor as it can easily achieve maximum 20A and thus is appropriate for our system. Furthermore, as the operating voltage of ACS712 is 5V, it has to be connected with the voltage divider to drop the voltage to the operation voltage of ALT. In the PV system location, two current sensors were installed: one to sense the PV current, and the other to sense the battery current.

5.5 Hardware Setup

In order to test this work, we have installed all of the required components in the lab at the Department of Electrical Engineering, Memorial University, St. John's, NL, Canada. Figure 5.6 shows the whole system connection.



Figure 5. 6: Hardware setup connection

5.6 Receiving Station Design

The receiving station consists of two functional sections, which are the computer and the microcontroller Arduino Lora Receiver.

The computer, which is used to show and save data, does not require advanced specifications, but it should support at least one USB port.

ALR is an open source microcontroller based on Atmeg328P and is designed to support a Lora Antenna Receiver. ALT has 30 GPIO, none of which have been used in this work. The power supply ranges between 1.8-3.6V.

5.7 Powering Needs for This Design

In this work, ALT, ALR and the current sensors are the energy consumers. Table 5.1 illustrates the currents and voltages measured by the multimeter for each unit.

| Device | Voltage Drawn | Current Drawn | Power |
|----------------|------------------|------------------|--------|
| ALT | 3.3V | 150mA | 500mW |
| ALR | 3.3V | 150mA | 500mW |
| Current Sensor | 5V | 12.5mA | 62.5mW |
| Current Sensor | 5V | 12.5mA | 62.5mW |
| | 1.125 W | | |

Table 5. 1: CURRENT AND VOLTAGE DRAWN

As can be seen in the table, the consumption power required to run the Monitoring and Data Acquisition System is approximately 1.125 Watts.

5.8 Software Used

5.8.1 Arduino IDE

IDE has been used to program ALT and ALR. IDE is open-source software that easily enables code to be written and uploaded to the boards. In addition, IDE can be installed on a variety of operating systems, including Windows, Mac OS X, and Linux. In this work, an important library has been added to IDE called library of Arduino Lora.

5.8.2 PUTTY

Putty is software that is easy to download and install on computers, and is only a few Mbytes in size. Putty is used here to show PVS data in real-time and can also save the data as a text file on a computer.

5.9 Range Testing

In this paper, we tested ALR and ALT across several different points in St. John's, NL, Canada, using Google Earth to determine the test locations. We placed ALT at the highest point in St. John's (Signal Hill), whereas ALR was placed at a different point. Figure 5.7 shows the paths between the points (line of site).



Figure 5. 7: Paths between ALT at Signal Hill and ALR at different points

Table 5.2 provides additional details about range testing. Specifically, it lists the coordinates for each ALR and ALT site under testing, and also shows the signal strength at each location.

| NO | ALT Coordinates | ALR Coordinates | Distance | Data Transfer |
|--------------------------------------|---|--|--|---|
| 1 | 47.57090894- 52.6810956 | 47.65064852- 52.80599718 | 13km | Excellent |
| 2 | // | 47.65019318- 52.81364684 | 13.5km | Poor |
| 3 | // | 47.675853- 52.808769 | 15km | Very Good |
| 4 | // | 47.589714- 52.719664 | 3.5km | Excellent |
| 5 | // | 47.595351- 52.726440 | 4.5km | Excellent |
| 6 | // | 47.600256- 52.732217 | 5.11km | Excellent |
| 7 | // | 47.601973- 52.738930 | 6km | Excellent |
| 0 | 11 | 47.610957- | | Verv |
| 8 | // | 52.751548 | 7.20km | Good |
| 8 9 | // | 52.751548 47.639795- 52.725150 | 7.20km 8.5km | Good Very Good |
| 8 9 10 | // | 52.751548 47.639795- 52.725150 47.617531- 52.723051 | 7.20km 8.5km 6km | Good Very Good Excellent |
| 8 9 10 11 | // // // | 52.751548 47.639795- 52.725150 47.617531- 52.723051 47.599384- 52.711474 | 7.20km 8.5km 6km 4km | Good Very Good Excellent Excellent |
| 8 9 10 11 12 | // // // 47.57325419- 52.73570346 | 52.751548 47.639795- 52.725150 47.617531- 52.723051 47.599384- 52.711474 47.57090894- 52.6810956 | 7.20km 8.5km 6km 4km 4.2km | Good Very Good Excellent Excellent Excellent |
| 8 9 10 11 12 13 | // // // 47.57325419- 52.73570346 // | 52.751548 47.639795- 52.725150 47.617531- 52.723051 47.599384- 52.711474 47.57090894- 52.6810956 47.567509- 52.742932 | 7.20km 8.5km 6km 4km 4.2km 1km | Good Very Good Excellent Excellent Excellent Excellent |
| 8 9 10 11 12 13 14 | // // // 47.57325419- 52.73570346 // | 52.751548 47.639795- 52.725150 47.617531- 52.723051 47.599384- 52.711474 47.57090894- 52.6810956 47.567509- 52.742932 47.59291283- 52.70756722 | 7.20km 8.5km 6km 4km 4.2km 1km 4km | Good Very Good Excellent Excellent Excellent Excellent No Signal |

Table 5. 2: Coordinates of ALT and ALR

According to Table 5.2, when ALT was placed at the first coordinate, ALR was tested at different points. The maximum distance between them was about 15 km, at which distance the signal was received without any delay or noise (line of sight). However, when ALR was placed at 13.5 km, the signal was delayed and was not clear. These issues appear to have been caused by obstacles such as and trees in the line of site.

Also according to the table, ALT was placed at the second coordinate (UC MUN St. John's, NL, Canada). The maximum distance for receiving data using ALR is about 4 km. Not surprisingly, when we expanded our distance to around 11 km, ALR could not receive the signal due to multiple obstacles blocking it. Obstacles were buildings, houses and trees.

Our field experiments showed that in order to obtain an optimal signal (i.e., without noise or delay) using ALR, the place used to install ALT should be higher than the destination. In addition, the presence of obstacles between ALT and ALR should be avoided. It is the opinion of the present author that ALT and ALR could work over a longer distance than tested if the area were an open space or ALT were installed on a high tower.

5.10 Cost Design

The items used to design the monitoring and data acquisition system should be of excellent quality but low-cost. Table 5.3 shows the main design requirements

accompanied by price. The prices were obtained from the amazon.ca website, and are listed in Canadian dollars.

| NO | Item | Quantity | Price |
|----|-------------------|-----------------|-----------|
| 1 | Current Sensor | 2 | 6.80 |
| 2 | Resistors | Package(600PCS) | 7.12 |
| 3 | Software (IDE) | | Free |
| 4 | software Putty | | Free |
| 5 | ALT and ALR | 1 | 49.36 |
| 6 | Micro USB | 1 | 7.0 |
| 7 | Old Computer | 1 | 140.0 |
| | | Total Cost | \$ 210.28 |

Table 5. 3: Cost Monitoring and Data Acquisition System

As shown, a high-quality monitoring system can be built for the relatively low price of CAD \$210. It should be noted, however, that the price is subject to change depending on the quality of the components being used. Major cost is computer if that is available or cell phone is used then cost will be less than CAD \$80.0.

5.11 Result and Discussion

In order to show the PV system data, we need to open a serial monitor of Putty. However, Putty settings must first be adjusted to show data on a monitor and to save it in a text file on a computer. The settings include: port of ALR and location text file on computer. Figure 5.8 illustrates these settings.



| Real PuTTY Configuration | | 8 23 |
|--------------------------|--|---|
| Category: Same Port | of ALR | |
| Category: Same Port | Basic options for your Puil Specify the destination you want to Serial line COM1 Connection type. Raw Telnet Raw Telnet Rogin Load, save or delete a stored sessi Saved Sessions Default Settings | TTY session connect to Speed 9600 SSH © Senal on Load Save Delete |
| Serial | Close window on exit: Always Never On | nly on clean exit |
| About Hel | p Open | Cancel |

Figure 5. 8: Putty settings

After the requirement settings have been adjusted and the "Open" button is hit, the Putty serial monitor will appear as shown in Fig. 5.9.

As mentioned earlier, our first aim is to monitor PV system data in real time over a long distance. This goal was achieved at a distance of 1 km. However, according to Table 2,

we can expand the distance between ALT and ALR as long as there are no obstacles between the points.

| 🚱 COM9 - PuTTY | | | | |
|---------------------|--------------------|---------------------|--------------------|-------------|
| Voltage-pv 13.42 | voltage-B 13.54 | current-pv 15.20 | Current-B 13.40 | Efficency ^ |
| 12.41 | 12.47 | 15.20 | 13.40 | 0.89 |
| 11.96 | 11.95 | 15.20 | 13.40 | 0.88 |
| 13.18 | 13.25 | 15.20 | 13.40 | 0.89 |
| 13.20 | 13.34 | 15.20 | 13.40 | 0.89 |
| 12.01 | 12.02 | 15.20 | 13.40 | 0.88 |
| 12.30 | 12.35 | 15.20 | 13.40 | 0.89 |
| 13.39 | 13.50 | 15.20 | 13.40 | 0.89∰ |
| 12.60 | 12.68 | 15.20 | 13.40 | 0.89 |
| 11.95 | 11.96 | 15.20 | 13.40 | 0.88 |
| 12.89 | 13.03 | 15.20 | 13.40 | 0.89 |
| | | | | |

Figure 5. 9 Serial monitor with data

Our second aim is PV system data acquisition and saving it in a text file on a computer. We can use these data to determine the specifications of the desired PV system. Figure 5.10 shows PV data in a text file.

| File Edit Format Yiew Help LORA Receriver Current-pv Current-B Efficency Yoltage-px Voltage-B current-pv Current-B Efficency Yoltage-px 11.89 15.20 13.40 0.889 12.98 13.09 15.20 13.40 0.89 13.99 11.96 15.20 13.40 0.88 11.96 11.97 15.20 13.40 0.88 11.96 11.97 15.20 13.40 0.88 11.96 11.97 15.20 13.40 0.89 12.00 12.07 15.20 13.40 0.89 13.29 13.39 15.20 13.40 0.88 11.97 11.94 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 13.39 13.47 15.20 13.40 0.88 11.97 11.95 15.20 13.40 0.88 < | result1 - Notepad | | | | | x |
|--|-------------------|---------------|------------------|--------------------|-----------|---|
| Uota Putty log 2017.08.20 16:29:39 Uotage-pv voltage-s current-pv Current-B Efficency 12.18 12.23 15.20 13.40 0.89 12.90 12.46 15.20 13.40 0.89 13.00 12.46 15.20 13.40 0.89 11.93 11.97 15.20 13.40 0.89 11.96 11.97 15.20 13.40 0.89 12.98 13.26 13.40 0.89 13.26 13.40 0.89 13.27 13.40 0.89 12.00 12.07 15.20 13.40 0.89 13.24 13.24 13.35 15.20 13.40 0.89 13.24 13.24 13.35 15.20 13.40 0.89 11.93 11.93 11.85 15.20 13.40 0.89 11.93 13.24 13.35 15.20 13.40 0.88 13.30 14.91 1.55 15.20 13 | File Edit Format | View Help | | | | |
| Loka Receiver voltage-pv voltage-s current-pv Current-s Efficency 12.18 12.23 15.20 13.40 0.89 11.90 11.89 15.20 13.40 0.89 12.19 15.20 13.40 0.89 12.98 13.09 15.20 13.40 0.89 11.93 11.98 15.20 13.40 0.89 12.98 13.09 15.20 13.40 0.89 12.00 12.07 15.20 13.40 0.89 13.26 13.38 15.20 13.40 0.89 13.29 13.39 15.20 13.40 0.89 11.97 11.48 15.20 13.40 0.89 13.29 13.39 15.20 13.40 0.88 13.99 13.53 15.20 13.40 0.88 11.97 11.94 15.20 13.40 0.88 13.39 13.16 15.20 13.40 0.88 < | =~=~=~=~=~=~ | =~=~=~= PuTTY | log 2017.08.20 1 | L6:29:39 =~=~=~=~= | | ~ |
| Voltage-Dv Voltage-B Current-pv Current-B Erricency 11.90 11.89 15.20 13.40 0.88 12.40 12.46 15.20 13.40 0.89 13.08 13.19 15.20 13.40 0.89 11.96 11.97 15.20 13.40 0.89 11.96 11.97 15.20 13.40 0.88 13.26 13.40 0.88 13.37 13.49 15.20 13.26 13.40 0.89 13.26 13.40 0.89 12.00 12.07 15.20 13.40 0.89 13.24 13.39 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.88 13.29 13.39 15.20 13.40 0.88 13.39 13.52 13.40 0.88 13.39 15.20 13.40 0.88 13.39 15.20 13.40 0.88 13.40 0.88 </td <td>LoRa Receiver</td> <td></td> <td></td> <td></td> <td></td> <td></td> | LoRa Receiver | | | | | |
| 12.18 12.23 13.20 13.40 0.83 11.90 12.46 15.20 13.40 0.89 13.08 13.19 15.20 13.40 0.89 12.98 13.09 15.20 13.40 0.89 11.93 11.97 15.20 13.40 0.88 13.37 13.49 15.20 13.40 0.89 13.26 13.34 15.20 13.40 0.89 13.29 13.29 13.40 0.89 13.29 13.39 15.20 13.40 0.89 11.93 11.88 15.20 13.40 0.89 13.29 13.39 15.20 13.40 0.89 11.97 11.94 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 11.97 11.95 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 11.99 11.95 15.20 13.40 | Voltage-pv | Voltage-B | current-pv | Current-B | Efficency | |
| 11.90 11.60 13.40 0.89 13.08 13.19 15.20 13.40 0.89 13.08 13.19 15.20 13.40 0.89 11.93 11.98 15.20 13.40 0.89 11.93 11.97 15.20 13.40 0.88 13.26 13.40 0.88 13.37 13.49 15.20 13.26 13.40 0.89 13.20 13.40 0.89 12.00 12.07 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.93 11.93 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.88 13.39 13.51 15.20 13.40 0.89 13.99 11.95 15.20 13.40 0.89 13.99 11.95 15.20 13.40 0.89 13.99 11.95 15.20 13.40 0.89 11.99 11.92< | 12.18 | 12.23 | 15.20 | 13.40 | 0.89 | |
| 13.708 13.709 15.20 13.700 0.869 12.98 13.09 15.20 13.40 0.889 11.93 11.97 15.20 13.40 0.889 13.37 13.49 15.20 13.40 0.889 13.26 13.38 15.20 13.40 0.899 13.26 13.39 15.20 13.40 0.899 13.29 13.20 13.40 0.899 13.29 13.20 13.40 0.899 13.29 13.20 13.40 0.899 13.29 13.20 13.40 0.899 13.29 13.20 13.40 0.899 13.29 13.20 13.40 0.889 13.29 13.20 13.40 0.899 13.29 13.20 13.40 0.899 13.99 13.53 15.20 13.40 0.88 13.39 13.47 15.20 13.40 0.89 13.99 13.147 15.20 13.40 0.89 13.37 13.41 1.40 0.89 | 12.40 | 12.46 | 15.20 | 13.40 | 0.88 | |
| 12: 08 13:00 15:20 13:40 0.859 11: 93 11: 98 15:20 13:40 0.88 13: 37 13:49 15:20 13:40 0.88 13: 26 13:49 15:20 13:40 0.88 13: 26 13:49 15:20 13:40 0.89 12: 00 12: 07 15: 20 13:40 0.89 12: 29 13: 39 15: 20 13: 40 0.89 13: 24 13: 35 15: 20 13: 40 0.89 11: 97 11: 94 15: 20 13: 40 0.88 13: 39 13: 52 13: 40 0.88 13: 39 13: 52 13: 40 0.88 13: 40 14: 40 0.89 14: 84 11: 85 15: 20 13: 40 0.88 13: 41 14: 85 15: 20 13: 40 0.89 14: 84 11: 85 15: 20 13: 40 0.89 15: 40 13: 40 0.89 11: 91 15: 20 13: 40 0.89 14: 95 11: 90 <td< td=""><td>13.08</td><td>13.19</td><td>15.20</td><td>13.40</td><td>0.89</td><td></td></td<> | 13.08 | 13.19 | 15.20 | 13.40 | 0.89 | |
| 11.93 11.98 15.20 13.40 0.88 13.97 13.49 15.20 13.40 0.88 13.26 13.38 15.20 13.40 0.89 12.00 12.07 15.20 13.40 0.89 11.93 11.88 15.20 13.40 0.89 13.29 13.39 15.20 13.40 0.89 13.29 13.39 15.20 13.40 0.89 11.97 11.94 15.20 13.40 0.89 13.29 13.53 15.20 13.40 0.88 11.93 13.53 15.20 13.40 0.88 11.93 13.53 15.20 13.40 0.89 13.00 13.16 15.20 13.40 0.89 12.68 12.282 15.20 13.40 0.89 13.37 13.47 15.20 13.40 0.89 12.68 12.282 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 12.68 12.02 13. | 12.98 | 13.09 | 15.20 | 13.40 | 0.89 | |
| 11.96 11.97 15.20 13.40 0.88 13.37 13.49 15.20 13.40 0.89 12.00 12.07 15.20 13.40 0.89 11.93 11.88 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.97 11.94 15.20 13.40 0.89 11.81 11.78 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 13.39 13.45 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 13.37 13.47 15.20 13.40 0.88 12.68 12.62 15.20 13.40 0.89 11.95 11.90 15.20 13.40 0.89 12.68 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 14.93 11.92 15.20 13.40 | 11.93 | 11.98 | 15.20 | 13.40 | 0.88 | |
| 13.37 13.49 15.20 13.40 0.89 12.00 12.07 15.20 13.40 0.89 11.93 11.88 15.20 13.40 0.89 13.22 13.39 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.97 11.94 15.20 13.40 0.89 13.39 13.53 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.89 13.39 13.53 15.20 13.40 0.89 13.99 13.16 15.20 13.40 0.89 13.99 13.47 15.20 13.40 0.89 11.99 11.95 15.20 13.40 0.89 12.68 12.82 15.20 13.40 0.89 12.68 12.62 15.20 13.40 0.89 12.468 15.20 13.40 0.89 12.468 15.20 13.40 0.89 12.68 12.62 13.40 0.89 | 11.96 | 11.97 | 15.20 | 13.40 | 0.88 | |
| 13.26 13.38 15.20 13.40 0.89 11.93 11.88 15.20 13.40 0.89 13.29 13.39 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.97 11.94 15.20 13.40 0.89 11.81 11.78 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.89 11.99 11.95 15.20 13.40 0.89 11.84 11.85 15.20 13.40 0.89 11.84 11.85 15.20 13.40 0.89 11.99 11.95 12.00 13.40 0.89 12.68 12.82 15.20 13.40 0.89 13.30 13.51 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.88 13.39 13.41 15.20 | 13.37 | 13.49 | 15.20 | 13.40 | 0.89 | |
| $\begin{array}{c ccccccccccccccccccccccccccccccccccc$ | 13.26 | 13.38 | 15.20 | 13.40 | 0.89 | |
| 11.329 11.839 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.97 11.94 15.20 13.40 0.88 11.81 11.78 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.88 13.00 13.16 15.20 13.40 0.89 11.84 11.85 15.20 13.40 0.89 13.39 13.53 15.20 13.40 0.89 11.84 11.85 15.20 13.40 0.89 11.85 15.20 13.40 0.89 11.99 11.85 15.20 13.40 0.89 11.99 11.85 12.02 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 12.08 12.02 15.20 13.40 0.89 12.19 15.20 13.40 0.89 11.93 12.78 12.88 15. | 11.02 | 12.07 | 15.20 | 13.40 | 0.89 | |
| 13.23 13.35 15.20 13.40 0.89 11.97 11.94 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.89 13.00 13.16 15.20 13.40 0.89 13.00 13.16 15.20 13.40 0.89 13.00 13.16 15.20 13.40 0.89 13.99 11.95 15.20 13.40 0.89 13.37 13.47 15.20 13.40 0.89 11.84 11.85 15.20 13.40 0.89 11.95 11.90 15.20 13.40 0.89 11.85 12.02 15.20 13.40 0.89 12.66 12.82 15.20 13.40 0.89 12.88 12.02 15.20 13.40 0.89 12.98 12.02 15.20 13.40 0.89 12.98 12.06 13.40 0.89 12.78 12.98 12.06 15.20 13.40 0.89 13.39 13.49 15.20 | 12 20 | 12 20 | 15.20 | 13.40 | 0.88 | |
| 11.67 11.94 15.20 13.40 0.88 11.81 11.78 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.89 13.00 13.16 15.20 13.40 0.89 11.99 11.95 15.20 13.40 0.88 11.84 11.85 15.20 13.40 0.88 13.37 13.47 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.89 12.78 12.88 15.20 | 13.24 | 13.35 | 15.20 | 13.40 | 0.89 | |
| 11.81 11.78 15.20 13.40 0.88 13.39 13.53 15.20 13.40 0.89 11.99 11.95 15.20 13.40 0.89 11.84 11.85 15.20 13.40 0.88 13.37 13.47 15.20 13.40 0.88 13.37 13.47 15.20 13.40 0.89 12.68 12.82 15.20 13.40 0.89 13.40 0.89 13.40 0.89 13.40 0.89 13.40 0.89 11.95 11.90 15.20 13.40 0.89 13.40 0.89 13.40 0.89 12.58 12.68 15.20 13.40 0.89 12.78 12.12 15.20 13.40 0.88 13.37 13.48 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 1 | 11.97 | 11.94 | 15.20 | 13.40 | 0.88 | |
| 13.39 13.16 15.20 13.40 0.89 13.00 13.16 15.20 13.40 0.89 11.99 11.95 15.20 13.40 0.88 11.84 11.85 15.20 13.40 0.88 12.68 12.82 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 12.68 12.82 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.88 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.55 15.20 13.40 0.89 11.56 13.52 13.40 0.89< | 11.81 | 11.78 | 15.20 | 13.40 | 0.88 | |
| 13.00 13.16 15.20 13.40 0.89 11.99 11.95 15.20 13.40 0.88 13.37 13.47 15.20 13.40 0.89 12.68 12.82 15.20 13.40 0.89 11.95 11.90 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 12.58 12.66 15.20 13.40 0.89 12.93 11.92 15.20 13.40 0.89 12.78 12.28 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.08 12.06 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 13.55 15.20 13.40 0.89 13.51 12.99 15.20 13.40 | 13.39 | 13.53 | 15.20 | 13.40 | 0.89 | |
| 11.99 11.95 15.20 13.40 0.88 13.37 13.47 15.20 13.40 0.89 12.68 12.282 15.20 13.40 0.89 11.95 11.90 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 13.40 0.89 13.40 0.89 13.40 0.89 13.40 0.89 12.58 12.68 15.20 13.40 0.89 11.93 11.92 15.20 13.40 0.89 12.08 12.12 15.20 13.40 0.88 13.37 13.48 15.20 13.40 0.89 12.08 12.06 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.49 15.20 13.40 0.89 13.49 15.20 13.40 0.89 13.49 15.20 13.40 0.89 13.41 13.23 <td>13.00</td> <td>13.16</td> <td>15.20</td> <td>13.40</td> <td>0.89</td> <td></td> | 13.00 | 13.16 | 15.20 | 13.40 | 0.89 | |
| 11.84 11.85 15.20 13.40 0.88 13.37 13.47 15.20 13.40 0.89 12.68 12.82 15.20 13.40 0.89 11.95 11.90 15.20 13.40 0.89 11.89 12.02 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 11.93 11.92 15.20 13.40 0.89 12.78 12.28 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.51 15.20 13.40 0.89 13.41 13.99 13.49 15.20 13.40 0.89 13.55 13.47 15.20 | 11.99 | 11.95 | 15.20 | 13.40 | 0.88 | |
| 13.37 13.40 0.89 12.68 12.82 15.20 13.40 0.89 11.95 11.90 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 12.68 12.02 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 11.93 11.92 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.88 12.78 12.88 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 13.37 13.48 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.51 14 13.23 15.20 13.40 0.89 13.45 15.20 13.40 0.89 13.45 0.89 13.55 13.47 15.20 <td>11.84</td> <td>11.85</td> <td>15.20</td> <td>13.40</td> <td>0.88</td> <td></td> | 11.84 | 11.85 | 15.20 | 13.40 | 0.88 | |
| 11.95 11.90 15.20 13.40 0.89 11.95 12.02 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 11.93 11.92 15.20 13.40 0.88 12.08 12.12 15.20 13.40 0.88 12.78 12.88 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.89 12.78 12.266 15.20 13.40 0.89 13.24 13.55 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 13.49 15.20 13.40 0.89 13.55 15.20 13.40 0.88 13.14 13.23 15.20 13.40 0.89 11.89 11.89 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.88 12.02 12.24 15.20 13.40 0.89 12.02 12.02 13.40 0.88 13.04 13.18 15.20 $13.$ | 12.57 | 12.82 | 15.20 | 13.40 | 0.89 | |
| 11.89 12.02 15.20 13.40 0.89 13.40 13.51 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 11.93 11.92 15.20 13.40 0.89 12.08 12.12 15.20 13.40 0.88 13.37 13.48 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.88 13.89 11.89 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.88 13.05 13.18 15.20 | 11 95 | 11 90 | 15.20 | 13.40 | 0.85 | |
| 13.40 13.51 15.20 13.40 0.89 12.58 12.68 15.20 13.40 0.89 11.93 11.92 15.20 13.40 0.88 12.08 12.12 15.20 13.40 0.89 13.37 13.48 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 13.51 14.49 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 | 11.89 | 12.02 | 15.20 | 13.40 | 0.89 | |
| 12.58 12.68 15.20 13.40 0.89 11.93 11.92 15.20 13.40 0.88 12.08 12.12 15.20 13.40 0.88 13.37 13.48 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.88 12.08 12.06 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.88 13.24 13.35 15.20 13.40 0.89 11.89 11.89 15.20 13.40 0.88 13.35 13.47 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.89 12.22 12.24 15.20 13.40 0.88 13.04 13.18 15.20 | 13.40 | 13.51 | 15.20 | 13.40 | 0.89 | |
| 11.93 11.92 15.20 13.40 0.88 12.08 12.12 15.20 13.40 0.89 13.37 13.48 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.89 12.08 12.06 15.20 13.40 0.89 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.56 11.56 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 13.05 13.18 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.04 13.18 15.20 | 12.58 | 12.68 | 15.20 | 13.40 | 0.89 | |
| 12.08 12.12 15.20 13.40 0.88 13.37 13.48 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.88 12.08 12.06 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.88 13.55 15.20 13.40 0.89 11.56 11.56 15.20 13.40 0.89 11.89 11.89 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.06 11.94 15.20 13.4 | 11.93 | 11.92 | 15.20 | 13.40 | 0.88 | |
| 13.37 13.48 15.20 13.40 0.89 12.78 12.88 15.20 13.40 0.89 11.89 11.91 15.20 13.40 0.88 12.08 12.06 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.89 11.56 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 13.55 15.20 13.40 0.88 13.14 13.23 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.90 11.88 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.02 15.20 13.40 0.89< | 12.08 | 12.12 | 15.20 | 13.40 | 0.88 | |
| 11.89 11.91 15.20 13.40 0.89 12.08 12.06 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.89 11.56 11.56 15.20 13.40 0.89 11.89 13.23 15.20 13.40 0.88 13.14 13.23 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 11.90 11.88 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.02 12.02 13.40 0.89 12.00 12.00 12.02 | 13.3/ | 13.48 | 15.20 | 13.40 | 0.89 | |
| 12.08 12.06 15.20 13.40 0.88 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.56 11.56 15.20 13.40 0.88 13.14 13.23 15.20 13.40 0.89 13.14 13.23 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 11.90 11.88 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.88 12.00 12.02 15.20 | 11 80 | 11 01 | 15.20 | 13.40 | 0.89 | |
| 13.39 13.49 15.20 13.40 0.89 13.24 13.35 15.20 13.40 0.89 11.56 11.56 15.20 13.40 0.89 11.89 11.89 15.20 13.40 0.88 13.14 13.23 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 11.90 11.88 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.89 12.22 12.24 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.88 12.00 12.02 15.20 13.40 0.89 12.00 12.02 15.20 | 12.08 | 12.06 | 15.20 | 13.40 | 0.88 | |
| 13.24 13.35 15.20 13.40 0.89 11.56 11.56 15.20 13.40 0.88 11.89 11.89 15.20 13.40 0.88 13.14 13.23 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 11.88 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.88 | 13.39 | 13.49 | 15.20 | 13.40 | 0.89 | |
| 11.56 11.56 15.20 13.40 0.88 11.89 11.89 15.20 13.40 0.88 13.14 13.23 15.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 11.90 11.88 15.20 13.40 0.89 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.89 12.22 12.24 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.88 | 13.24 | 13.35 | 15.20 | 13.40 | 0.89 | |
| 11. 89 15. 20 13. 40 0. 88 13. 14 13. 23 15. 20 13. 40 0. 89 13. 35 13. 47 15. 20 13. 40 0. 89 12. 19 12. 28 15. 20 13. 40 0. 89 11. 90 11. 88 15. 20 13. 40 0. 88 12. 95 12. 99 15. 20 13. 40 0. 88 13. 05 13. 18 15. 20 13. 40 0. 89 12. 22 12. 24 15. 20 13. 40 0. 88 11. 96 11. 94 15. 20 13. 40 0. 88 13. 04 13. 18 15. 20 13. 40 0. 89 12. 63 12. 74 15. 20 13. 40 0. 89 12. 00 12. 02 15. 20 13. 40 0. 88 | 11.56 | 11.56 | 15.20 | 13.40 | 0.88 | |
| 13.14 13.23 13.20 13.40 0.89 13.35 13.47 15.20 13.40 0.89 12.19 12.28 15.20 13.40 0.89 11.90 11.88 15.20 13.40 0.89 12.19 12.29 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.89 12.22 12.24 15.20 13.40 0.88 13.06 11.94 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.89 | 11.89 | 11.89 | 15.20 | 13.40 | 0.88 | |
| 12.19 12.28 15.20 13.40 0.89 11.90 11.88 15.20 13.40 0.88 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.89 12.96 11.94 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.89 | 13.14 | 13.23 | 15.20 | 13.40 | 0.89 | |
| 11.90 11.88 15.20 13.40 0.88 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.89 12.22 12.24 15.20 13.40 0.88 13.06 11.94 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.89 | 12.19 | 12.28 | 15.20 | 13.40 | 0.89 | |
| 12.95 12.99 15.20 13.40 0.88 13.05 13.18 15.20 13.40 0.89 12.22 12.24 15.20 13.40 0.88 13.06 11.94 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.89 | 11.90 | 11.88 | 15.20 | 13.40 | 0.88 | |
| 13.05 13.18 15.20 13.40 0.89 12.22 12.24 15.20 13.40 0.88 11.96 11.94 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.89 | 12.95 | 12.99 | 15.20 | 13.40 | 0.88 | |
| 12.22 12.24 15.20 13.40 0.88 11.96 11.94 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.88 | 13.05 | 13.18 | 15.20 | 13.40 | 0.89 | |
| 11.96 11.94 15.20 13.40 0.88 13.04 13.18 15.20 13.40 0.89 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.88 | 12.22 | 12.24 | 15.20 | 13.40 | 0.88 | |
| 12.63 12.74 15.20 13.40 0.89 12.00 12.02 15.20 13.40 0.89 (| 11.96 | 11.94 | 15.20 | 13.40 | 0.88 | |
| 12.00 12.02 15.20 13.40 0.88 | 12 63 | 12 74 | 15.20 | 13.40 | 0.89 | |
| < ISING 15140 0.00 | 12.00 | 12.02 | 15.20 | 13.40 | 0.88 | |
| ۲ ۲ | 12.00 | 12.02 | 13.20 | 13.40 | 0.00 | |
| ▼ b. 4 mm | | | | | | |
| ال ا | | | | | | - |
| | • | | III | | | • |

Figure 5. 10: PV data in text file

As can be seen, the data appear in four columns. The first column shows the PV voltage while the second column represents voltage battery. The other columns indicate PV current, battery current, and efficiency.

5.12 Conclusion

The main aim of this paper was to design a low-cost monitoring and data acquisition system using Arduino Lora. The design was applied on a small PV system to collect data pertaining to PV and battery current and voltage of PV as well as PV efficiency. ALT was installed at a PVS location and successfully collected data from the PV system, while ALR was placed in a control room located approximately 1 km from ALT. The purpose of ALR was to receive data and show these data on Putty's serial monitor. The data was then saved on the computer as a text file. Maximum range was found at 15Km.

The total cost of this system was around CAD \$150, which covers all necessary system components. The power consumption of the system was calculated to be about 1.125W.

5.13 Acknowledgment

The authors would like to thank the Libyan government for providing graduate student funding for this research.

5.14 References

[1] Benghanem, M. "Measurement of meteorological data based on wireless data acquisition system monitoring." Applied energy 86.12 (2009): 2651-2660.

[2] Green, Martin A., et al. "Solar cell efficiency tables (Version 45)." Progress in photovoltaics: research and applications 23.1 (2015): 1-9.

[3] Oozeki, Takashi, et al. "An evaluation method of PV systems." Solar Energy Materials and Solar Cells 75.3 (2003): 687-695.

[4] Nishioka, Kojiro, et al. "Evaluation of output performance of various photovoltaic systems in the hokuto mega-solar project." Telecommunications Energy Conference (INTELEC), 32nd International. IEEE, 2010.

[5] Tahri, Ali, Takashi Oozeki, and Azzedine Draou. "Monitoring and evaluation of photovoltaic system." Energy Procedia 42 (2013): 456-464.

[6] Ramachandra, T. V., et al. "Solar energy decision support system." International Journal of Sustainable Energy 24.4 (2005): 207-224.

[7] Blaesser, G. "PV system measurements and monitoring the European experience." Solar energy materials and solar cells 47.1-4 (1997): 167-176.

[8] Wilshaw, Alison R., Nicola M. Pearsall, and Robert Hill. "Installation and operation of the first city centre PV monitoring station in the United Kingdom." Solar Energy 59.1-3 (1997): 19-26.

Chapter 6

Conclusions and Future Works

6.1 Conclusion

SCADA software can be divided into two types. First, proprietary, Companies develop proprietary software to communicate to their hardware. However, these systems are expensive and overwhelming reliance on the supplier of the system. Second, open software systems, they have gained popularity because of because of the interoperability they bring to the system. These systems are inexpensive and they are could be programmed without supplier company.

In this thesis, a Low–Cost SCADA system has been designed with open source technique. Four studies have been done, and they were implemented on a small photovoltaic system. First, a Low-Cost SCADA system has been designed by using Arduino Uno, Raspberry-pi, and ubidots serve. Chapter 2 includes more details include: hardware design, used sensors, used software, hardware setup, and results. Second, a Low-Cost SCADA system has been designed as well, however in this case is based on Arduino Uno and Reliance SCADA software. Chapter 3 explains following: hardware design, used sensors, hardware setup, software requirements, code, communication system, cost of the design, and the results. Third, microcontroller ESP32 has been used to

design a Low-Cost web server to monitor a small photovoltaic system. This designed is based on ESP32, Wi-Fi, and SD card. Chapter4 shows steps design a Low-Cost web server include: PV system description, system design, SD card reader, Wi-Fi, sensors, hardware setup, software used, power needs for design, implementation and result. Finally, in this study, Arduino LoRa has been used to design a low-cost monitoring and data acquisition system for a small photovoltaic system. In this design is just based on Arduino Lora. In addition, Arduino Lora has been tested in many locations in St. John's, Newfoundland and Labrador, Canada. The maximum desistance founded to get excellent signal at 15 Km, more details find in chapter 5 include: PV system, hardware configuration, transition station design, hardware setup, receiving station design, power needs for this design, software used, range testing, cost design, and result and discussion.

This research indicates ESP32 based system is the lowest cost SCADA option and requires least power. If monitoring at a distance is required then we recommend Arduino LoRa based system.

6.2 Future Works

• Arduino Uno, Raspberry-Pi, and Ubidots server have been used to design Low-Cost SCADA system for a photovoltaic system. Raspberry pi can be used with the server (Ubidots) without Arduino, but you need an external device to read analog measurements. In addition, Arduino Uno can be connected with Ubidots for many applications.

- Arduino Uno, and Reliance SCADA software have been used for Low-Cost SCADA system for a small photovoltaic. The data were transferred from Arduino to user interface on a computer by using USB cable. The data can be transferd by Wi-Fi, and user interface can be connected with internet to open it from anywhere and anytime.
- A Low Cost Web server has been designed to monitor and collect data from a small photovoltaic system; the data was downloaded as text file by using a computer or a cell phone; this work was based on ESP 32, SD card, and Wi-Fi. Arduino Uno or microcontroller ESP8266 can be used to design the same system, and you will get the same result. However, they might be different in processing speed of data, and power consumption.
- Arduino Lora has been used to monitor and collect data from a small photovoltaic system. The desistance between PV location and control room was about 1km. Moreover, Lora has been tested in many places in St. Johns, Newfoundland and Labrador, Canada. The maximum range is to get excellent signal at 15km. Arduino Lora can be tested in further distance more than 100Km, but without obstacles between receiver LoRa and transmission LoRa.
- Arduino Uno, ESP32, Raspberry Pi can be used for other SCADA applications.