# A Communication Method for Remote Control of Grid-tied Converters

by

© **Kumbalatara Arachchige Terashmila Lasagani**

A thesis submitted to the School of Gradate Studies in partial fulfillment of the requirements for the degree of Master of Engineering.

Faculty of Engineering and Applied Sciences

Memorial University

May, 2018

St. John's, Newfoundland and Labrador, Canada

# Abstract

This thesis presents selection, design, development, and validation of a communication method for remote control of a grid-tied inverter. There is a gap in the smart grid field for a secure and reliable communication method. To fulfill this gap three low-cost wireless such as LoRa, Radio Teletype, and UHF/ VHF based data communication technologies were tested, compared and presented in the thesis. Based on the test results, range, data transfer rate, and power consumption, LoRa based communication are selected as the most suitable method to satisfy the problem. Security and reliability issues have been identified in the LoRa based communication. Therefore, an encryption algorithm is developed to improve the security of the LoRa communication and data have been logged into a local data storage to improve the reliability. To increase the reliability of the developed system further, Power Line Carrier (PLC) communication link has been combined in parallel to the LoRa communication link. To evaluate the reliability of the developed system fault tree and Monte-Carlo simulation based reliability model has been proposed. Based on available data and assumed data reliability of the system had been calculated. Results obtained by testing the developed system with an Inverter which is being designed at the University of New Brunswick, presented in the thesis verify the operation of the development. Therefore, this study substantially contributes to the field of SCADA systems by developing a low cost and reliable communication method.

# Acknowledgements

First and foremost, I would like to express the deepest appreciation to my supervisor, Dr. Tariq Iqbal, for his endless guidance, valuable advices and generous support throughout the research work and the financial support provided. I greatly acknowledge with gratitude the guidance, valuable advices and the enormous support provided by my co-supervisor Dr. George Mann. I also acknowledge the funding provided by the NSERC Energy Storage Technology Network (NESTNET) fund and the School of Graduate Studies, Memorial University.

Furthermore, I highly appreciate the support given by the Office of the Associate Dean (Graduate) Faculty of Engineering and Applied Science, Memorial University and Dr. Bala Venkatesh, the principal investigator of the NESNET fund, Dr. Liuchen Chang, the theme two leader and the network manager Karen Ho-Cespedes for their support.

I acknowledge the support provided in preparation of the manuscripts by Dr. Thumeera Wanasinghe and Dr. Premkumar Thodi. I consider it an honor to work with all the people in the Intelligent Systems Lab. And my sincere thank goes to lab technicians in the electrical lab who provided all equipment without hesitation.

Finally, I would like to thank my parents and my husband, Sarinda Jayasinghe. Thank you!

# Table of contents

# List of tables

# List of figures

# List of symbols

| | |
|---|---|
| $d$ | distance between transmitter and the receiver |
| $h_t$ | hight to the Transmission antenna |
| $h_r$ | hight to the receiving antenna |
| $\lambda$ | wave length of the signal |
| $P_r$ | receiving Power for the signal |
| $d_0$ | breakdown distance |
| $P_{r_{d0}}$ | receiving Power at $d_0$ |
| $P_t$ | transmitting Power of the signal |
| $n$ | path-loss exponent |

# List of abbreviations

| | |
|---|---|
| PLC | Power Line Career Communication |
| LoRa | Long Range communication |
| QoS | Quality-of-Service |
| VHF | Very High Frequency |
| UHF | Ultra High Frequency |
| SCADA | Supervisory Control and Data Acquisition |
| AMI | Advanced Metering Infrastructure |
| DSL | Digital Subscriber Lines |
| EMI | Electro-Mechanical Interferences |
| HAN | Home Area Network |
| NPV | Net Present Value |
| WPAN | Wireless Personal Area Network |
| FHSS | Frequency Hopping Spread Spectrum |
| WLAN | Wireless Local Area Network |
| DSSS | Direct Sequence Spread Spectrum |
| CSMA-CA | Carrier Sense Multiple Access- Collision Avoidance |
| OFDM | Orthogonal Frequency Division Multiplexing |
| MIMO | Multiple Input Multiple Output |
| LMR | Land Mobile Radio |
| OWC | Optical wireless communication |
| WiMAX | Worldwide Interoperability for Microwave Access |
| LoS | Line of Sight |
| SNR | Signal to Noise ratio |
| RSS | Received signal strength |
| V2V | Vehicle to Vehicle |

# Chapter 1

# Introduction and Literature Review

## 1.1 Introduction

Due to challenges like growing population and increasing demand for energy, governments and utilities have been provoked to address these issues with a new approach to electricity generation. To address those concerns, renewable energy generators have emerged as a promising technology[1]. The integration of such renewable generators into the main grid is transforming the present power system into a large scale Distributed Generation (DG) system which is incorporating different technologies [1], [2], [3]. Therefore, the number of inverters connected to the main grid is increasing, and the communication technologies have become essential.

Along with the new concept of a smart grid, several communication technologies have been employed in communication between inverters and central location. This central location can be a single computer serving as a server and SCADA system with sophisticated equipment depending on the energy source capacity. Figure 1.1 shows a typical arrangement of a SCADA system. Both wired and wireless media communication technologies have been used for data transmission. Depending on the situation wireless communication methods show some advantages over wired techniques like low-cost infrastructure and ease of connection to difficult or unreachable areas. When it comes to long ranges wireless systems fails to perform well and when it tries to increase the range cost again going up. Therefore, this thesis addresses the issue of the requirement for a low cost, long range communication method.

This research is conducted to explore available communication methods for remote monitoring and control of inverters and to select the most suitable way considering the range, cost, security and the reliability. First two chapters explain the selection process of the most suitable communication method. After testing in the field state of the art technology, LoRa has been selected as the most suitable communication method. Chapter three presents a novel algorithm developed to improve the security and the reliability of the LoRa communication channel. Monte-Carlo simulation based reliability model is designed to analyze the reliability of the developed device, and that explains in chapter 4.

## 1.2 Literature Survey

The primary objective of this research is to develop communication method for a SCADA which is used for monitoring and controlling of a grid-tied inverter. To connect the knowledge and provide a comprehensive overview of the current state of the art, this chapter provides a systematic literature review of the existing communication methods used in SCADA systems. In general, with the survey we aim to achieve the following objectives:

- Provide a basic classification framework in the form of a taxonomy to classify

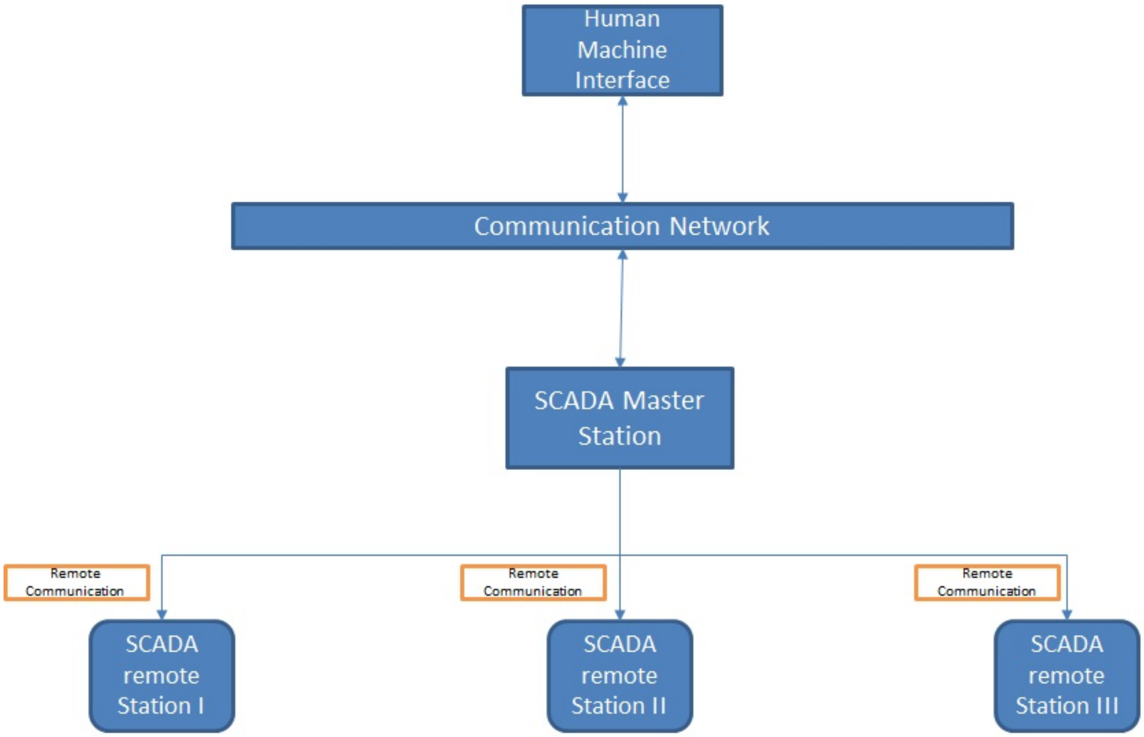


Figure 1.1: Problem definition I

existing communication technologies.

- Provide an overview of the current state of the art in the communications technology.

- Point out current trends, gaps, and directions.

A large number of papers have been examined from multiple research subareas, which are published in electrical engineering journals and conferences. As the part one and two, a detailed survey has been carried out to identify communication technologies that can be adapted for monitoring and controlling of an Energy Storage System. Available techniques are then be categorized based on the communication medium. In part three of the literature survey, features need to be embedded in a communication link used for a SCADA system are being reviewed.

## 1.3 Available Technologies

This section describes communication technologies that are being utilized for SCADA systems. This section mainly broke into two parts, wireless and wired.

### 1.3.1 Wired Technologies

There are a handful of wired technologies that are being used in SCADA applications where Power Line Communication can be recognized as the most suitable wired technology with existing infrastructure for smart grid applications. And here many publications can be found where Digital subscriber lines are being used in SCADA applications. The state of art technologies such as fiber optic is used for SCADA applications. In the following sections, all those areas are being reviewed as a part of the literature survey.

#### Power-line Communication

In this wired technology, the existing power lines are being used to send signals at high speed like 2-3 Mb/s rate from one to another device. Powerline communication (PLC) is an important technique because of the reasons like it has been the first technology which used for communication with the electricity meter since it has a direct connection with the meter. Also, this has been capable enough to successfully deploy the advanced metering infrastructure (AMI) in urban areas while other technologies were struggling to satisfy the requirements of the utility.

M. Y Zhai et al. [5] present the measurement results of channel properties of LV PLC systems after giving a general overview of the topologies for the typical LV distribution networks in China. Their results show that only $2-$ 10 MHz frequencies can be used where it is impossible to use frequencies from $10-$ 20 MHz frequencies due to attenuation. But in their work, they have not addressed the presence of transformer issue in PLC.

A well-structured survey on PLC can be found in the 5th chapter of smart grid communications and networking. Sara. B. et al. [6] give a comprehensive overview for PLC. In general, using power lines, smart meters and data concentrators have been connected to the PLC network and using cellular technologies those data is being transferred to the data center. Using this method, by combining an electrical device such as a power line smart transceiver-based meter to the power line, data can be transmitted to a central location. Following are some examples of PLC networks which have been used in several countries for data communication.

Linky meter project: In France, there has been a project which was updating 35 million traditional meters to smart meters and PLC network is being used for data communication from intelligent meters to data concentrator, and GPRS technology is being used to transfer data from data concentrator to the utility‘s data center. ENEL, the Italian electric utility: PLC network has been chosen to move data from smart meters to data concentrator and GSM technology has been used to send data from the concentrator to the data centers.

Zhiming. W. et al. [7] proposed a novel method using PLC by combining PLC and wireless technology to solve the uplink channel problem in non-signal or poor-signal areas encountered in the on-site implementation.At present, wireless communication modules of the concentrator are installed as discrete modules. The uplink wireless communication and the local power line communication were achieved by the internal gateway of the concentrator.

**Discussion** Furthermore, this technology also comprises its advantages as well as some disadvantages which need to be considered while selecting PLC as a communication technology in a project. As an advantage, since PLC can be connected directly to the existing power grid infrastructure, the installation cost which needs to be spent on the infrastructure in PLC is low compared to other technologies, mainly in smart grid deployments can be recognized. PLC has been popular technology because of the reasons such as availability, cost-effectiveness and standardization efforts on PLC networks. Data transmissions are broadcasted in nature for PLC. Hence, the security aspects are critical. Confidentiality, authentication, integrity, and user intervention are some of the essential issues in smart grid communications. HAN application is one of the most significant applications for PLC technology [8]. Moreover, PLC technology can be well suited to urban areas for smart grid applications, such as smart metering, monitoring, and control applications, since the PLC infrastructure is already covering the sectors that are in the range of the service territory of utility companies.

On the other hand, there are some technical difficulties also in the PLC networks. Since PLC rely on power line networks; the channel modeling is complicated due to harsh and noisy environmental conditions of the transmission medium of power lines.

Furthermore, the quality of the signal transmitted over the power lines is adversely affected by the features of the network topology, the number and type of the devices connected to the system, and the distance between the transmitter and receiver. In the end, once the pros and con study regarding the PLC, it appears that PLC is not suitable for data transmission due to the reasons like the sensitivity to the disturbances and dependency on the quality of data transmission. However, hybrid solutions can be made using PLC along with technologies like GPRS or GSM. This method cannot be used if there are transformers in the transmission system.

### Digital Subscriber Lines

DSL can be recognized as one the diminishing technology after the introduction of fiber optics. The high-speed digital data transmission technology which uses the wires of the voice telephone network is called as Digital Subscriber Lines (DSL). Since the infrastructure is already existing, the installation cost is less with this technology also. That has become a reason for selecting this technology by many companies for their smart grid projects. Following are some examples of the deployments of DSL in smart grid projects. Zhang D. et al. proposed a power system environmental parameter monitoring, data acquisition system using ADSL and TCPIP protocols. This paper offers a low cost, and high-performance program monitors the integrity of an entire distribution network. The framework adopts embedded Ethernet-based peer-to-peer architecture [9].

**Discussion** Since any technology is including its pros and cons depends on the application, it is typical for DSL technology as well. When it comes to the smart grid deployment with smart metering and data transmission, the DSL technology is the first choice selected by the electricity suppliers due to its inherent features such as widespread availability, low cost, and high bandwidth. However, for the mission-critical applications, the DSL is not a suitable choice because of its low reliability and high potential downtime [8]. Except that the dependency on the distance and lack standardization makes the DSL not acceptable for such applications. Moreover, the wired DSL based communication system deployments in rural areas may not be adequate due to the high cost of fixed infrastructure since these systems require communication cables to be installed and it needs a regular maintenance as well.

### Optical-Fibre Communication

This wired communication technology is existing in the market now for about decade of time. And these optical-fiber systems have replaced traditional electrical wire communication especially in cases like a long distance, high demand applications and high speed required applications. This link can be considered as a promising technology which can accommodate the demanding data requirements of the smart grid.

When considering the capacity of the fiber optic link, these channels are with extremely high capacity. The currently available networks are with the capability to provide the bit rates up to 10 Gbps using single-wavelength transmission and using wavelength division multiplexing (WDM) these links offer data rates of 40 Gbps to 1600 Gbps. Further, due to these high capacities, delays like channel access and queuing are also minimal in optical-fiber systems. While standard T1 or coaxial communication systems require repeaters at every 2km distance, these optical fiber networks need repeaters at 100 to a 1000km interval to transmit the signal along the fiber, to ensure that the signal is not distorted or weaken. Therefore, optical fiber systems are very cost effective due to this attenuation quality of these systems. Another appealing feature of the optical fiber network is these lines can be installed in environments even with high EMI. That implies these optical-fiber networks can be used in situations like high/medium voltage substations, alongside utility lines, power lines and railroad tracks. Further, there is no crosstalk between other optical-fiber lines even when they run alongside for a long distance unlike to electrical transmission lines. Optical fiber links can be considered as a good option when it comes to safety and security. For high-security deployments, these links are well matched since these optical-fiber links are not electromagnetically radiating and unauthorized physical access is also difficult without disrupting the signal. Further, these lines can be installed in high flammable or explosive environments as well since they do not cause any sparks [6].

**Discussion** Although these links are comprising many technical advantages, the limiting factor is its high cost of installation, fiber material, and transceivers [8]. However, with the time, prices are seeming to be decreasing, and the fiber backbones installation is seeming to be profitable since the broad bandwidth of the optical fiber can be shared among many applications like broadband access of residential customers. Even though above facts appear to be green lights for the installations of optical fiber networks concerning cost, infrastructure developments within cities are comparatively difficult and time-consuming. As a recent example the smart grid city project in Boulder, Colorado can be considered. This project has been over-budgeted due to cost overruns in the installations of optical-fiber networks. Therefore, this project has marked a negative press attention and customer dissatisfaction regarding the smart grid deployments in general. Fiber optic based communication can be considered as the most suitable option for real-time control. But the high installation cost will eliminate this method being used.

### 1.3.2 Wireless Technologies

The main drawback of a wired communication link has the significant installation cost. It would be wasteful to invest that much amount of money for communication infrastructure for rural areas with small data handling requirements. Comparing to wired technologies wireless offers significant benefits such as low-cost installations, rapid deployment, easy user access, and mobility.

### Cellular Network Communication

Some publications can be found on this wherein a smart grid concept, the communication between smart meters and the utility and between remote nodes is very much important. And the existing cellular network can be considered as a good option for that since the current infrastructure assists to the utility to reduce both the installation cost and time of new infrastructure for communications. Also, the cellular network allows for spreading of smart grid application into a vast area environment. The available mobile options for smart metering deployments are 2G, 2.5G, 3G, WiMAX, and LTE and following are some of the examples of usage of cellular networks into the smart grid applications.

Zhou et al. present a communication system using general packet radio service(GPRS) and code division multiple access (CDMA) wireless communication networks in SCADA system. In their developed approach, GPRS and CDMA used one at a time per the signal strength. This paper does not take the availability of GPRS/ CDMA signal into consideration [10].

C. Fu et al. present a Supervisory Control and Data Acquisition System (SCADA) based on wireless communication mechanism combined with General Packet Radio System (GPRS) platform and wireless sensor networks (WSN). The sensor network is being developed using Zigbee. In this approach, the range of the ZigBee and a monthly subscription fee of GPRS link must be taken care of [11].

**Discussion** Cellular networks also have advantages as well as disadvantages when using for smart grid applications. There is no additional cost of infrastructure for utilities if the cellular network is used for smart grid deployments since the cellular network already exists and that is one of the major advantages of using this. Although a significant amount of data is gathered due to data gathering on smaller intervals, the cellular networks are capable enough to provide sufficient bandwidth for such applications. Due to widespread and cost-effective benefits, the cellular networks are one of the frontline options for smart grid communication applications. Moreover, with the strong security controls in cellular networks, the data is secured on cellular networks. One of the prominent advantages of cellular networks compared to other solutions is that due to almost 100% coverage of cellular networks even in urban or rural areas, it is easier to spread the smart grid concept into low-density areas as well with the use of cellular networks using as a communication option. Other than that, both GSM and GPRS support AMI, Demand Response, Home Area Network (HAN) applications as well. The features of GSM like anonymity, authentication, signal protection and user data protection are the strengths of GSM regarding the security. Also, cellular networks are with features like lower cost, better coverage, lower maintenance costs and fast installation which makes it as the most suitable solution for deployments such as HAN, outage management, demand response management, etc. However, cellular networks are shared by customer market, and due to that, there can be network congestions or decrease in network performances which cannot be acceptable in some mission-critical power grid applications where it requires a continuous communication. Such reasons thrust the utility to build their infrastructure for communication purposes. Also, cellular networks may not be able to provide guaranteed service during abnormal situations like a wind storm an, etc. which can be considered as another disadvantage of mobile networks on smart grid deployment. And there will be availability problems in some places in the world, especially in rural areas. This method is widely used these days. Concerning this project, the main issue with this approach is availability and the cost. Allocation of money for a monthly fee will reduce the NPV of any system. Lower NPV becomes a significant factor when considering this option for a small project like this.

### Bluetooth over IEEE 802.15.1

Bluetooth is a designed standard for short-range wireless radio systems to organize as a Wireless Personal Area Network (WPAN). The cables used to connect secondary components into the computer such as a mouse, keyboards, joysticks, and printer can be replaced, with the Bluetooth. Piconet and Scatternet are the two connectivity topologies defined using the Bluetooth. By connecting two or more devices like modern cell phone or PDA via Bluetooth, the Piconet topology works. On the other hand, scatter net is a type of connection which interconnects some piconets to support to communicate with more than eight devices. These characteristics of the Bluetooth allow using a communication protocol in smart grid deployments such as HAN or home automation.

Conti et al. propose a method combining Bluetooth and GPRS to make a vehicle to vehicle communication protocol and to make a smart grid. But they do not address the range issue of the Bluetooth based systems[12].

**Discussion** When considering the advantages of the Bluetooth, it is capable of providing reliable and secure service due to the use of Frequency Hopping Spread Spectrum (FHSS) as an access technique as well as due to the in-built RC4 algorithm. Other than that, Bluetooth requires little power, and there is no need of dedicated transmitter since Bluetooth is an embedded feature in cellular phones can be considered as advantages of Bluetooth compared to other methods of communication [8].

However, the limited coverage area is the primary challenge of Bluetooth. Network area covered by Bluetooth is limited to 100m. In addition to that, increasing number of nodes which can be served from Bluetooth also limited. Though the scatter net configuration provides the connection to several numbers of nodes, there is a limitation on expanding the number of nodes.

### Wi-Fi (WLAN Over IEEE 802.11)

To get an uninterrupted communication with high data rates, the Wireless Local Area Network (WLAN) over IEEE 802.11 is a proper option. Also, this method provides interference depression, multiple user access along with other technologies. To achieve that target, the spread spectrum technologies such as Direct Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FHSS) and access technique like Carrier Sense Multiple Access- Collision Avoidance (CSMA-CA) have assisted this service. This service is yielding a maximum data rate of 11 Mbps with the use of DSSS modulation, and this operates on 2.4 GHz ISM frequency [8]. The use of Orthogonal Frequency Division Multiplexing (OFDM) in packet-based communication has made it possible to achieve higher data rates like 54 Mbps on ISM band. Further, in the IEEE 802.11n standard, Multiple Input Multiple Output (MIMO) technology is being used to achieve the data rates up to 600 Mbps. Nevertheless, the coverage area provided by this service is limited 100m which limits this in only medium range smart grid deployments like Home Area Network (HAN), remote monitoring, distribution protection systems and Advanced Metering Infrastructure (AMI).

**Discussion** The high flexibility provides for application specific deployments from this service using two modes of operation namely Ad-Hoc and Infrastructure can be considered as one of the significant advantages of using this service. Also, the ability of optimizes the network performance using load balancing of the network is another crucial benefit [8]. Also, with use of MIMO, this service provides the enhanced data rates from 54 Mbps to 600 Mbps. Further, there are some developments in WLANs to improve the security like, IEEE 802.11i standard which is a security specific standard build using Advanced Encryption Standard (AES). To significantly enhance the coverage area in both urban and rural areas, WLAN can operate in mesh networking mode and can use the multi-hop routing. Also, high data rates, the mobility of devices, ease of installation and cost effectiveness of WLAN helps to achieve secure and reliable communication. WLAN provides safe and reliable communication over a network facilitating high data rates, the mobility of devices, easy installation and cost-effective deployment.

Though the network span of WLAN can be increased using mesh topology, it adds up to routing complexity. As the packets, must pass through each node; an extra overhead is required which causes a reduction in bandwidth utilization. Moreover, in smart grid applications such as Home Area Network (HAN) or Advanced Metering Infrastructure (AMI), the crucial metering information must be passed through every access point. WLAN requires robust encryption techniques for data security. Hence, broad coverage area like cellular technologies is a challenging task in WLAN, especially in urban areas. Also, if mesh network is implemented then network capacity, fading and trade-off between reliable and flexible routing will be the most significant challenges faced, along with the interference and network coexistence.

### Radio Teletype

Traditional Land Mobile Radio (LMR) networks are being used for voice communications. But over decades same LMR devices used for data communications as well [13]. Before the Internet coming to the arena hobbyist used this Radio Teletype technology for a keyboard to keyboard communications.

### VHF/ UHF Wireless Data Transmission

Radio communications rely on the allocation of suitable carrier frequencies, done in the UK through the Radio Communications Agency (RA). All frequencies where transmission power more than 110 mW is employed are allocated. In the UK, the 457.5-458.5 MHz and 463.0-464.0 MHz sub-bands of the 450-470 MHz telemetry and Tele control band are available for SCADA applications, covered by the MPT 141 1 specification. Use of transmission powers less than ten mW are not regulated as such provided equipment use is specified and the equipment has Type Approval, but some have been allocated for particular use. MPT 1329 covers the UHF 458.5-458.8 MHz band and is available for low power industrial remote control and telecommand services. More recently the RA has allocated the 183.51 84.5 MHz frequency band for Utility metering purposes (although the use of the band for additional Utility customer services is not precluded). There are limitations in the range of most UHF radio communications, but hybrid systems utilizing telephone connections to a 'collector' station can overcome this drawback. The UK El has its band allocated for Private Mobile Radio (PMR) - a cellular system for operational use - using paired frequencies of 139 and 148 MHz. One company has been trailing use of the control (data) band of the PMR system for a DA application.

### Optical Wireless Communication (OWC)

OWC can be considered as a replacement for RF communications since this is more like RF wireless. However, light replaces the radio waves in this format, and free-space optical transceivers replace the antennas. Though RF and OW look similar from the surface OW exhibits some significant features compared to RF such as OW links are broadband, optical frequencies are in the infrared and visible spectrum which are free band neither regulated nor licensed. Further, cheaper optical components and low power consumption of these elements compared to high-speed RF components are several appealing attributes of OW. Most importantly, unlike RF, OW links do not suffer from multipath fading as well as no inference problems with RF-sensitive electronic devices. Nevertheless, when considering the facts like coverage area and mobility, the OW deployments have been limited.

### WiMAX (IEEE802.16)

Worldwide Interoperability for Microwave Access (WiMAX) technology can be regarded as a primary wireless broadband technology which is proposed for Wireless Metropolitan Area Network (WMAN). In this technique, long distance coverage and very high data rates are appealing features. WiMAX can cover the area of 50 km while handling data rate of 70 Mbps. However, the coverage area varies depends on Line Of Sight (LOS) and none line of sight deployment. To enhance the range in Non-Line of Sight applications, Orthogonal Frequency Division Multiple Access (OFDMA) interface is being used in mobile WiMAX [8]. Since WiMAX technology can provide efficient broadband access service even for a longer distance, it is called as Last Mile technology. Hence this technique is considered as a good substitution for wired technologies such as cable modems, Digital Subscriber Line (DSL) and T1/E1 links.

**Discussion** WiMAX service also has its advantages as well as disadvantages when using for the communication purposes. For wireless meter reading in Automatic Metering Infrastructure (AMI), WiMAX can be considered as a good alternative due to its inherent features such as long distance coverage and sufficiently high data rate. Real-time pricing models can be developed based on real-time energy consumption using WiMAX technology, most importantly without any human involvement. To enhance the reliability of the power system, electric outage detection and restoration is critical factors and to achieve those targets WiMAX can be used to develop a reliable two-way communication. One of the most appealing features of this technology is this WiMAX service using adaptive modulation technique, and thus it can have an optimized performance at any desired Signal to Noise ratio (SNR). Therefore, even in a harsh environment with low Received signal strength (RSS), this service is capable enough to provide a little operation. Further, seamless communications, low cost of deployments, appropriate bandwidth, high data rates, well-developed security protocols, extended network span and scalability features can be considered as the other advantages of this WiMAX network [8]. The high cost, when used for small-scale deployments due to the need for dedicated hardware for WiMAX frequencies, can be considered as one of the significant disadvantages of this service. Also, the AMI deployments using WiMAX service cannot be implemented in urban areas since the WiMAX frequencies above 10 GHz cannot penetrate through the obstacles. Further, if it tries to lease third party low frequency band since lower frequencies and unlicensed ISM band are already crowded, it is also further added to the cost which can be regarded as another disadvantage.

### Satellite Communication

Satellite systems are capable enough to provide a broad coverage even for an area in which terrestrial communication systems are failed to provide their services. Because of that, this network has been used for voice and data communications for many years. As an example, customized small aperture terminals (VSATs) are commercially available for remote monitoring and controlling purposes of substations, and currently, they have been deployed already by some utilities for the monitoring purposes of substations which are situated in rural areas. Also, critical data can be redirected over satellite links in a case of failure of physical connections which means these links are being used as a secondary method for communication for utility communication networks [8]. There are three main categories of satellites namely geostationary earth orbit satellites (GEO), middle earth orbit satellites (MEO) and low earth orbit satellites (LEO). GEOs are tracking over the equator in the directions of earths rotation. Since it's installed at an altitude of 35,786 km, the one-way latency is about a quarter of a second which is not an acceptable level for most of the applications. However, LEOs are located at 160-2000 km above the earth surface, and therefore its latency in other words round-trip delay is comparable to terrestrial networks. MEOs which are like GPS satellites are being located In between LEO and GEO level. As attracting point regarding the satellites, ease of setting up a connection by installing a transceiver at the desired location can be considered because in some cases these can self-installed. Further, satellite links can be taken as a cost-effective solution for areas where there is no other communication infrastructure is available. However, the light of sight requirement of the satellite link is the challenging factor for setting up a connection since the environmental feature can change with growing foliage. Also, characteristics of these relationships are mostly varying per the weather conditions [8]. Moisture and precipitation primarily affect to the throughput of these links and therefore keeping large rain margins, reducing bit rates during rainfall and adaptive power control are some of the techniques deployed by these links to overcome the fading during the rain. Also, lower data rates, higher initial cost of satellite modems, high subscription fees are some other disadvantages with satellite links.

### Zigbee

The ZigBee Alliance developed the ZigBee protocol with the intention for low data rate and short range (10m-100m) wireless networking applications. Further, ZigBee has defined three types of devices namely ZigBee coordinator, ZigBee routers and ZigBee end-device. ZigBee coordinator acts as the bridge to other networks while controlling the authentication process and security keys. On the other hand, ZigBee routers can perform as a router as well as can run the application functions. ZigBee end-device is the simplest node among these three devices which have limited features which only do communication with routers or with the coordinator. ZigBee is well suited for applications like smart lighting, energy monitoring, home automation and automatic meter reading due to features like low power usage, less complexity, robustness and low cost of deployment. Additionally, ZigBee is ideal for those cases because of easy network implementation, operation within an unlicensed spectrum and especially ZigBee is a standardized protocol based on the IEEE 802.15.4 standard [8]. Also by the U.S. National Institute of Standards and Technology (NIST) has been recognized ZigBee and ZigBee Smart Energy Profile (SEP) as the most suitable communication standard for smart grid residential network domain. Since the communication between smart meters and communication between intelligent home appliances are necessary, ZigBee can be integrated to perform these tasks. Further, ZigBee SEP facilitates functions like sending messages to the homeowners and get information on their real-time consumption. Further ZigBee has 16 channels in the 2.4 GHz band, and each channel is comprised of 5 MHZ of bandwidth.

**Discussion** However, ZigBee networks are prone to interferences since it operates in unlicensed ISM band. Low processing capabilities, small memory size, small delay requirements are other limiting factors of ZigBee [8]. Due to the interference of 802.11/b/g near ZigBee, it is more concerned about the robustness of the ZigBee under high noise conditions. Therefore, energy-efficient routing protocols and interference reduction schemes need to be introduced to get a reliable and energy efficient network performance using ZigBee. However, ZigBee cannot be chosen as an option for this project due to its low range capability.

### LoRa (LoNG RaNGE)

The LoRa communication technology is one of the promising wide-area IoT technologies which is introduced by Semtech and LoRa alliance promoted it further. LoRa has its key features as described below.

**Long range for communication** LoRa can communicate over 22km of distance with line-of-sight (LOS), and even without the LOS, it can communicate with more than 2km. Usage of limited data rate and low sensitivity of receiver can be highlighted as the main reasons for achieving long-range communication from LoRa since distance and data rate is inversely proportional in wireless technology. In other words, when the range over which communication must be done increases, the data rate must be lowered. Since LoRa communicates with a low data rate, it has achieved the ability to communicate over an extensive range. Not only that but also, since it sends the signals in low-frequency data, the receiver even should have low sensitivity. Therefore, in LoRa, that requirement has also been adequately fulfilled which sets the background to communicate over vast distance successfully. Due to the broad rangeability of LoRa, it directly reduces the size of backbone network which includes repeater stations and gateways. Because with other technology several repeaters need to be used for a distance in which one single LoRa can communicate. Therefore, with the use of LoRa cost can get lowered too.

**Low path loss** LoRa uses the 868MHz, and 915MHz ISM bands for its communication unlike most of the other technologies uses a 2.4GHz band for the communication which is one of the specific features in LoRa compared to other technologies. Added to that point, above the mentioned band in which LoRa communicates can be used in any country which creates it suitable for the utilization of any nation. However, more than that with use of the low-frequency band in LoRa, it has achieved lower path loss than conventional wireless technologies.

**Good sensitivity** In wireless communication, obtaining a higher link budget is very useful for successful communication. In LoRa, due to the lower sensitivity of the receiver as about -134dBm and with an output power of 14dBm, it can achieve around 148 dB link budget which is significant value in link budget point of view. And that implies LoRa is capable of handling a successful communication.

**Fewer Interference** As explained earlier LoRa uses 868MHz and 915 MHz ISM bands which are not as famous as 2.4 GHz band in the communication. Therefore, it has lower inferences than the popular bands such as 2.4 GHz band. Lower the interferences mean higher the signal quality and reliability.

**Good obstacle penetration** Since LoRa uses lower frequency than other communication methods, it has more excellent ability to penetrate, and that makes LoRa is more suitable to use in the urban environment. Added to that point, LoRa can penetrate even through brick walls, trees, and concrete. Therefore, LoRa finds fewer losses in transmission in urban environments than communication methods which work in higher frequency bands due to this penetration ability. Finally, due to its adaptive data rate in chirp modulation technology, LoRa is capable of communication over long range with low power consumption and with low-cost for designing can be considered as the heart of its success. There are not many applications about LoRa. But within 2015 and 2016 large number of research is being conducted on LoRa networks.

Jaeyoung So. Et al. has discussed the execution of LoRa network server on OpenStack. Using the services provided by the OpenStack, they have been updated the operations of loRa network server to achieve flexible and scalable service. They have been developed an experimental set up using commercially available hardware related to LoRa like LoRa terminal, LoRa gateway and OpenStack open source software to validate their LoRa networks server on the OpenStack platform [14].

To satisfy the demand for low-power, long-range, high-data transmission need which arises with the commercialization of IoT, Dong Hyun Kim ET. Al. has introduced a method which combines loRa and Wi-fi service together. Though loRa can provide low-power, long-range communication, its data rate is small. Therefore they have chosen wi-fi service to satisfy the high-data-rate problem. Using these two technologies, they have introduced a multi-interface communication module which can achieve long-range and low-power requirements using loRa and a significant amount of data using wireless LAN (wi-fi)[15].

Since recent studies show that safety applications which are supported by vehicle-to-vehicle(V2V) communication have potential to limit the road crashes by 80%, in this paper, they have proposed a method related to loRa. However, packet loss due to communication channel characteristics is a problem related to V2V communication to achieve optimum transmission rate to meet stringent delay and reliability requirements. Therefore, Yuan Yao et. Al. have introduced a loss differentiation, rate adaption scheme using loRa for this V2V application in high way environment. This system can estimate the average packet loss and can differentiate the fading loss from interference loss. Further results show that loRa has the potential to give better effectiveness from features like reliability as well as performance [16].

To study the coverage of LoRaWAN technology, Juha Petajajarvi ET. Al. has experimented using commercially available equipment, mainly for two cases in the city of Oulu, Finland. By connecting a node on the ground and water, they have collected the information regarding the range of communication and using those they have estimated the communication range of LoRa for either case as 15km on ground and 30km on water. Further, channel attenuation model has also been derived using the collected data from the experiment [17].

**Discussion** LoRa has some inherent features that make LoRa to be selected as a communication method for an inverter main reason is the high range (2- 20 km). When using as a communication link, some security measures need to be taken into consideration.

## 1.4 Features of Communication System

The primary objective of this part of the review is to identify the required features that the adapted communication technology should consist.

### 1.4.1 Security

Security can be identified as the most critical factor of a communication link that used in power system applications.

As per Q. Yang et al. power utilities require a secure storage and transportation of information, especially for purposes of billing and grid control. To enhance the security of the power grids, development of effective security mechanisms and standardization efforts are vital, mainly to avoid cyber-attacks [18].

Though it speaks about security it does not shows how to improve the security of a communication link.

(D. Dzung 2005) In their paper, D. Dzung et al. give a comprehensive overview of security features. A framework with eight security features is suggested. Based on those elements the communication link that would be adopted into the SCADA will be modified to improve the security [19].

**Confidentiality:** implies the prevention of disclosure of information to unauthorized persons or systems. In case of SCADA or an inverter data flowing in the communication is critical when those are used for the dispatching. Therefore, the confidentiality must be protected throughout the communication process.

**Integrity:** implies the prevention of undetected modification of information by unauthorized persons or systems. When it comes to controlling authorized person or system would try to hack into the system and control selected communication link must be able to prevent those kinds of attacks.

**Availability:** ensures that unauthorized persons or systems cannot deny access or use to authorized users. This is another problem in communication links where unauthorized people can jam communications between authorizing individuals and illegal people which also needs to be eliminated.

**Authentication:** determines the identity of a system user and mapping of this identity to a system-internal principal (e.g., valid user account) by which this user is known to the system.

**Authorization:** Authorization refers to the mechanism that distinguishes between legitimate and illegitimate users for all other security objectives. Only authorized people should be able to use the communication link. And unauthorized people should not be able to use the link. The developed relationship needs to be able to detect unauthorized and authorized people.

### 1.4.2 System Reliability, Robustness, and Availability

(K. Moslehi and R. Kumar 2010) Moslehi et al. critically review the reliability impacts of primary smart grid resources such as renewable, demand response, and storage. From the most relevant requirements of the power utilities, system reliability holds a major position. Rising energy consumption and peak demand and the aging of power infrastructure can be considered as the reasons which cause the unreliability in the system. Connecting the latest and secure communication protocols, the communication and information technologies, fast and reliable control devices, embedded intelligent devices throughout the power system will significantly enhance the reliability and the robustness of the power grid. As per K. Moslehi et al. reliability of the communication link is also a critical feature[20].

### 1.4.3 Scalability

As per Gungor et al. explains that a smart grid should have the capability to provide a smooth operation of the power grid. In smart grid concept, many smart meters, smart data collectors, smart sensor nodes and renewable energy resources are combining to the communication network. Therefore, the smart grid should be scalable enough to handle the integration of advanced web services, reliable protocols with advanced functionalities, such as self-configuration, security aspects[21].

### 1.4.4 Quality-of-Service (QoS)

Li et al. a QoS mechanism is proposed for the communication system in smart grid, which incorporates the derivation of QoS requirement and applies QoS routing in the communication network. The communication between the power supplier and electricity customers is a critical issue of the smart grid. Performance degradation like delay or outage may compromise stability. Therefore, a QoS mechanism must be provided to satisfy the communications requirements (for example high-speed routing) and a QoS routing protocol must be applied in the communications network. In the adopted communication technology, QoS would be a significant aspect to be considered[22].

## 1.5 Reliability

Reliability analysis is an important aspect of designing and evaluating a safety-critical and fault-tolerant system[23]. Therefore it has become a critical and an essential to comprehensively and quantitatively assess the reliability of a system to ensure its availability and reliability meet the requirements for safe and economical operation [24]. Because of that many works of literature can be found on reliability assessments which have executed on power system related systems such as SCADA, wide area monitoring systems, phasor measurement systems and even of the electrical component like motors and generators.

The [25] has completed a reliability evaluation for a SCADA system considering the performance requirement. This approach has been conducted based on the model for evaluating data transmission time which allows finding the operation time needed to complete SCADA functions.[25] also described and applied an analysis on the system component connectivity to assess the availability of SCADA controls.

Reliability evaluation method for the communication systems in wide-area protection (CSWAP) has described in [26] and in that study system structure, and characteristics of the CSWAP are hierarchically analyzed from layers of substation communication, local communication, and wide-area communication. The results show that the models have particular physical significance, and the proposed method is helpful for quantitative reliability assessment, influencing factor analysis, and weakness identification.

A detailed reliability evaluation is carried out by [24] for phasor measurement unit since it is the primary essential element in the Wide area monitoring system (WAMS). They conclude a sensitivity analysis and redundancy design for the system following the comprehensive reliability evaluation.

Since a reliable operation of the facilities at a system control center (SCC) is essential in a secure operation of the power system,[27] has performed reliability assessment of the operational functions of an SCC.The selection of SCC function reliability indices and performance targets, the assessment approaches are explained in their paper.

[28]carried out a reliability analysis of WAMPAC system since Wide are monitoring, protection and control plays a critical role in smart grid development. From their study, they illustrate the influence of different components on the system reliability.

Other than the compound systems, reliability evaluations are carried out to identify the possible failure causes for electronic components by [29] and they conclude the major failure causes and corresponding measures to prevent component failure. Also [30] presents a technique to quantitatively analyze the reliability of digital relay since the digital relay is rapidly increasing in power system due to the advantages offered by the technology such as economical, high flexibility and reliability. And their result proves that the most suitable method to enforce relay reliability is to improve its self-checking ability.

As the modern power systems have become more complicated, reliability in power systems has been studied extensively[28]. This development is facing an ever-growing demand for quantified reliability and safety. Because of that, there is a significant number of reliability assessment methods is available for power system deployments.

Most of those methods are based on either analytical or simulation approaches. Reliability block diagram, fault tree, and Markov modeling are the most common and IEC 61511 standard recommended methods for analytical approach while Monte Carlo or simulations with stochastic distribution methods are based on simulation approach [31]

The Markov approach can be applied to the random behavior of systems that vary discretely or continuously concerning time and space, characterized by future states been independent of all past states, except the directly previous one[32].Since distribution systems contain a large number of possible states, many simplifying assumptions must be made to limit the Markov model to a manageable size. Markov modeling has been successfully applied to transmission systems [33] and distribution systems [34]. However, it becomes more difficult to depict an accurate logic relationship between the components when it comes to the complex system. Therefore it is not feasible to use Markov modeling when there is a complex system [24].

Fault tree analysis is a commonly used method to derive and analyze potential failures and their possible influence on system reliability and safety. FTA is a proven and accepted procedure in reliability and safety engineering.[35] provides a mathematical and graphical representation of combinations of events that can lead to system failure. [36] through Boolean logic, fault trees represent the relationship between causes and undesired hazardous events. The root of the fault tree represents the unwanted event while leaves represent the causes or failures [35].

FTA was first conceived in 1961 by H.A. Watson of Bell telephone laboratories in connection with a US Air Force contract to study the Minuteman Missile launch control system [37]. Ever since this has been used for the qualitative and quantitative analysis of the failure modes of critical systems. [36].

Depending on the types of gates used a fault tree is classified as static or dynamic[36].

If a fault tree uses only static gates (AND, OR, and k-out-of-n), it is called a static fault-tree. The binary decision diagram BDD provides an efficient means to solve a static fault-tree. If a fault tree contains sequence-dependent gates such as FDEP (Functional Dependency Gate), PAND (Priority And Gate), CSP (Cold Spare Gate), WSP (Warm Spare Gate), HSP (Hot Spare Gate) along with the traditional static gates, it is called a dynamic fault-tree. For the solution of dynamic fault-trees, a Markov model is used [36].

Fault tree construction is a complicated and time-consuming task. Also since these are usually generated manually, it requires highly skilled and experienced engineers to analyze the system based on existing documents that describe the system [35].

However [35] has described an approach to automate reliability analysis by automatic generation of fault trees due to its features like complexity, manual error-prone, costly, and usually incompleteness [35].

Nevertheless, it is infeasible to model a large number of real system components and their dependent reliability characteristics by analytical approaches [28]. As an example following constraints can be explained.

When the PMU analyzed from Markov modeling only a single fault pattern is considered while multiple fault patterns were ignored. Therefore the accuracy of the reliability indices built by Markov modeling is compromised due to inaccurate reliability modeling[24].

Also, [38] claims that Markov modeling and DFTA are constrained to use only the exponential distribution, and as a result of the electric gates the size of the state matrix can explode exponentially by the increasing number of primary events.

To overcome the limitations in the analytical methods, simulation approach like Monte Carlo system can be identified. Monte Carlo Simulation was first introduced in adequacy assessment of power system with HVDC links in [39] . A modified IEEE-RTS [39] load model was used in the example, and the results showed that replacing a particular double circuit AC transmission line by a DC transmission line does not have a significant effect on the system reliability.

### 1.5.1 Discussion and Conclusion

#### Wired Technologies

Considering all aspects costs, security, reliability and availability PLC can be selected as the most suitable available wired technology to be used for monitoring and controlling of an inverter. But disadvantages discussed under the PLC category above need to be eliminated before continuing to use. Presenting a transformer will not allow using PLC for the project because in larger scale energy storage will contain a transformer. Since there is no the method to eliminate or bypass this a new method must be developed to implement the PLC above communication link. To improve the security of the link encryption is need to be added to the link.

#### Wireless Technologies

The objective of this research is to develop find out the most suitable communication method for monitoring and controlling of a grid-connected inverter. Definition the word most suitability comes as a combination of cost, security reliability, and availability. For the project, the range is expected to be higher than 3km. From described wireless technologies only LoRa, radio teletype and UHF/ VHF wireless data transmission systems and satellite, mobile network providers have that much of communication range. Both satellite communications and cellular network providers costs will be higher. And with mobile networks are availability issues associated. Therefore, only first three options are available. In the project, those available technologies will be tested for the range. Since LoRa is a novel LPWAN technology, there is not much literature about LoRa. Table 1.1 shows how field and the data rates are changed in different communication technologies.

Table 1.1: **Comparison of Available Wireless Technologies**

| **Technology** | **Service** | **Data Rate** | **Coverage** | **Spectrum** | **Cost** |
|---|---|---|---|---|---|
| Cellular (2G and 4G) | Data | Low-Moderate | Large | 2GHz | Operating cost |
| LTE(4G) | Data | High | Large | 1.7GHz-2.6GHz | Operating cost |
| Wi-fi | Data | High | Small | 2.4GHz | Operating cost |
| Bluetooth | Data | High | Very Small | 2.4GHz | Low |
| RF | Audio, Data | Low | Small | 300-400MHz | Low |
| LoRa | Data | Very Low | Large range | ISM band | Low |
| UHF/ VHF | Data | Very Low | Large range | UHF band | Low |
| Radio Teletype | Voice | Very Low | Large range | UHF band | Low |

## 1.6 Problem Statement

Communication can be treated as the most critical factor in a SCADA system. Therefore selecting the most suitable communication method will improve the performance of the overall SCADA system. Based on the literature review it can be identified that there is a gap in the field of smart grid for a secure and reliable communication method and the Figure 1.1 illustrates the problem.

Through the developed taxonomy after the literature survey, Power Line Communication has been identified as the most suitable wired technology, and LoRa VHF/ UHF data transmission and radio teletype have been recognized as the most suitable

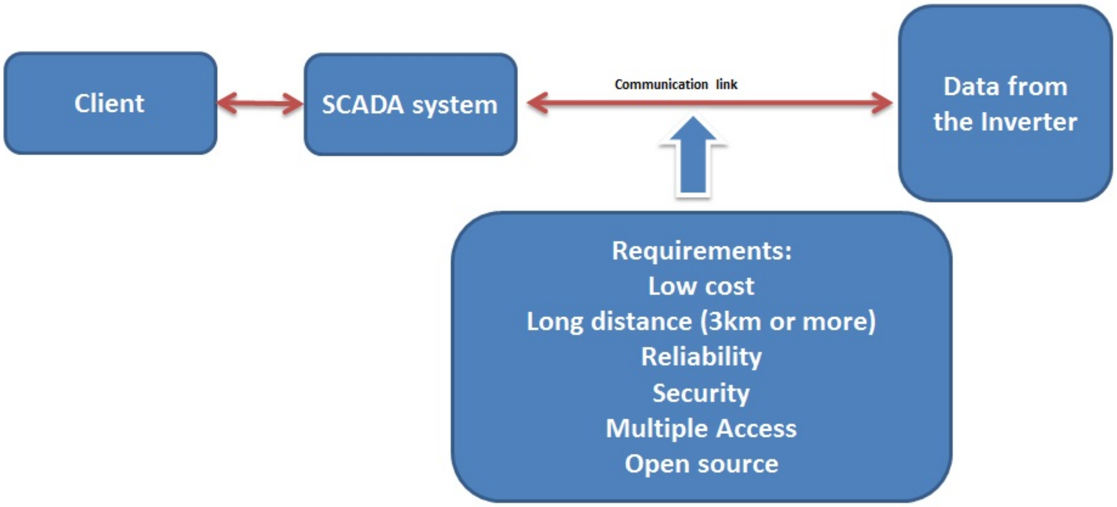


Figure 1.2: Block diagram: Problem Definition II

wireless technologies.

## 1.7 Objectives and thesis Overview

This section gives an overview of the thesis. Figure 1.1 and Figure 1.2 illustrate the main objective of this research.

### Objective 1

Identify a low cost, long range, a secure and reliable communication method for monitoring and control of a grid-connected inverter.

### Objective 2

Develop the selected communication method regarding security and reliability.

### Objective 3

Develop a model to analyze the reliability of the developed communication method.

### 1.7.1 Thesis Overview

To achieve above objectives, this thesis presents a novel method. The thesis is organized as follows,

**Chapter 1:** elaborates developed taxonomy based on the literature survey, and it also gives a brief overview to the project and next chapters.

**Chapter 2:** presents test results and a comparison of few low-cost wireless communication methods.

**Chapter 3:** shows a LoRa based communication method that has been developed for the fulfillment of the main objective.

**Chapter 4:** elaborates a Monte-Carlo based reliability model developed for the measurement of the reliability of the model presented in chapter 3.

**Chapter 5:** concludes the research work by recommending future works.

# Chapter 2

# Test Results and A Comparison of Few Low-Cost Wireless Communication Methods

## 2.1 Introduction

This chapter presents test results, and it compares few low-cost wireless communication methods. Context elaborated in this chapter have been presented in CCECE 2017

As suggested in literature survey, communication between inverters and SCADA or server can be done via power line communication (PLC), optical fiber or wireless communications such as Zigbee, Cellular (3G), LTE (4G), Wi-Fi, Bluetooth and Radio Frequency (RF)[47]. However as per the study each of which method has their pros and cons based on their deployment. Monthly operational cost of both Cellular (3G) and LTE (4G) and especially the low range of 3G technology can be presented as the limiting factors. Even though Zigbee is with relatively low power usage, less cost and less complexity, since it has low processing capability and small delay requirements, it cannot be considered as an excellent selection for practical implementations.The high cost and additional time requirement for building the infrastructure are the primary constraints in wired technologies like optical fiber and PLC, but their ability in providing a high capacity, reliability, and security is significant [47].

The purpose of this study is to find out a low-cost communication method which can be deployed in between inverter of the renewable energy source and server or SCADA. Therefore, in the current research, we focus on three wireless low-cost communication methods, namely LoRa, old Radio teletype technology and UHF/VHF wireless data transmission modules. In the rest of the chapter, the technology and the comparison concerning cost, range and baud rate and the power consumption is presented. The experimental setup which used for testing those methods in suitability for data transmission, an overview of each selected is further explained in this chapter.

## 2.2 Range Testing Setup

Three available technologies are tested to find out the maximum distance that each can communicate successfully. The measurements took in St. Johns, Newfoundland, Canada in 2016 December. One end of the communication system kept stationary at one location, and another end of the communication system was moved. To give fair conditions to all options, the fixed location was same for each, and the same route was followed for all four technologies. Figure2.1 shows path followed for testing. The point which appears as The Attic was the fixed location. It is situated 7 meters above the ground level. Moving end was mounted on a car, and iPhone was used to track the path traveled from the fixed location.

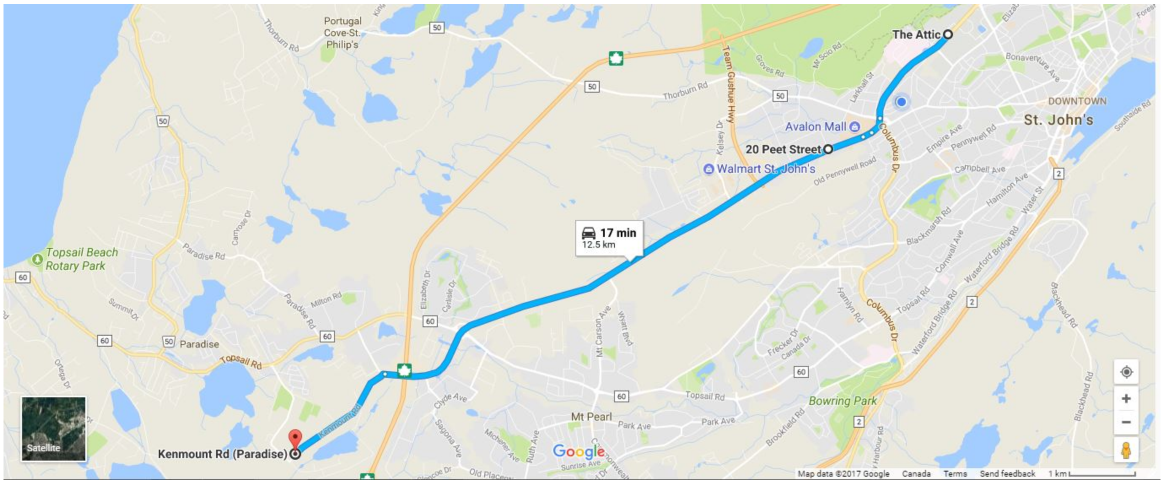


Figure 2.1: Path Traveled

## 2.3 LoRa

Four leading communication technologies are emerging for IoT based applications. Most of the research is being conducted to develop a cellular network for low power wide area network. Sigfox, LTE-M, NB-IoT develop their mobile network for IoT [49]. The main drawback is that all these three comes with a periodical fee for the infrastructure like a mobile service provider. For example, Sigfox comes with an annual subscription fee and technology like Sigfox does not have enough coverage. LoRa has also developed for IoT based applications, but it allows the user to use own backbone network [50]. LoRa is a new spread-spectrum modulation technology which is named as LoRa which uses wideband linear frequency modulated chirp pulses for encoding data. In Direct Sequence Spread Spectrum (DSSS), carrier phase of transmitter changes with a code sequence which is generated by multiplying the data with a spread code. Chirp Spread Spectrum was used in Radar applications for military applications. Due to its many features, such as low power requirement, inherent robustness, resistance to Doppler effect IEEE adopted this for Low-Rate Wireless Personal Area Networks (LR-WPANs) (standard 802.15.4) [51]. LoRa can

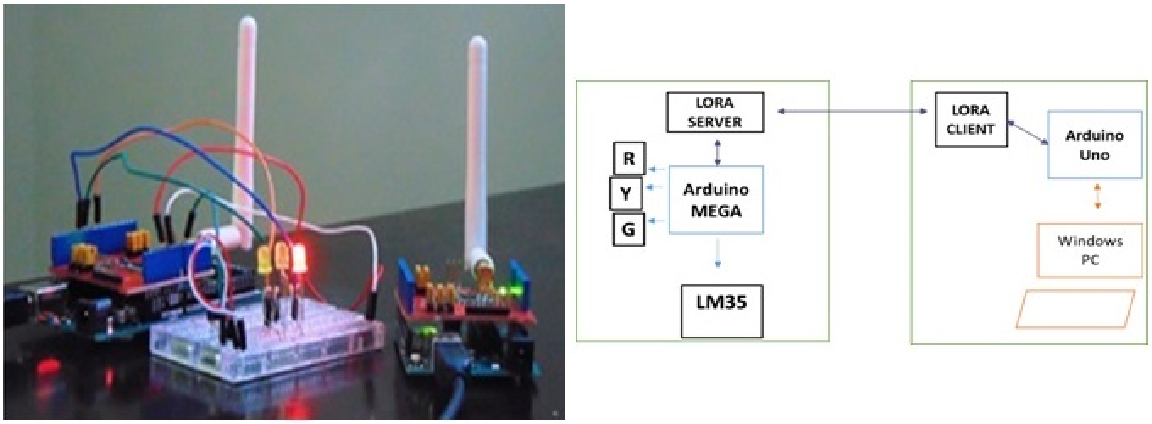


Figure 2.2: Dragino LoRa Shield Based Setup

communicate over a broad range. However, it communicates with lower data rate. Further, key features of the LoRa have explained in the literature survey.

During this project, two sets of LoRa kits from different manufacturers being used for testing namely as Dragino LoRa and Libelium LoRa.

### 2.3.1 Dragino LoRa Shield

Figure 2.2 shows Dragino LoRa Shield that is a low-cost shield. Two units cost only CAD 55.00 but communication range ability wise it only gives up to 220m distance as per our measurements. The experimental setup has been built based on the block diagram shown in the figure 2.2. The LoRa client side kept stationary while the LoRa server side was moving on the path illustrated in figure 2.1. Arduino Mega was used on the server side, while Arduino UNO was utilized on the client side. In this setup, LM35 measured the temperature and sent it per each second to the client through the LoRa link (one bit per second). From the code, the system has been implemented as a full-duplex link server to operate.

### 2.3.2 Libelium LoRa Shield

And the other LoRa kit produced and distributed by Libelium is shown in figure 2.3. Libelium LoRa offers 80 channels with a bandwidth of 500kHz. In the libelium LoRa modules, there are three transmission power levels such as 0dBm, 7dBm, and 14dBM. It costs about CAD250.00. The communication range for this LoRa is 4.5km as per the measurements. And it has more embedded features than the Dragino shield 2.2. It provides 80 channels to the user which allows the user to develop own cellular network. It also allows the user to control the transmitting power and this feature is important to design the cell network and to determine cell size, cell radius, etc. Since the objective of this test to obtain the maximum distance, the nodes were configured to use the longest distance by maximizing the output power to 25mW. To check the range these modules moving end programmed to send a number sequence for each four second at 9600 bps. And the receiving end was scheduled to post the received number to the PC for observations. Figure 2.3 shows a picture of the stationary side while testing. For both ends, Arduino Uno boards were used.

## 2.4 Radio Teletype

Traditional Land Mobile Radio (LMR) networks are being used for voice communications. But over decades same LMR devices used for data communications as well [10]. Before the Internet coming to the arena hobbyist used this Radio Teletype technology for the keyboard to keyboard communications. In this approach, two sets of half duplex ham radios made by BAOFENG are being used for data communication. These are 5W radios that use $400 - 470$ MHz band for communications. Software called FLDIGI is used as the interface. There is some open source RTTY software which can be found on the Internet such as MMTTY. Figure 2.4 shows the block diagram of

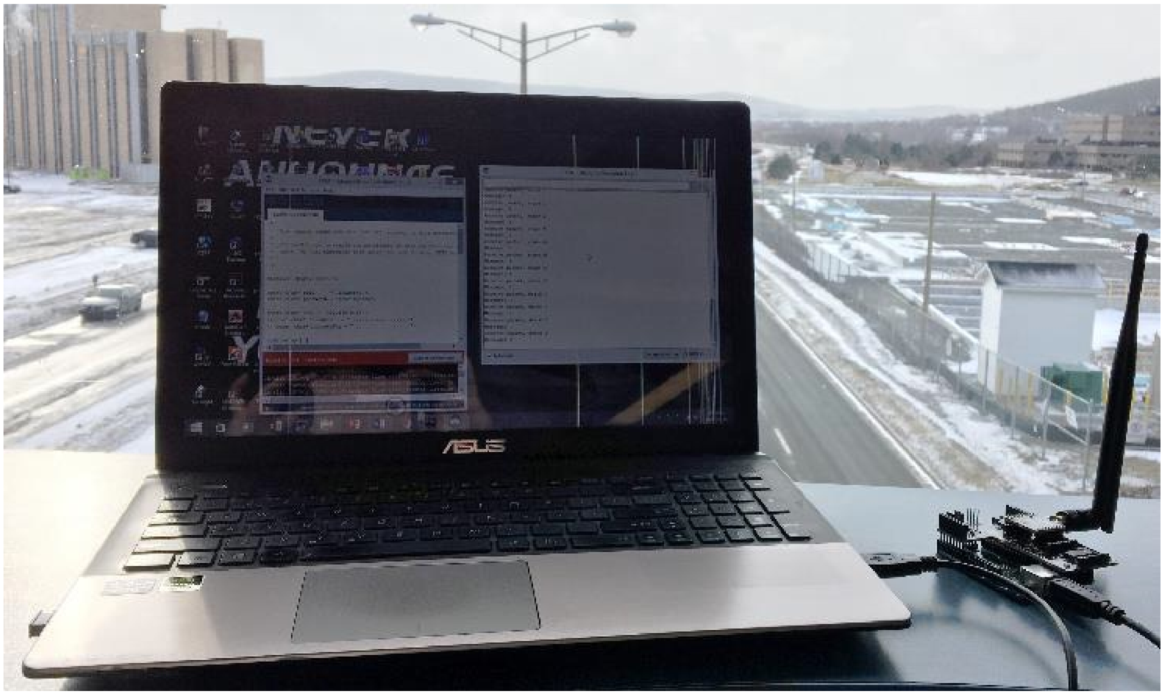


Figure 2.3: Libelium LoRa Shield Based Setup (while testing)

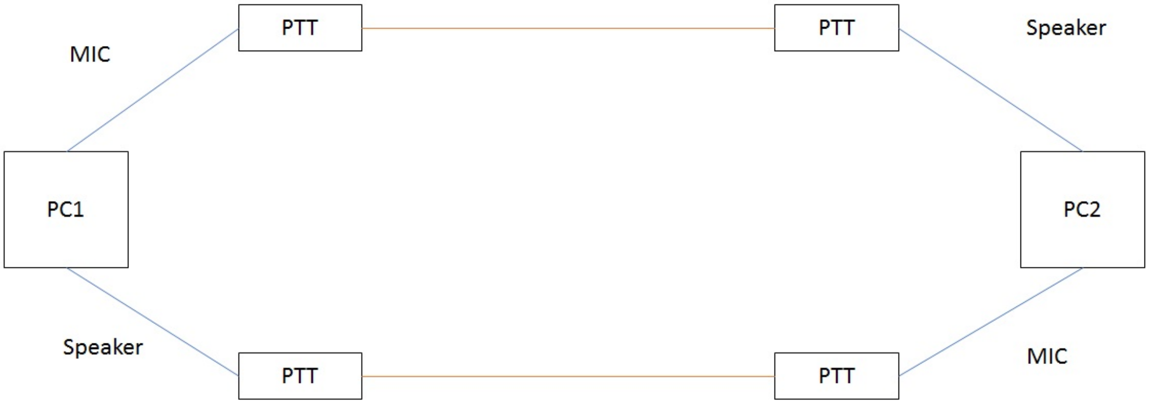
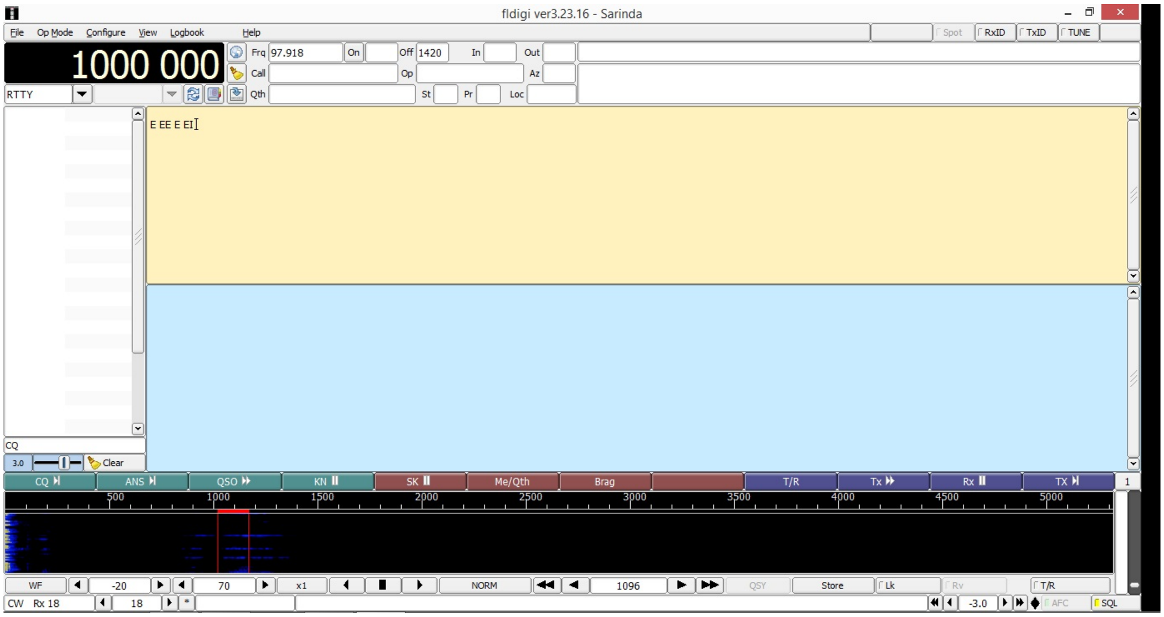


Figure 2.4: Block Diagram for RTTY Based Setup



fldigi ver3.23.16 - Sarinda

1000 000

Frq 97.918

Off 1420

RTTY

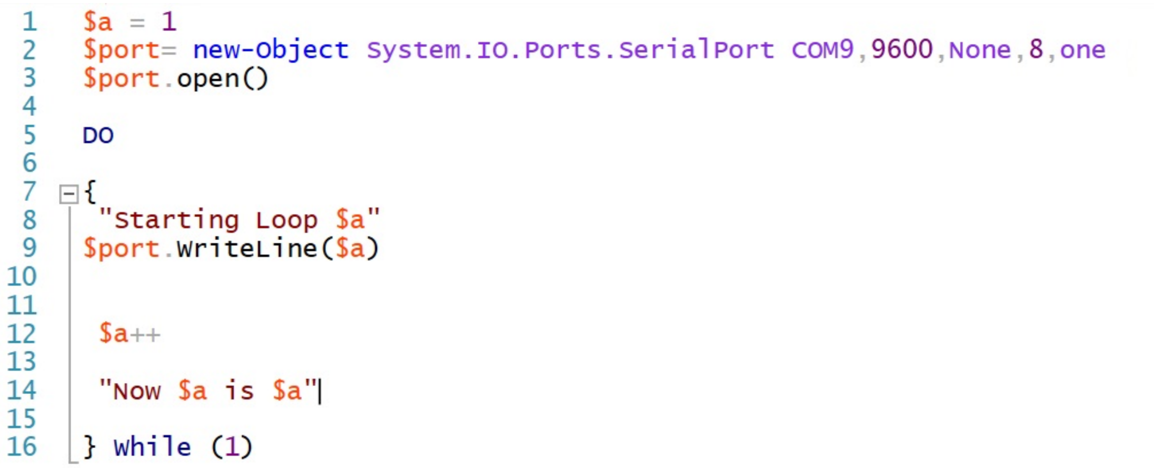
E EE E EI

Figure 2.5: FLDIGI Interface During Tests in Progress.

the system and as shown in figure 2.4 this approach is simple. Two computers with FLDIGI installed have been used as the client and the server side [52]. FLDIGI can be used for RTTY communications. This software is mainly used by Amateur radio operators. Microphone and the speaker of the ham radio are connected to computers headphone and microphone connections. To test the range, one end kept stationary while another end was moving on the path specified in the figure 2.1. And it gave 7km range during the measurements. Figure 2.5 shows the FLDIGI interface during the testing.

## 2.5 VHF/UHF Wireless Data Modem

From the middle of the last century, some companies develop these data modems. For testing and verification in this research, two set of data radios made by Lensen are being used. The main advantage is that both are easy to use where the user can directly send data using a serial port. To test this approach PowerShell script shown



```
$a = 1
$port= new-Object System.IO.Ports.SerialPort COM9,9600,None,8,one
$port.open()

DO

{
 "Starting Loop $a"
$port.WriteLine($a)


 $a++

 "Now $a is $a"

} while (1)
```

Figure 2.6: PowerShell Script

in figure 2.6 has been used. It will generate a number from the non-moving end. And the moving end monitor for that figure. During testing, in a case of moving end unable to receive the desired number that end had been stopped running and the antenna was taken out of the vehicle. Since the objective of this research is to measure the range, lowest baud rate has been used which is 1200bps.

### 2.5.1 1W Wireless Data Module

LS-U1000 is 1W wireless data module smaller in size. The only difference is that the second one is 25W. The communication link costs around 120.00 CAD. This 1W module can be powered with USB power even through the PC. And it has 1200, 2400, 4800, 9600, 19200bps baud rates that can be chosen as per the requirement. The main drawback with this system is that these are the half duplex. To make it full duplex, time division multiplexing must be done through the program. Figure 2.7 shows both ends of the communication link.

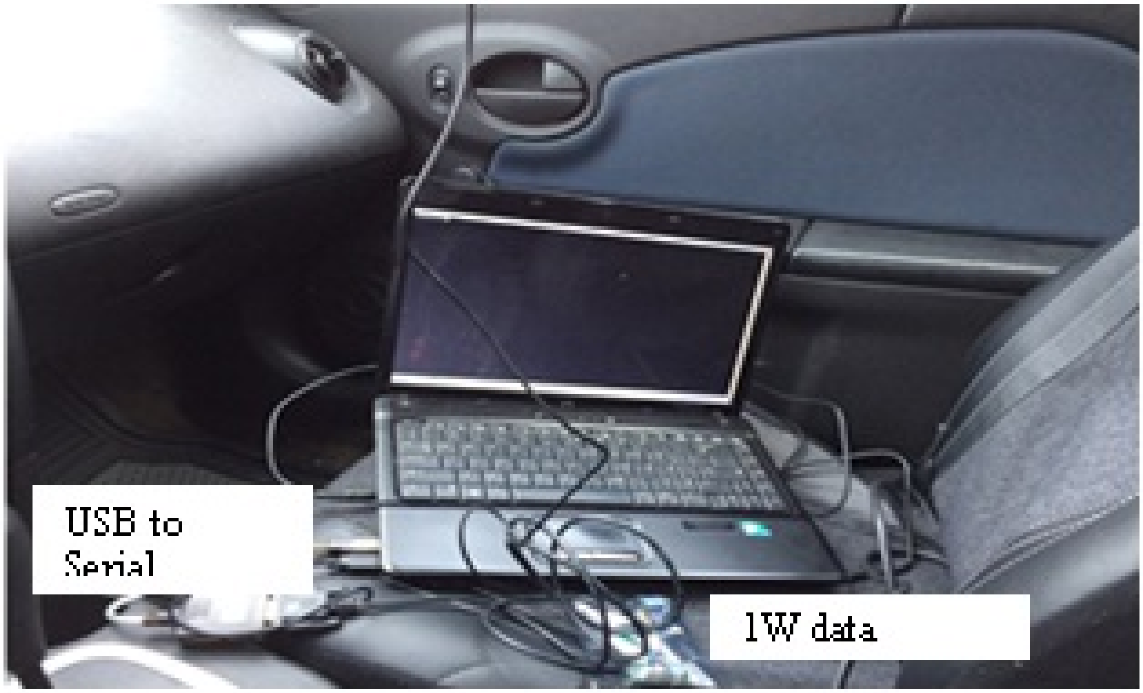


Figure 2.7: Testing 1W Wireless Module

### 2.5.2 25W Wireless Data Module

LS-V25000 is 25W and separate power supply is needed to supply power to the module. And it uses MSK modulation (Minimum-shift keying) and 1200, 2400bps baud rates.

## 2.6 Comparison

All available technologies are compared concerning cost, power consumption, measured range, and data rate and tabulated in the Table 2.1. As per Table 2.1, Dragino LoRa Shield has only 220m measured range while all other technologies have more than 2km range while testing. Due to distance restriction, Dragino LoRa Shield must be omitted. Plus, unlike Libelium LoRa this doesnt divide the spectrum into channels. TDM and TDMA need to use for a population of inverters. Ham Radio has the second highest distance among them all. But data rate wise it has a little baud rate.

Plus, this uses popular band which is being used for many applications which results in high interferences and little security. Due to those issues radio teletype technology, has also been rejected. But this can be combined with another option which is half-duplex like the 1W VHF/UHF data modem. 1W can be considered as a good option because it has an excellent power per km factor which is about 0.303W/km. But this module is half duplex, and it provides only one channel for both transmission and receiving. Therefore, time division multiplexing can be used to make it a full duplex. But when it comes to using these for more than one inverter wherein a population of inverters it will be hard to provide multiple access. With TDMA this can be improved up to limit. Because when the number of users increased it will be difficult to provide real-time data and real-time controlling for the unit. And main drawbacks in TDMA like synchronization problems will make this task more difficult. Libelium LoRa module and 25W wireless modem are the two options left. Depends upon the distance requirement one of these can be selected. Libelium LoRa module provides 13 channels the user can use FDM, and it also allows the user to improve the network with FDMA. If the user needs to improve the number of inverters in the population user will be able to use Hybrid FDMA/ TDMA method. Therefore, with using this technology, a LoRa network can be designed for a large population of inverters distributed across a wide area based on cellular architecture. And the other advantage which Libelium LoRa is having is that it allows the user to control power that will enable the user to control the size of the cell as well.

Table 2.1: Comparison of Tested Open Source Wireless Technologies

| | **Cost (CAD)** | **Power (tx)** | **Range** | **Data Rate (bps)** |
|---|---|---|---|---|
| **Dragino LoRa** | 55.00 | 100mW | 220m | 30000 |
| **Libelium LoRa** | 250.00 | 25mW | 4.5km | 9600 |
| **Ham Radio** | 140.00 | 5W | 7.2km | 50 |
| **1W Data Radio** | 120.00 | 1W | 3.3km | 1200,2400 |

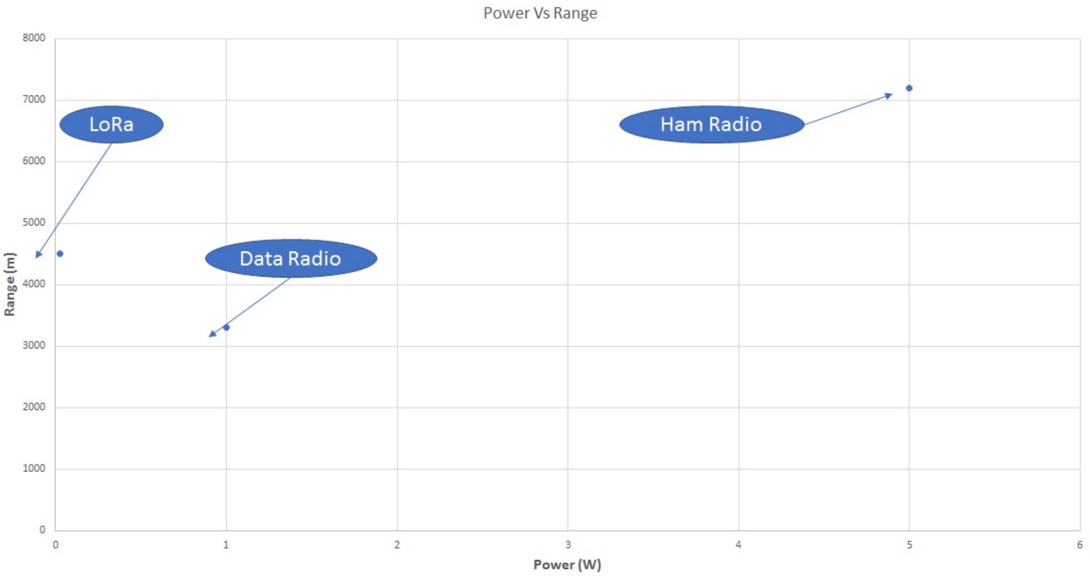


Figure 2.8: Power Vs Range

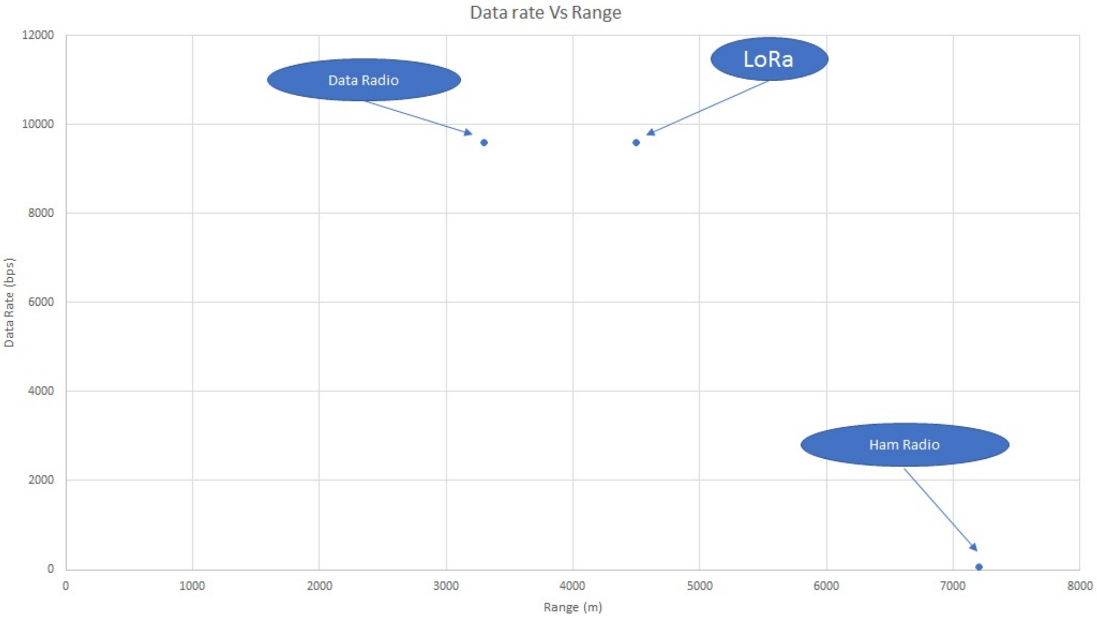


Figure 2.9: Data Rate Vs Range

## 2.7 Conclusion

The main objective of this research is to find out a communication method for a SCADA where equipment is distributed away from the central processor around few kilometers. Data gathered are needed to be sent to the central processor for data processing. Available technologies are documented through a detailed literature survey. Three free band based technologies, LoRa, radio teletype technology, and VHF/UHF data modules are being tested.

Table 1.1 has made based on the range measurement from the previous chapter data and other data from data sheets. From the given information in the Table 2.1 two graphs are used to compare available open source low-cost technologies. Figure 2.8 shows the comparison between power and the range and the Figure 2.9 indicates the rate Vs range graph. From those two graphs and Table 2.1, it can be concluded that the LoRa has more advantages concerning power, speed, and the range compared with other two technologies.

# Chapter 3

# Data Logging and Control of A Remote Inverter Using LoRa with Local Data Storage

## 3.1 Introduction

This chapter demonstrates developments done for a LoRa communication link. Over last decades Internet-connected people all around the world. Internet of Things (IoT) has connected things over the web. Many communication methods are developed in Europe and North America to connect things all around the world. Low Power Wide Area Networks (LPWAN) is designed especially for this purpose. These LPWANs designed for long-range communications at a low rate.

There are many LPWANs such as NB-IoT, LTE-M, Sigfox, LoRa, etc. LoRa is a new ISM band wireless solution introduced by Semtech and LoRa alliance promoted it further. Low power consumption and long range are its significant advantages. LoRa uses a chirp-spread spectrum modulation (CSS) which makes LoRa resistant to multipath fading and Doppler effect. Further, this enhances the receivers sensitivity too. LoRa operates in 433 MHz, 868 MHz and 915 MHz bands depending on the jurisdiction. However, as a disadvantage, LoRa has a low date transmission rate.

LoRaWAN is LPWAN technology incorporates LoRa wireless into network infrastructure. The LoRaWAN can address main requirements like bi-directional communication, mobility and localization services of IoT applications. A typical network of LoRaWAN includes end devices (ED), gateways and a server which gathers and analyses the data collected from the EDs. Since LPWAN typically has the star topology, this LoRaWAN network also has the star of stars topology which means EDs are connected to gateways by LoRa links, and then the gateways are attached to the server via internet protocol (IP) based link. Further, LoRaWAN defines three classes of EDs namely class A, B and C which has differences based on their functionality limitations [44, 45, 46].

As introduced in the last chapter LoRa is a novel technology that developed since 2014. Therefore, there is not much literature.

## 3.2 Problem Statement

The number of literature on the application of LoRa are few, and there are not any developments or publications regarding the usage of LoRa for SCADA systems or monitoring and controlling of an inverter.

**Security issue:** As per the literature, an AES encryption algorithm improves the security of a LoRaWAN network. Researchers have not addressed the safety of a LoRa link used outside a LoRaWAN for private communication. AES slows the process when used in an Arduino. Therefore, as part of the research, this chapter demonstrates the development of a device that can be utilized for monitoring and

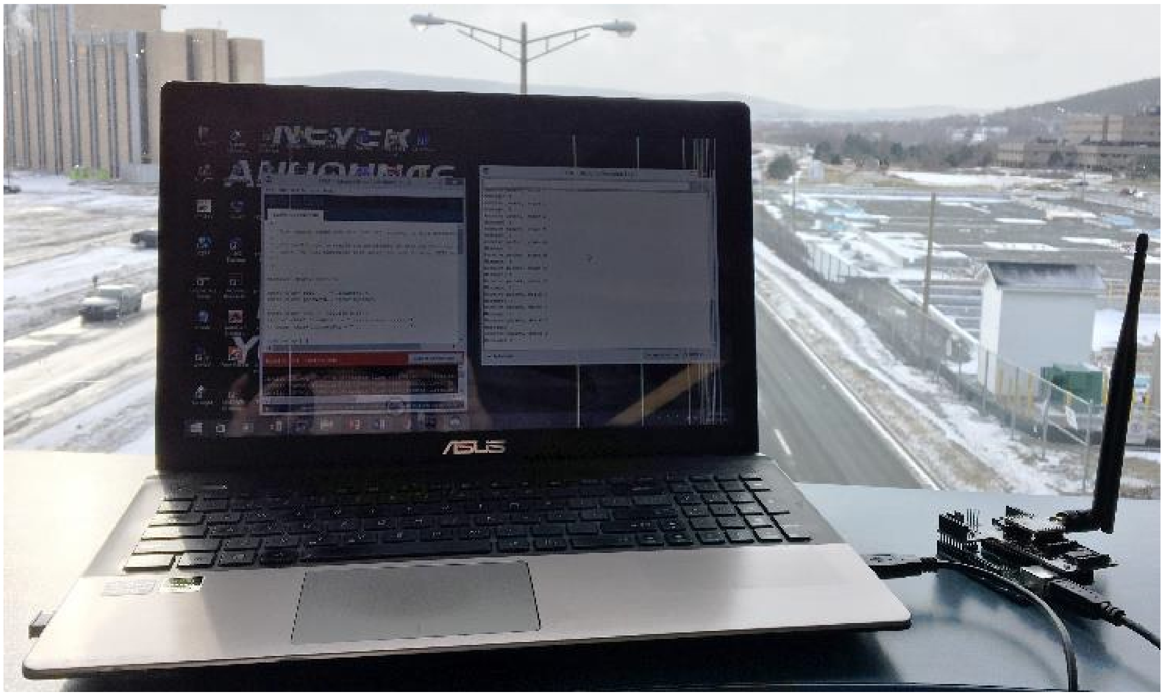


Figure 3.1: Stationary End

controlling of an inverter using LoRa.

**Reliability issue:** The method also facilitates local storage system to improve reliability as well. And another problem that has been addressed is the packet loss in LoRa. Section 3.3 elaborates developments that have done to improve the security and the data redundancy.

## 3.3 Proposed Methodology

The following section explains the methodology used to improve the LoRa communication link. Since the primary objective of the research is to develop a communication link, two sides of the communication link are named as inverter side and the server side. Figure 3.2 illustrates the two sides of the system as described.

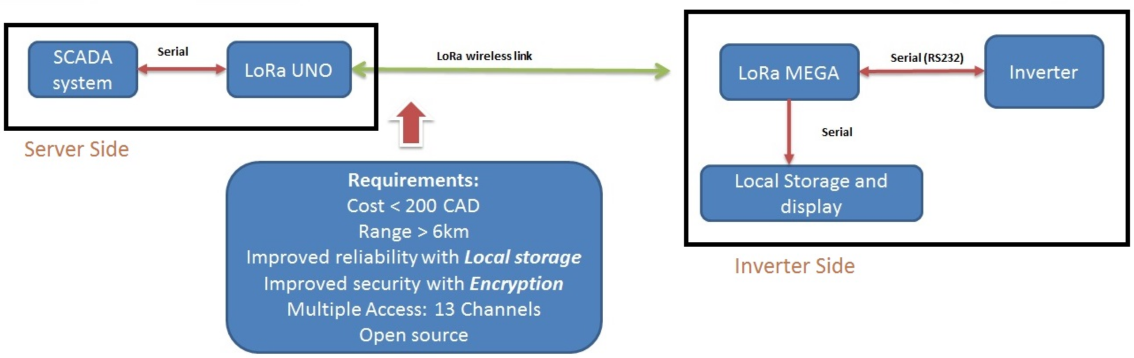


Figure 3.2: Client - Server Model for LoRa

### 3.3.1 Modeling The LoRa Channel

The channel is modeled using free space model and the General Exponential model.

- Frequency: 902 MHz - 928 MHz (ISM band)
- Band width: 500kHz - 125kHz
- Transmission power in three levels:
  - Low: 0 dBm
  - High: 7 dBm
  - Max: 14 dBm
- Receiver sensitivity: -134 dBm
- Link Budget: 148 dBm (Maximum)

Following values are used for parameters in the modeling of the channel.

$\lambda$ Wave length of the signal = 0.32 m

$P_r$ Receiving Power for the signal = -134 dBm

$P_t$ Transmitting Power of the signal = 14 dBm

$G_t$ Transmitting antenna gain = 1

$G_r$ Receiving antenna gain = 1

Free space model can only be used when the distance is lower than the breakpoint distance given by the equation 3.1 [53]. In this case, based on the range testing experiment, Height to the Transmission antenna $[h_t]$ is taken as seven meters, and the Height to the receiving antenna $[h_r]$ is considered as 1m.

Using the equation 3.1 break-point distance is 87.5m.

$$d \geq \frac{4h_t h_r}{\lambda} \tag{3.1}$$

The equation 3.2 illustrates the free space model.

$$P_r = \frac{P_t G_t G_r \lambda^2}{4\pi d^2} \tag{3.2}$$

General Exponential model has been used for distances higher than break-point distance. The equation 3.3 shows the general exponential model.

$$P_r = P_{r(d_0)} (\frac{d_0}{d})^n \tag{3.3}$$

Using equation 3.3 and equation 3.2 the path-loss exponent n can be calculated as 5.0101 for the selected terrain in the winter. And the same testing has been carried out in the summer while keeping all other conditions same and got a maximum distance of 4.10 km and for the summer path-loss exponent has become 5.1313.

There is a requirement to control the transmission power to improve the energy efficiency of the overall system as well as to reduce interferences between users.

Path-loss exponent helps for signal strength prediction and simulation in macro environments. And it facilitates the user to determine the maximum distance at different transmission power levels and let the user control the power. Also, the tested method can be utilized to determine the path-loss exponent for various terrains, and that can be applied to determine the optimum power level.

## 3.3.2 Inverter Side

Inverter side is mainly responsible for following three main tasks and achieve those objectives Arduino MEGA based system has been designed and tested.

- Communication with the inverter through RS232 port.
- Store and display data using SD card and a local LCD.
- Securely and efficiently communicate with the server side.

### Physical Arrangement

Figure 3.3 shows the physical arrangement of the circuit where Arduino MEGA is acting as the master. It uses RS232 interface to communicate with the inverter using UART protocol and uses SPI to communicates with the LoRa communication shield. To save and display received data Master MEGA board sends data to another Arduino MEGA connected to the circuit using UART protocol. The above said slave board then transmit data to the TFT display for displaying purposes and data logging purposes using SPI protocol. Figure 3.5 illustrates the stacked circuits and this method allows the user to save space.

To protect circuits, an enclosure has been designed and rapid prototyped.

### Enclosure for the Server Side

Another particular feature that is developed is the enclosure and designed using SolidWorks, and then 3D printed. The enclosure aims to protect the circuit from the

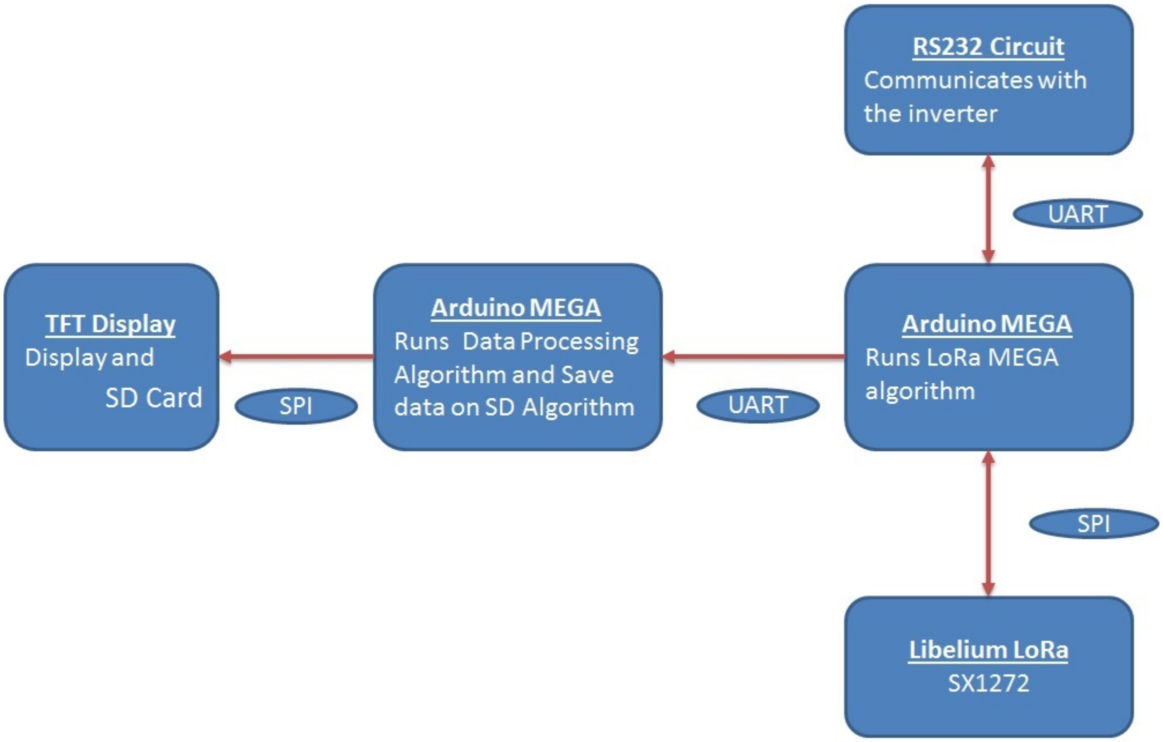


Figure 3.3: Inverter Side Block Diagram

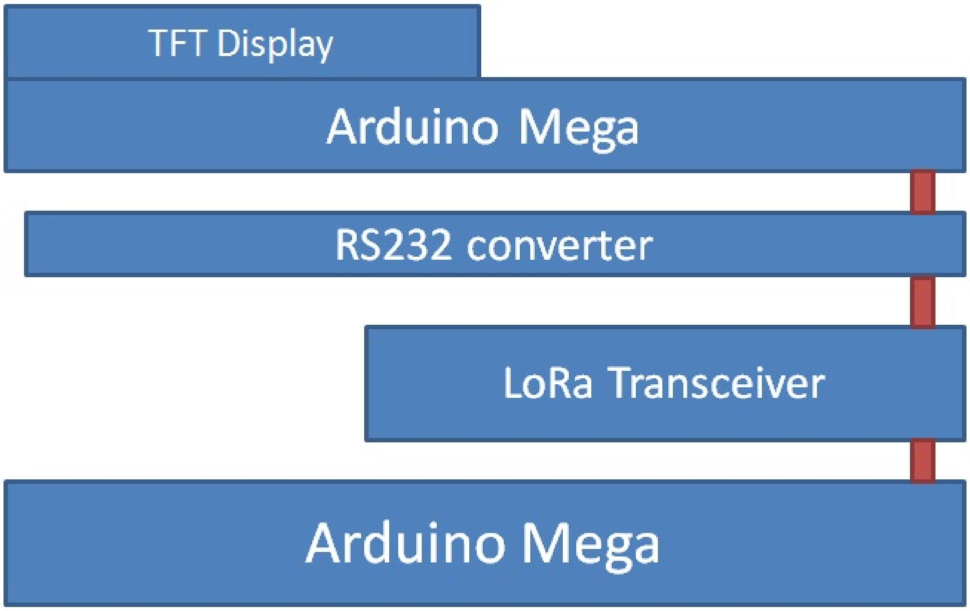


Figure 3.4: Block Diagram of the Inverter Side

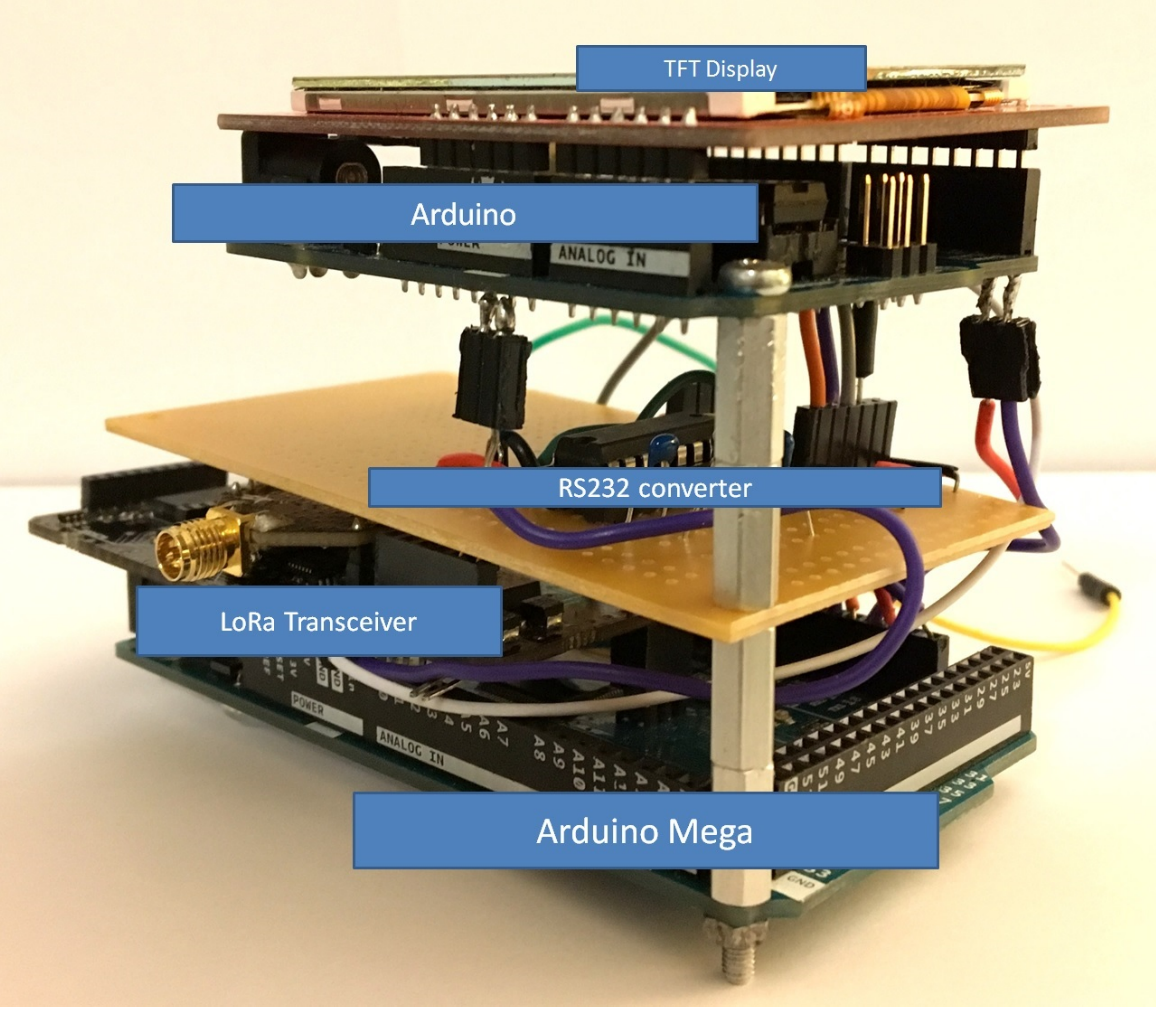


Figure 3.5: Actual Circuit Diagram

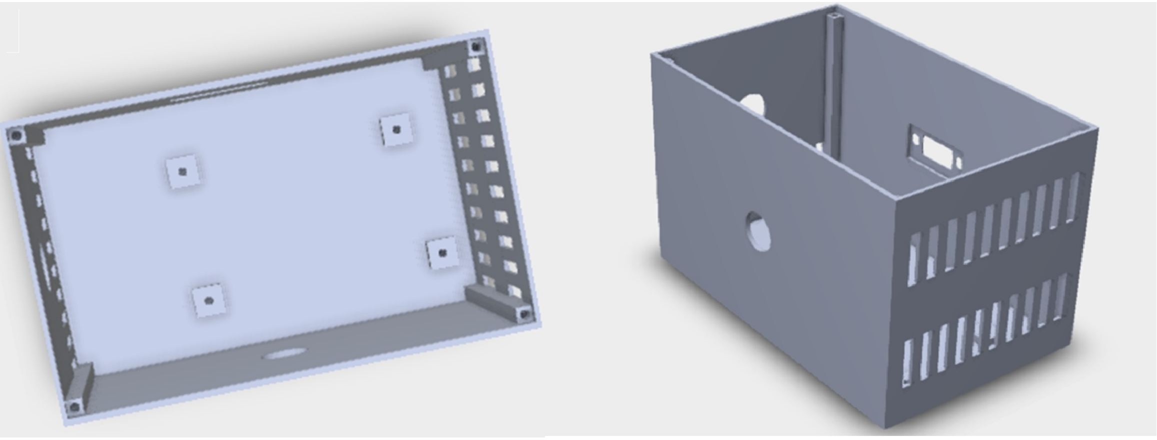


Figure 3.6: Enclosure

outside. It provides IP20 protection for the circuit and allows heat to flow through fins.

### Main Algorithm for the Inverter Side

The main algorithm is developed to perform three primary tasks stated above. As shown in the flowchart in Figure 3.7 master MEGA acquires data from the inverter and sends it to the slave MEGA for displaying and data logging purposes. The data are encrypted for security purposes.Transmitted data using LoRa once the data is encrypted. As the next step, master MEGA will be waiting maximum five secs to receive data from the server side. If it receives data, the decrypted data will be sent to the inverter. Algorithm 1 shows the pseudo code for the developed algorithm. A copy of the code is included in the Appendix A.

**Algorithm 1** LoRa MEGA algorithm

```
procedure SETUP                                ▷ One time run Setup function
    Serial begin                               ▷ Setup com. with computer
    Serial 1 begin                             ▷ Setup com. with display
    Serial 2 begin                             ▷ Setup com. with inverter
    LORAconfig()                               ▷ Configure LoRa module
end procedure
procedure Main Loop
    String key1                                ▷ Define keys for encryption
    String key2
    Call serialreading()
    Send same String on display UNO
    Call encryptt(key2)
    Call LoRaTransmit ()
    Call LoRaReceive ()
    Call decryptt(key1)
    Send decrypted message to Inverter
end procedure
```

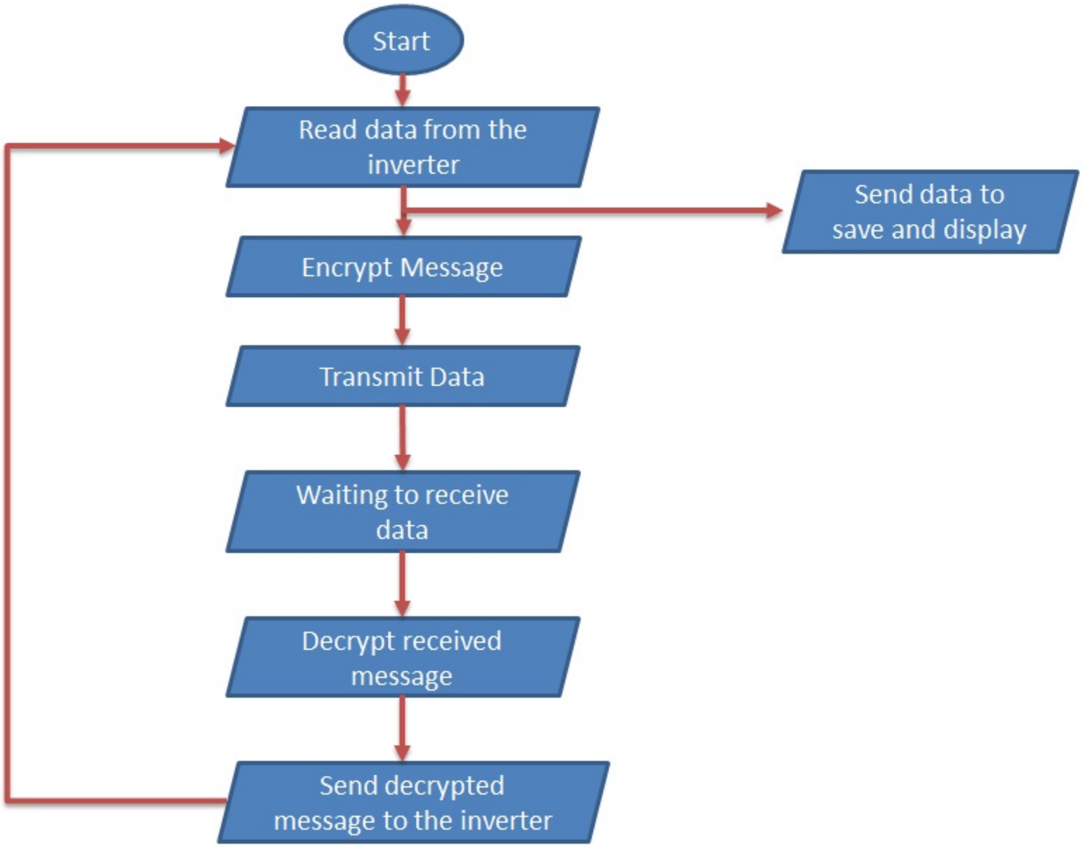

Start
Read data from the inverter
Send data to save and display
Encrypt Message
Transmit Data
Waiting to receive data
Decrypt received message
Send decrypted message to the inverter

Figure 3.7: Flow chart for the Main Algorithm

### Encryption and Decryption Algorithms

Although LoRaWAN technology is developed using encryption methods like AES, it is harder to implement such encryption algorithm with a low-cost Arduino without compensating the speed. The primary objective of this algorithm is to improve the security of the communication link without sacrificing the processing speed by encrypting data using two encryption methods.

First one is using a shift cipher. And the second one is using Vigenere Cipher with a critical word and shifting each letter in the plain text by different number per each letter in the keyword. Since in most of the SCADA applications, the length of the plaintext has a constant length, the keyword is selected such that it would be longer than the plain text. Therefore, each letter is shifted differently.

Figure 3.8 shows the flow chart for the encryption algorithm and algorithm is shown in Algorithm 2. For decryption, the encrypted message is going as per the block diagram shows in Figure 3.9. And algorithm used for decryption is shown in algorithm 3. For this particular application, the length of the message is known therefore the key is selected with the same length. While loop is running to change each character in the message. In each cycle, ANSI value of the character in the message is assigned to X, and the ANSI value of the corresponding character in the key is assigned to Y. As shown in line 6 in the algorithm 2, Z is found. In this case, X and Y are used for Vigenere Cipher, and 21 is used for shift Cipher, 65 is subtracted to get the value of Z to an ANSI value of a simple character. Same steps are done in reverse in the decryption algorithm.

**Algorithm 2** Encryption Algorithm

```
procedure ENCRYPTION($key, message$)
    $messagelength \leftarrow LengthofMessage$
    while $i < messagelength$ do
        $x \leftarrow ANSIvalueofmessage[i]$
        $y \leftarrow ANSIvalueofkey[i]$
        $z \leftarrow x + 21 + y - 65$
        $encryptedmessage[i] \leftarrow Characterofz$
        $i \leftarrow i + +$
    end while
    return
end procedure
```

**Algorithm 3** Decryption Algorithm

```
procedure ENCRYPTION($key, message$)
    $messagelength \leftarrow LengthofMessage$
    while $i < messagelength$ do
        $x \leftarrow ANSIvalueofmessage[i]$
        $y \leftarrow ANSIvalueofkey[i]$
        $z \leftarrow x - 21 - y + 65$
        $decryptedmessage[i] \leftarrow Characterofz$
        $i \leftarrow i + +$
    end while
    return
end procedure
```

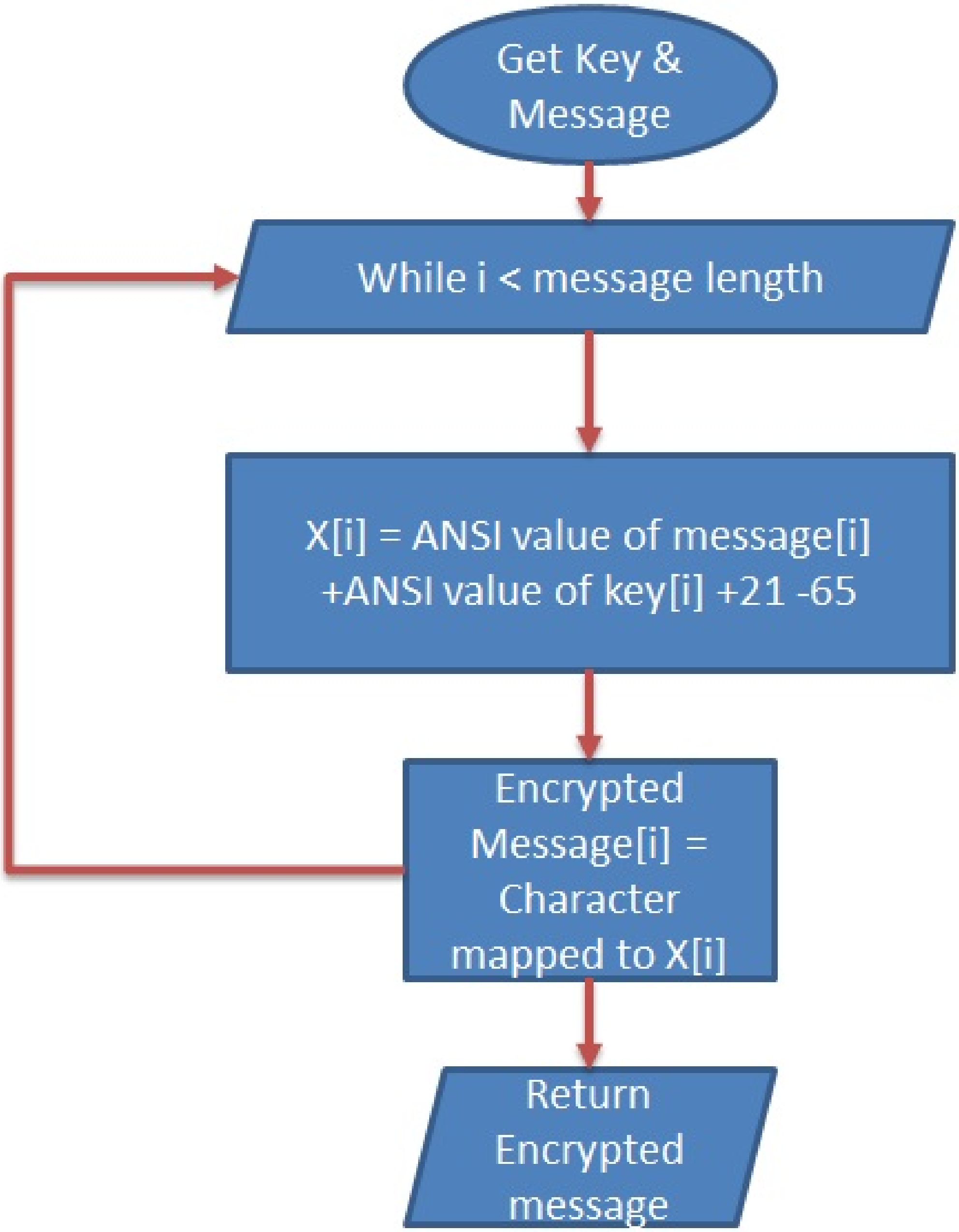


Figure 3.8: Flow chart for the Encryption Algorithm

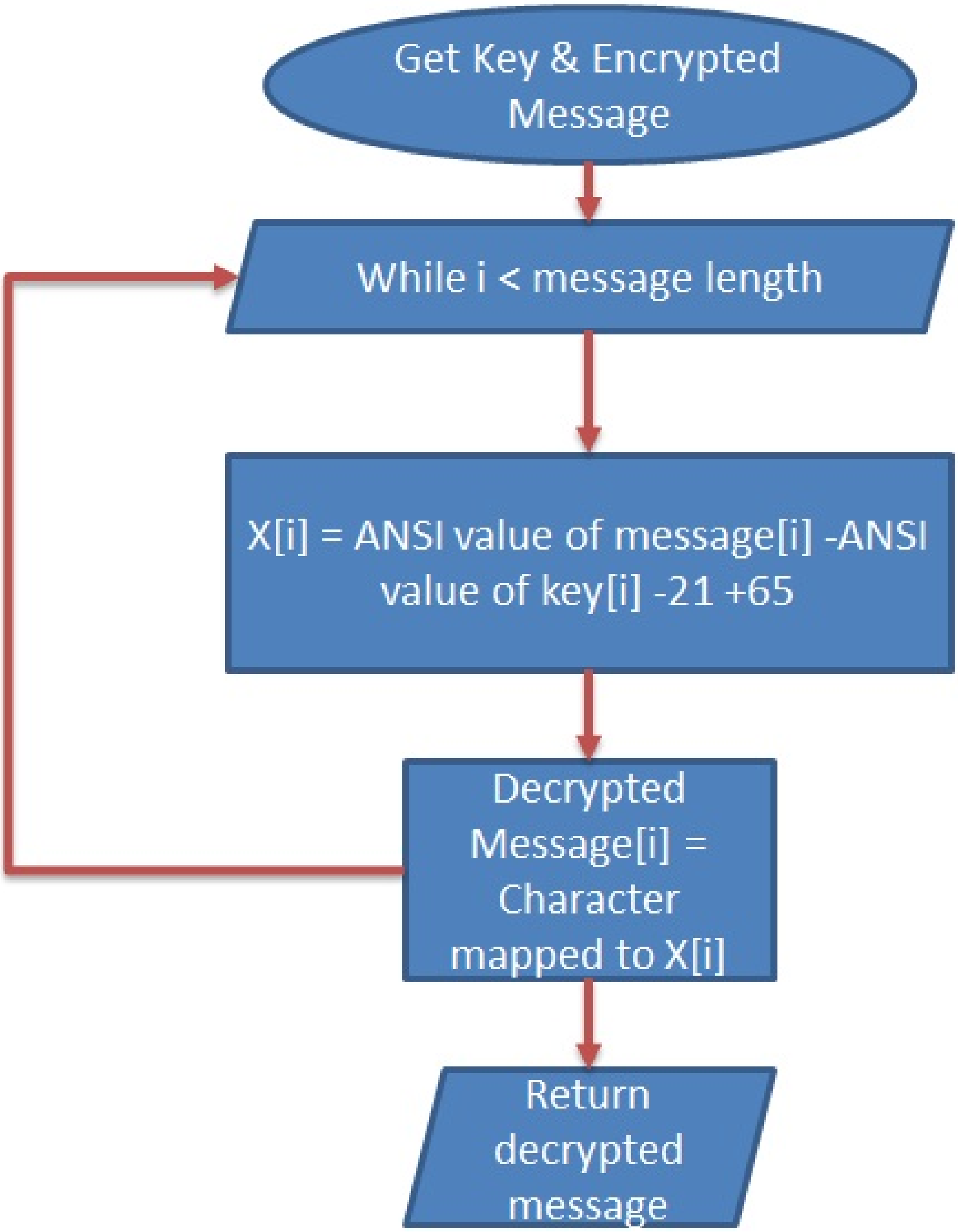

Get Key & Encrypted Message
While i < message length
X[i] = ANSI value of message[i] -ANSI value of key[i] -21 +65
Decrypted Message[i] = Character mapped to X[i]
Return decrypted message

Figure 3.9: Flow chart for the Decryption Algorithm

### Data Processing Algorithm

Data coming from the inverter are sent to the slave MEGA for displaying and data logging purposes. At the beginning of the program, it reads data stored in the EEPROM for time setting goals. And self-set date and time. Also, it allows the user to set date and time where if the user does not adjust the time within 5 seconds system will automatically take the saved date/time on the EEPROM as the present date and time. Then it will set up the SD card. After the setup, the main loop runs which starts with taking serial input from the Master MEGA. Data string for the developed application is coming as a HEX string. And it also comes in Little Endian format. And as the next step of algorithm data string will be split and converted into DEC for displaying and data logging. This algorithm is explained in the figure 3.10 as a flow chart and the pseudocode is given in algorithm 4.

### Save Data on the SD Card Algorithm

Next important feature that the system offers is the data logging on the SD card. Data string split into parts according to the Appendix D and save in the card in each 5 second in CSV format where information contains a time stamp, voltage, current, power, the version number of the inverter respectively. The advantage of this CSV file is that it can be directly uploaded to an IoT server which developed as another part of this research.

The block diagram of this algorithm is shown in the figure 3.13. Transferred data supposed to be saved at a secure location, therefore, there is no requirement to keep data inside the local SD storage for a longer time. Therefore to solve this issue, data will only be stored for seven days. Algorithm 3.13 addresses this issue. Since data saved in every 5 seconds, 17280 data strings saved for a day $(12 * 60 * 24)$. After keeping data for seven days algorithm automatically deletes it.

**Algorithm 4** Data Processing Algorithm

```
procedure SETUP
    Serial begin                                        ▷ Setup com. with LoRaMEGA
    Display setup
    Read IndexID and date/time from EEPROM
    Set date and time
    Setup SD
end procedure
procedure MAIN LOOP
    Read Serial                                         ▷ Read data from inverter
    Split the String
    Allocate data to variables
    Convert hex to dec
    Update time
    IndexID++
    Convert data String to CSV format
    Save data on SD
    Save IndexID and Date/Time on EEPROM
    Call displayValues()
    if $OKbuttonpressed$ then
        Close files
        Display ”Safe to Remove SD and Press OK to start”
        if $OKbuttonpressed$ then
            Reset the program
        end if
    end if
end procedure
```

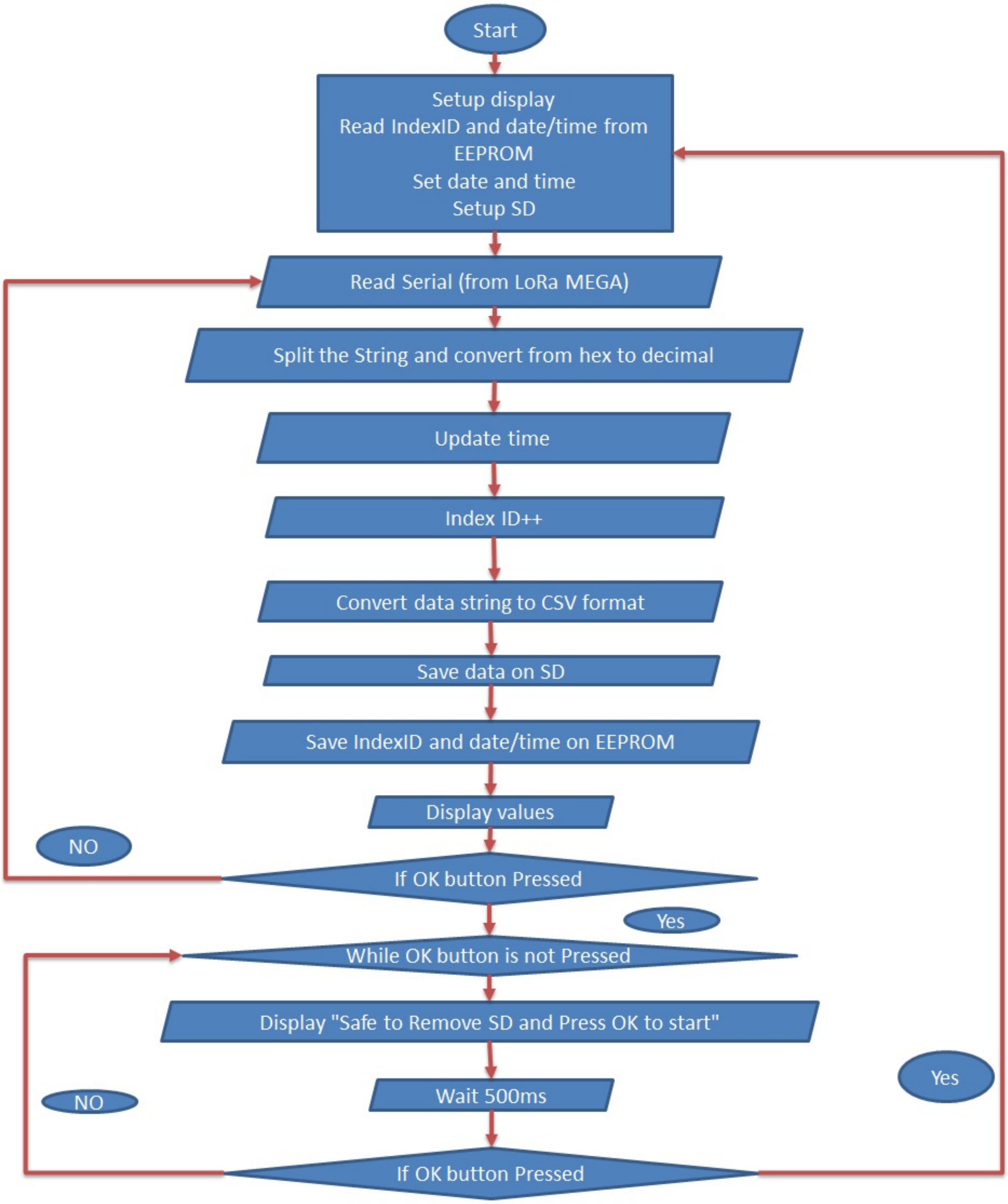

Start
Setup display
Read IndexID and date/time from EEPROM
Set date and time
Setup SD
Read Serial (from LoRa MEGA)
Split the String and convert from hex to decimal
Update time
Index ID++
Convert data string to CSV format
Save data on SD
Save IndexID and date/time on EEPROM
Display values
NO
If OK button Pressed
Yes
While OK button is not Pressed
Display "Safe to Remove SD and Press OK to start"
Yes
NO
Wait 500ms
If OK button Pressed

Figure 3.10: Flow Chart for the Data Processing Algorithm

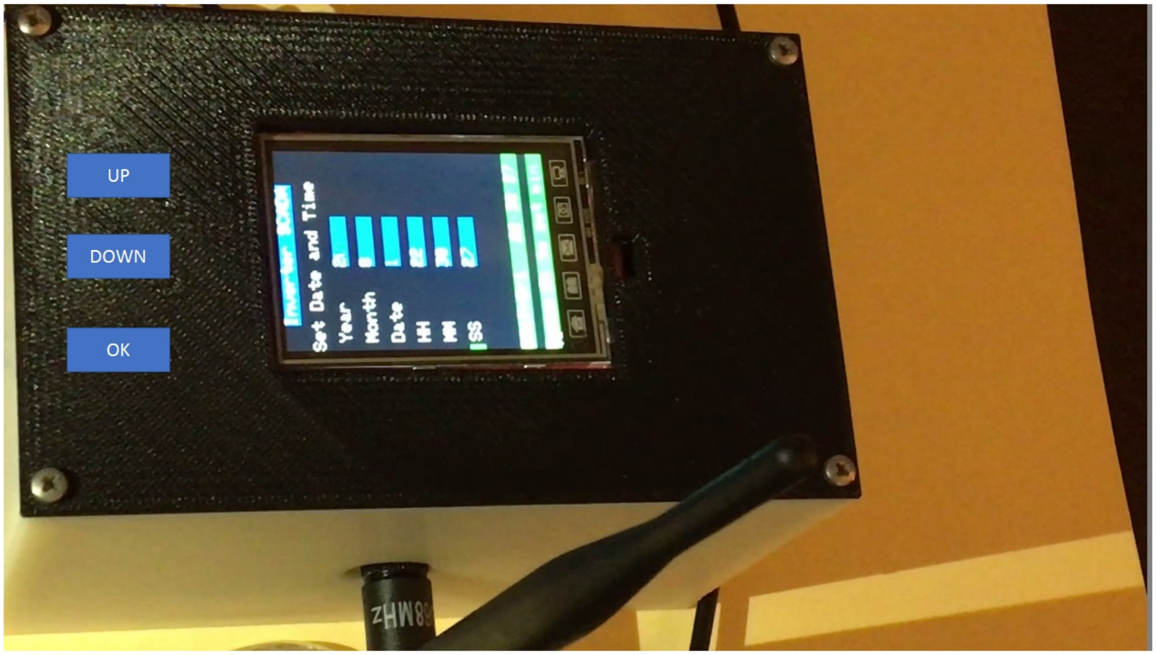


Figure 3.11: Set Date and Time

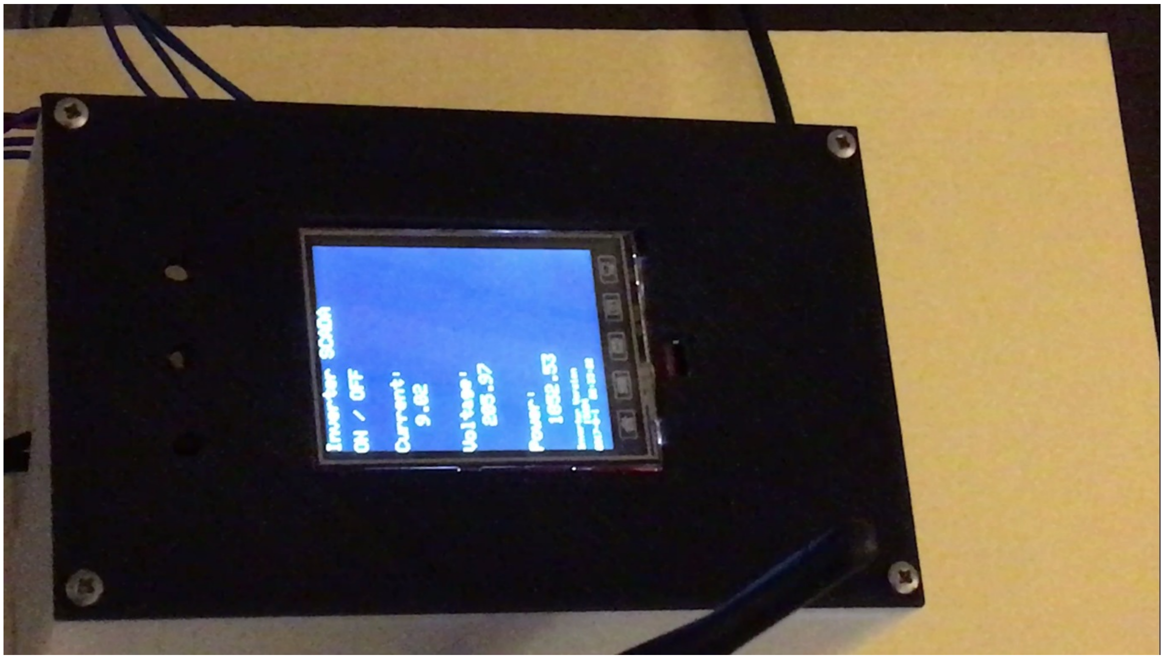


Figure 3.12: Displaying Data on the Local Display

**Algorithm 5** Save Data on Storage Algorithm

```
procedure sdSave($dataString$)
    if $indexid = 0$ then
        if $1.CSV exists$ then
            Delete 1.CSV
            Create new 1.CSV
            Save data string to 1.CSV
        end if
    else if $0 < indexid < 17280$ then
        Save data string to 1.CSV
    else if $indexid = 17280$ then
        if $2.CSV exists$ then
            Delete 2.CSV
            Create new 2.CSV
            Save data string to 2.CSV
        end if
    else if $17280 < indexid < 2 * 17280$ then
        Continue this for upto 7.CSV
    else
        IndexId =0
    end if
    return
end procedure
```

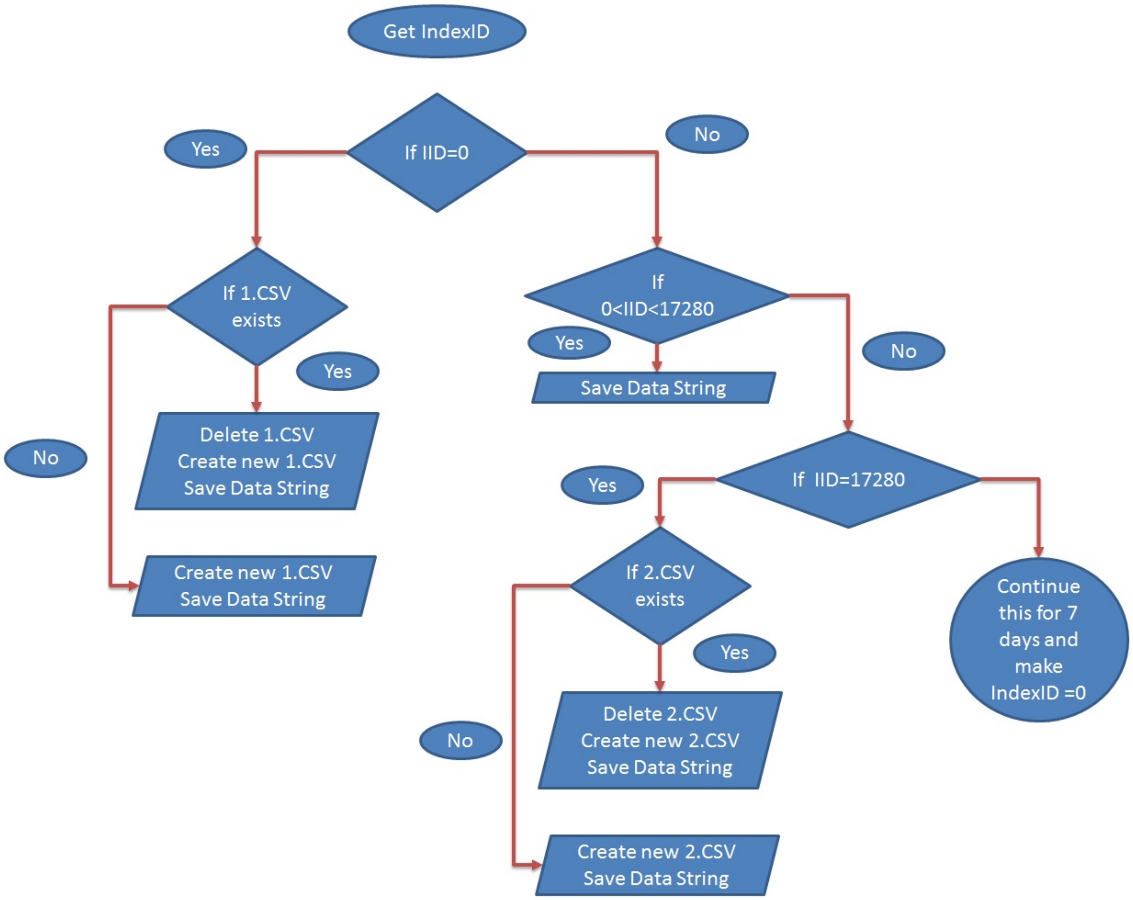


Figure 3.13: Flow chart for the data saving Algorithm

### 3.3.3 Server Side

On the server side, LoRa SX1272 module is connected to an Arduino UNO. And that component will communicate with the server computer through a USB. Block diagram for algorithm is shown in 3.14. And algorithm 6 explains algorithm used on the server side Arduino UNO. Copy of the full code is in Appendix C.

**Algorithm 6** LoRa UNO Algorithm

```
procedure SETUP
    Serial begin                                  ▷ Setup com. with computer
end procedure
procedure Main Loop
    String key1                                   ▷ Define keys for encryption
    String key2
    Call LoRaReceive ()
    Call decryptt(key2)
    Send decrypted message to Computer
    Call serialreading()
    Send same String on display UNO
    Call encryptt(key1)
    Call LoRaTransmit ()
end procedure
```

### 3.3.4 LoRa and Power Line Carrier communication Based Redundant Model

Wireless networks are inherently vulnerable since it can be easily interfered by a third party. LoRa uses free ISM band, and anyone can use the same band which allows anyone to add noises to the signal by adding higher interferences. Methods like slow frequency hopping will be helpful to overcome this issue. But in a case where the entire band interferes then frequency hopping will not be useful. To overcome temporary data losses, an SD card based local storage is added to the system. But this does not solve the communication loss between the server and the inverter which is more

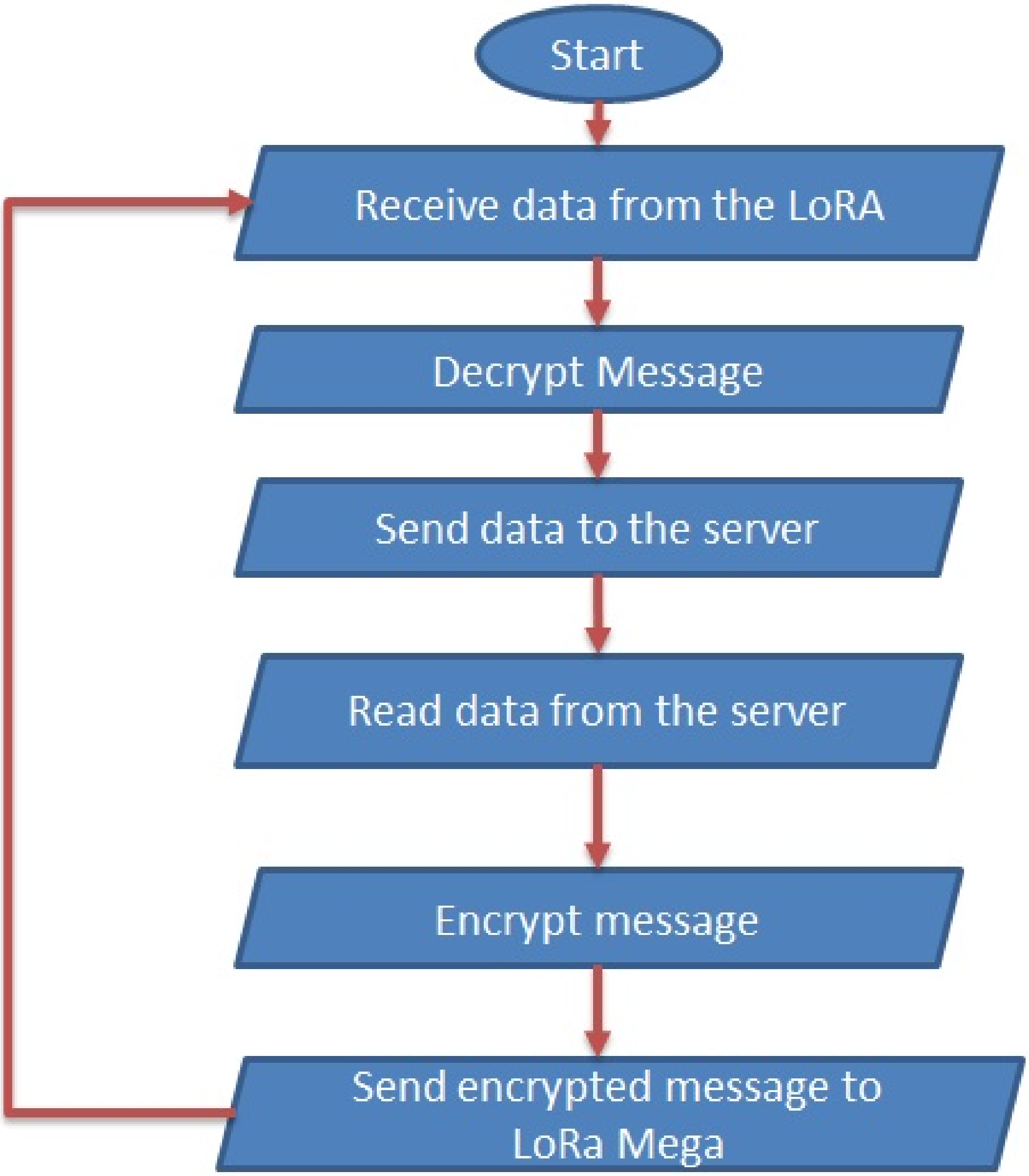

Start
Receive data from the LoRA
Decrypt Message
Send data to the server
Read data from the server
Encrypt message
Send encrypted message to LoRa Mega

Figure 3.14: Flow Chart for the Server Side

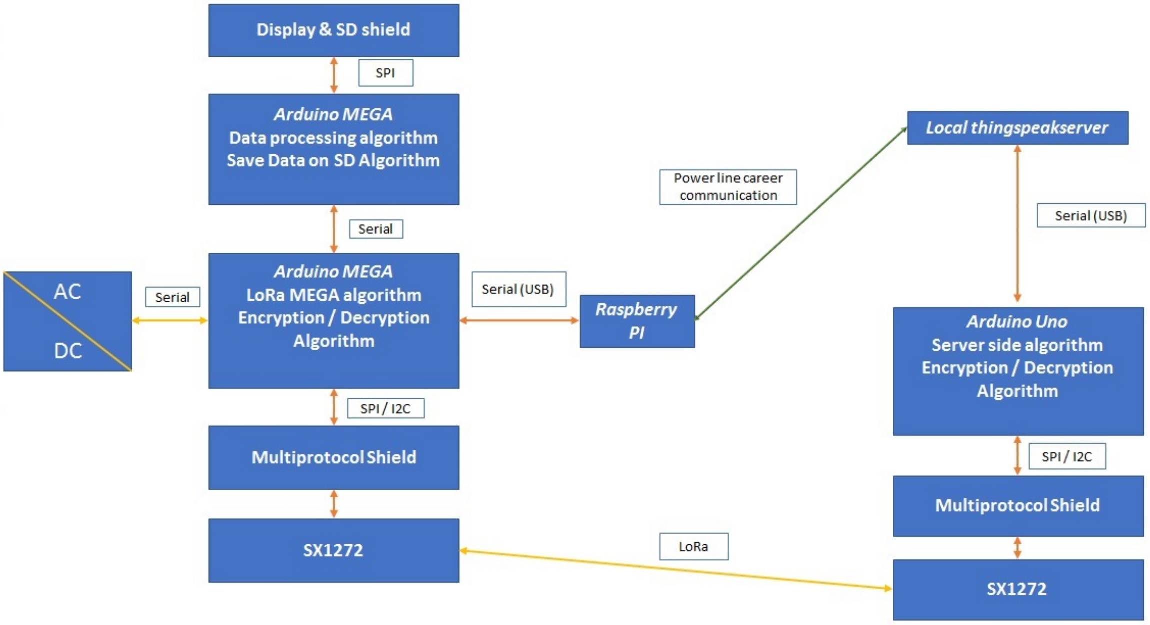


Figure 3.15: Block Diagram for LoRa and PLC setup

critical than the data loss.

Therefore, this section proposes a solution to improve the reliability of a LoRa communication link by using a **power line carrier based plus LoRa** solution where it is possible to use. Powerline carrier communication is a wired communication technology which uses power lines for communications. To achieve this objective, inverter side of the developed LoRa link is connected to a Raspberry PI using USB port of the Arduino Mega through the Serial interface. Same data string received by the Arduino Mega through Serial2 interface is written on the Serial0 interface. As shown in the Figure 3.16 Raspberry PI is connected to one end of the PLC module. As Figure 3.15 emphasizes data flows in two communication channels, where one set of data flows through the power lines and the same data set flows through the LoRa as wireless. This redundant system will improve the reliability of the communication model.

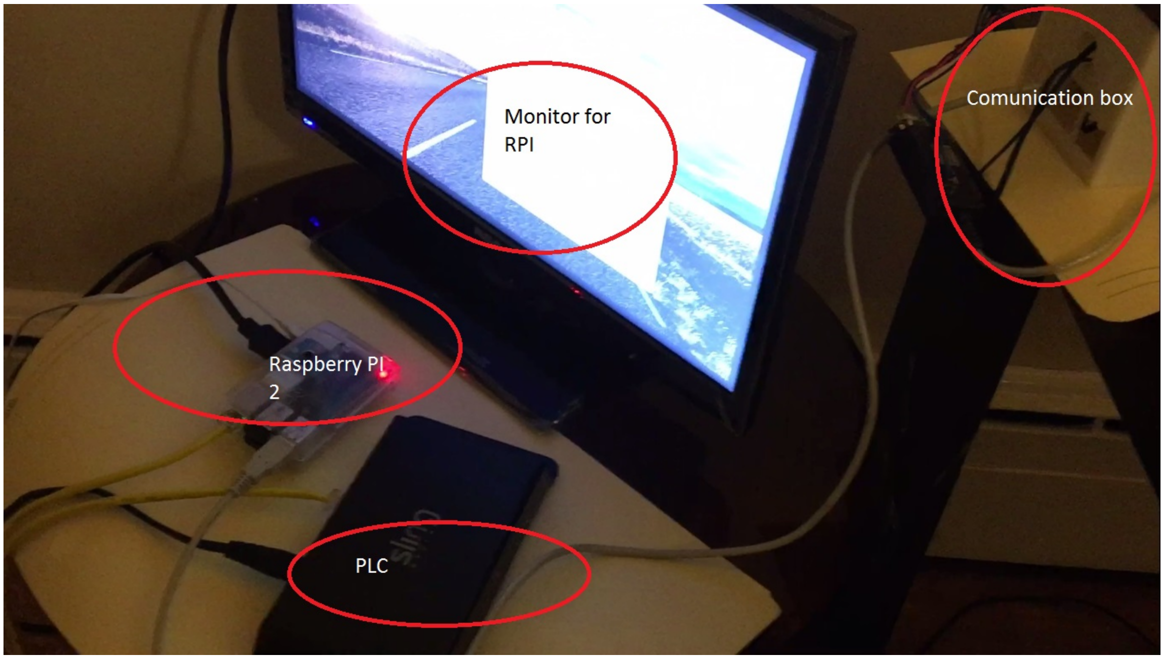


Figure 3.16: Inverter side of the PCL + LoRa link

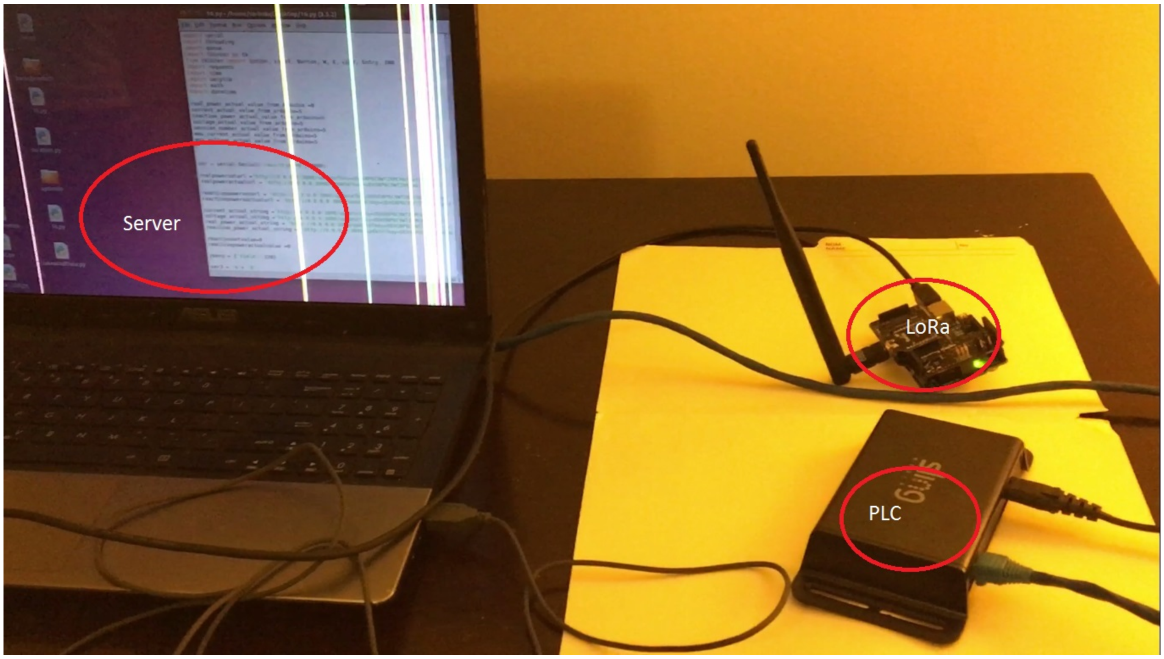


Figure 3.17: Server side of the PCL + LoRa link

### 3.3.5 Two Channel Model

Based on the results, it can observe that LoRa based model can be improved by using frequency division multiplexing instead of using time division multiplexing. But the cost of the system and the power consumption of the system will be increased due to the addition of new components.

As shown in the figure 3.18 data are taken from the inverter by the Arduino MEGA which runs algorithm 1 and algorithm 2. Those data are then sent to another Arduino MEGA for data processing and data logging. This model suggests to use two channels for each uploading and downloading.Another Arduino will be added to the receiving side in this model for the data receiving purpose which will reduce the waiting time for data receiving. The primary development will be made to the receiving side with the addition of a new Arduino for data receiving. It reduces the waiting time for data receiving.

## 3.4 Testing

The proposed LoRa based system has been prototyped and tested in the field environment. And it successfully communicated over 4.5 km range at the medium power level, and after decoding, values are displayed on display and stored in the SD card.

### 3.4.1 Range Testing

After prototyping the proposed loRa based system. It has been tested to measure the range, using the same method described in 2.2.

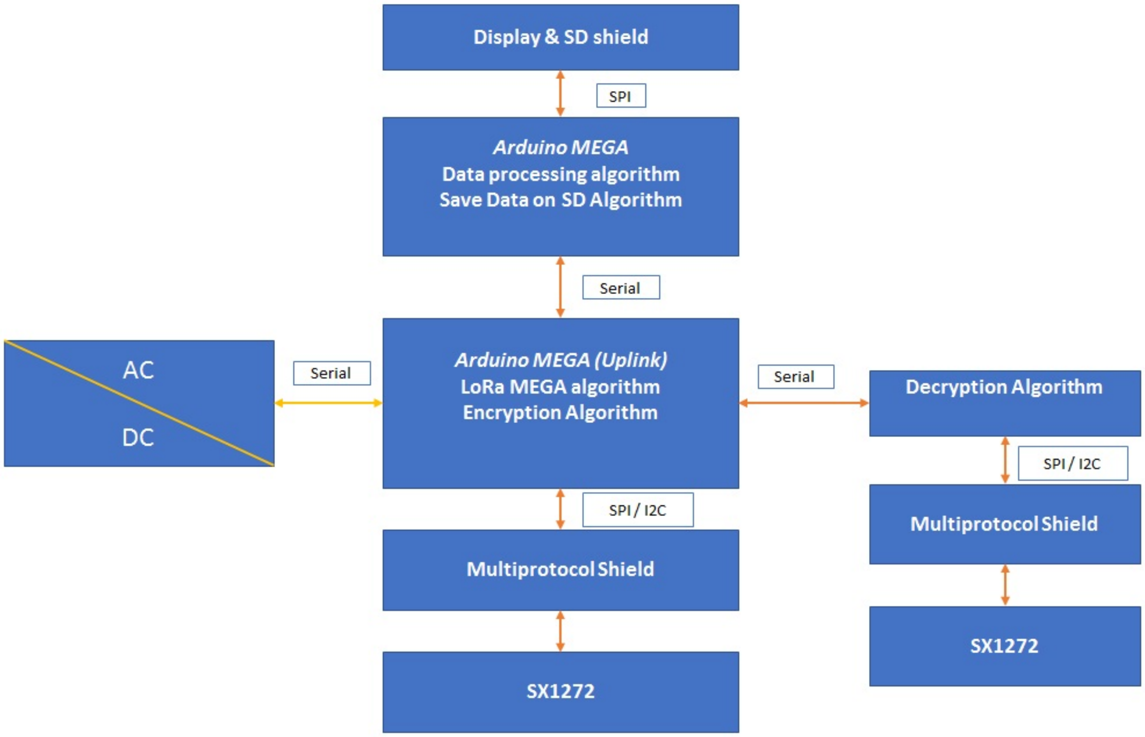
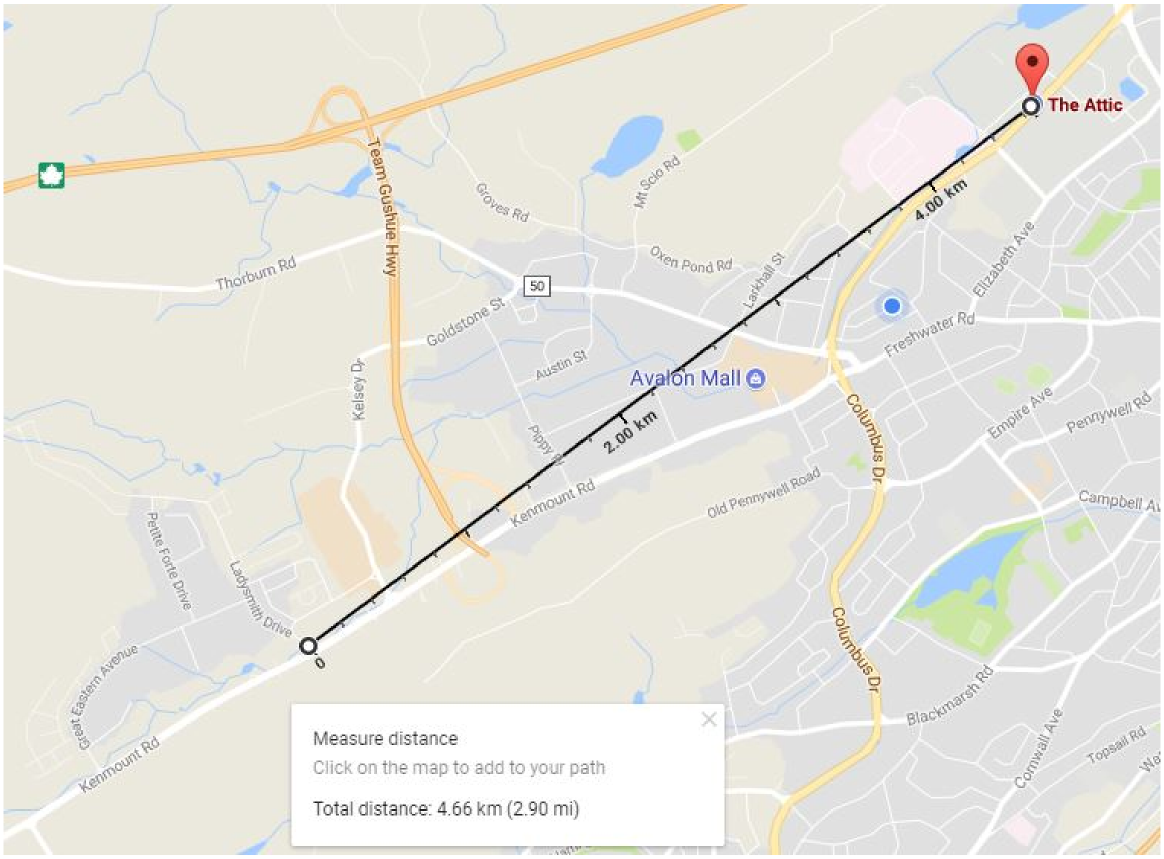


Figure 3.18: Flow Chart for Two Channel Model(Inverter Side)



Measure distance

Click on the map to add to your path

Total distance: 4.66 km (2.90 mi)

Figure 3.19: LoRa range testing results

### 3.4.2 Testing with An Actual Inverter

This project is being funded by the NSERC Energy Storage Network fund. The network is divided into Four Main themes and the research documented in this thesis is carried out under theme 2. The objective of the theme two is to develop power converters for Energy Storage systems. And the aim of the project 2.4 which is conducted in the MUN is to create low-cost SCADA system for a power converter which is developed at the University of New Brunswick (UNB).

At the end of the development of the system, it has been tested with the inverter developed in UNB Sustainable Power Research Lab. Figure 3.20 show the inverter developed by UNB. They are using a DSP2407A board for communication purposes with the inverter and as shown in the Figure 3.21 that board has an RS232 port for UART communication. The inverter side of the developed communication channel has been connected to the serial port. Data string sent by the inverter according to the protocol shown in the Appendix D has been successfully decoded by the system and communicated to the other side through the LoRa link. Figure 3.22 shows data shown on display on the developed system as well as the Inverter itself.

**Decoding:**

Sample data string:

**55AA0200000064BF07001B4A76001B3E1E00A6009C02400FF71C643B1231**

As per the Appendix D first two bytes represent the synchronous word and next four bytes represent the status. Where next twelve bytes represent data as current, voltage and power respectively in four bytes. The number of samples, maximum voltage and the maximum current value and the version number sent in two bytes each respectively. Following equations are implemented in the Arduino Mega. Code is

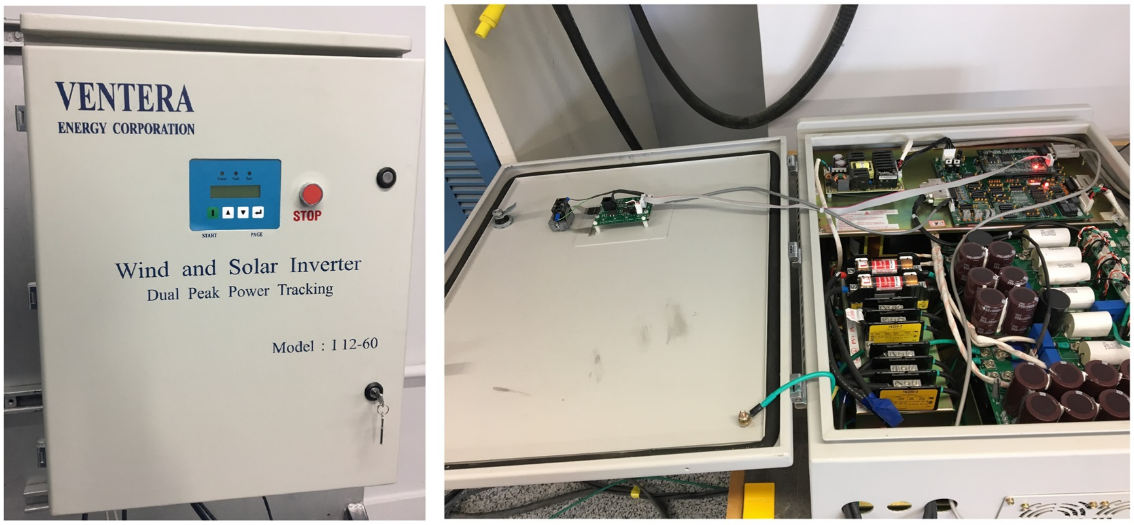


Figure 3.20: Wind and Solar Inverter Developed by UNB

shown in the Appendix B. Equation 3.4 shows the decoding of current and the equation 3.5 shows the decoding of the voltage and the equation 3.6 shows the decoding of the real power.

$$Current = \frac{\sqrt{\frac{Current}{Samples}} * currentmax}{4096} \tag{3.4}$$

$$Voltage = \frac{\sqrt{\frac{Voltage}{Samples}} * currentmax}{4096} \tag{3.5}$$

$$Realpower = \frac{Power * voltagemax * currentmax}{samples * 4096 * 4096} \tag{3.6}$$

## 3.5 Discussion

This section critically reviews the developed channel regarding security, reliability, scalability and the quality of service.

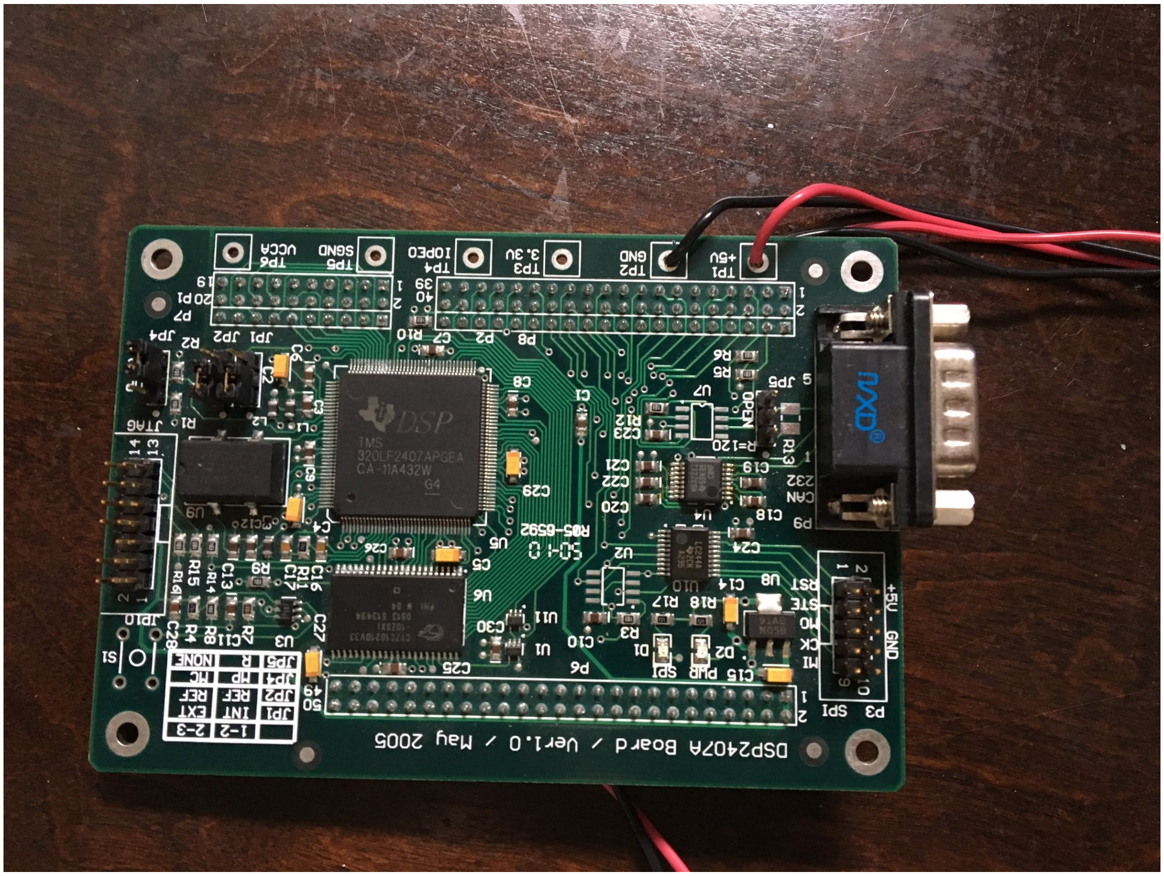


Figure 3.21: DSP board used for communication

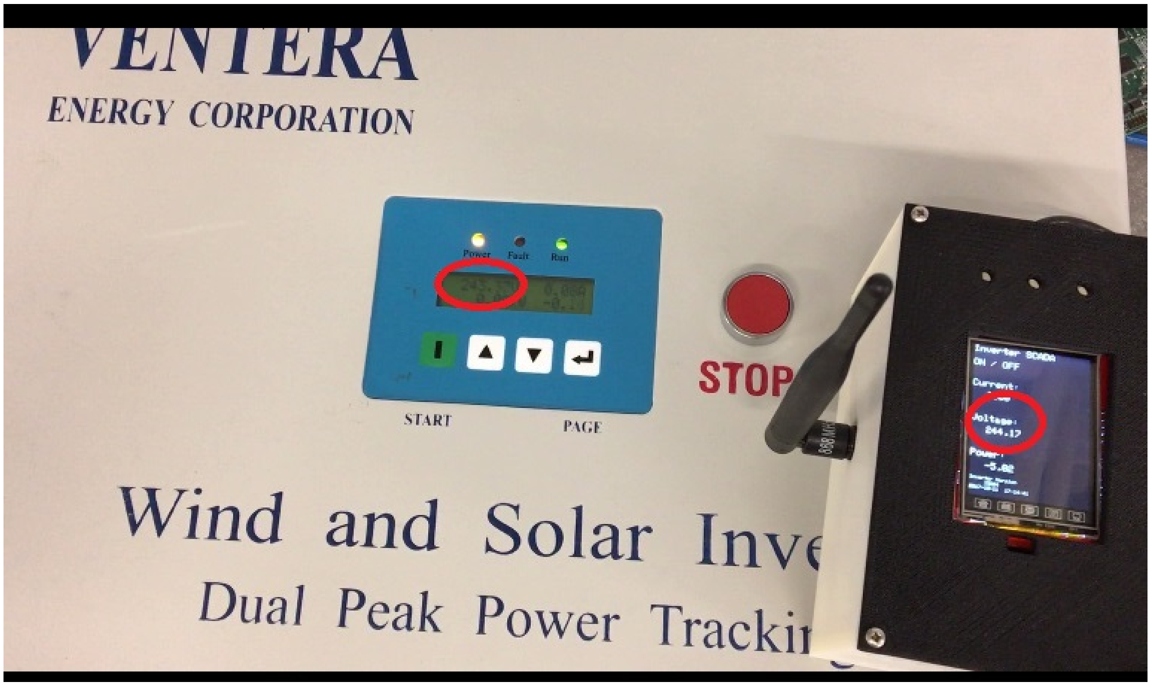


Figure 3.22: Displaying decoded data on the system and the inverter

### 3.5.1 Security

Security of the developed channel has been achieved through an embedded encryption while it ensures the confidentiality of data by preventing disclosure of information to unauthorized persons. Cyclic redundancy check algorithm embedded into the LoRa algorithm guarantees the integrity of data. There are some issues regarding availability where a hacker can jam the signal which is common to every wireless technique. Since there is a local storage of data, short-term data jamming can be ignored.

Overall, the developed system has an improved security compared to a LoRa link without encryption.

### 3.5.2 System Reliability, Robustness, and Availability

Reliability of the system is critical in smart grid applications. LoRa itself has inherent robustness. The addition of local storage has improved the data redundancy.

The proposed two-channel model is more reliable than the time division multiplexing model.

### 3.5.3 Scalability

In the aspect of scalability, the developed system has many advantages compared to available technologies due to the open source nature. Developers can comfortably adopt this system to their smart grid applications while ensuring all security and reliability aspects of the system.

## 3.6 Conclusion

Monitoring and controlling of energy storage have become highly important for dispatching purpose. This chapter mainly discusses the developments of the loRa based communication link for monitoring and control of a remote inverter.

This chapter proposes and implements algorithms to improve the performance of the LoRa link by adding security features, reducing power consumption and improved data redundancy. To enhance security encryption algorithm has been added. Power controlling algorithm has been introduced by using range testing data. Data redundancy has been increased with using local storage for the inverter side. Local display is being provided to the system to improve the user-friendliness as well.

From field testing, it has been verified that this communication link can be used for secure communication between a SCADA and a controlled device. Though the system is specially developed for an inverter, this can be easily converted to be used to make secure communication in power system dispatching.

# Chapter 4

# Reliability of developed LoRa based communication system

## 4.1 Introduction

The primary objective of this chapter is to evaluate the reliability of a communication link developed to serve the communication requirement between the server and the remotely located grid-tied inverter.

SCADA systems are designed to provide and maintain the high level of confidence demanded power system operation [25]. To keep a secure and economical operation of the SCADA system reliable communication link between the grid-connected inverter and the server is very much critical. Otherwise, the failure of SCADA due to any fault of the communication link may consequently lead to severe and costly consequences. Therefore reliability analysis of the communication link is essential both for the design and operation of such a critical system [28].

The communication link may fail to perform when any of its subsystems such as inverter side or server side fails, and that may create a massive problem in the whole system. Therefore it is essential to evaluate the reliability of the communication link quantitatively to ensure reliable performance.

From the literature review, it can be identified that the reliability analysis is critical for a SCADA. Therefore, reliability model proposed in this chapter would be useful to complete the design of the SCADA system. In section 4.2 of the chapter, the structure of the system is explained. And in section 4.4, the static fault tree constructed based on the system is detailed. Reliability modeling and Monte Carlo simulation approach is in section 4.5. In section 4.6 numerical study has been conducted to evaluate the reliability of the system.

## 4.2 Structure of the communication link

Figure 4.1 shows the structure of the communication link with two primary subsystems with several modules. Inverter and server sides are the primary subsystems in the formation of the communication link other than the communication channel in between those two subsystems. The circuit board (M1) which mainly comprises the RS232 interface communicates through UART protocol with the inverter. Then the circuit board sends the data string taken from the inverter to the Master Arduino MEGA board (M2) which acts as the master on the inverter side. Circuit board and the Master Arduino MEGA board communicate through the UART protocol. Afterward, data will be encrypted for security purposes before sending it to LoRa modules for transmission. For the transfer of data, the Master Arduino MEGA board (M2) communicates through SPI protocol with the loRa shield (M3), and that will send the data to loRa module (M4). LoRa module (M4) and the loRa shield ($M3$) communicates through SPI protocol. Then, data will be transmitted via the communication channel (MC) to the server side. In the meantime, after acquiring the data from the inverter, the Master Arduino MEGA board (M2) sends the data to another Arduino MEGA board (M5) for local displaying and storing purposes. This Arduino MEGA board (M5) is represented as the slave Arduino MEGA board (M5) in the figure 4.1, and this communicates with Master Arduino MEGA board through UART protocol. This slave Arduino MEGA board (M5) is connected to TFT shield which comprises TFT display (M7) and the SD card (M8) using for local displaying and storing of data. Both units are communicating with the slave Arduino MEGA board (M5) using SPI protocol separately. On the other hand server side also comprises a loRa module (M8) which communicates with the loRa module (M4) on the inverter side via the communication channel. This loRa module (M8) connected to loRa shield (M9) on the server side through SPI protocol and the other end of the shield is connected to Arduino UNO board (M10). This board takes the data transmitted from the inverter side and send to the server computer through USB interface after decrypting the data. Also, this performs the encryption of data when commands from the server side need to be sent to the inverter side through the loRa communication link. The reliability of these modules is subjected to study under the proposed reliability approach in this chapter in following sections.

## 4.3 Reliability modeling of the communication link

In this research, static fault tree method is applied to create a reliability model for the communication link. This model is a representation of the failure logical relationships of the analyzed components. For the logic diagram construction, AND and OR basic logic gates have been used. Construction of reliability model for the communication link has been done in two steps. Firstly, reliability model has been build for the basic unit which is the LoRa link and studied its reliability indices using the Monte

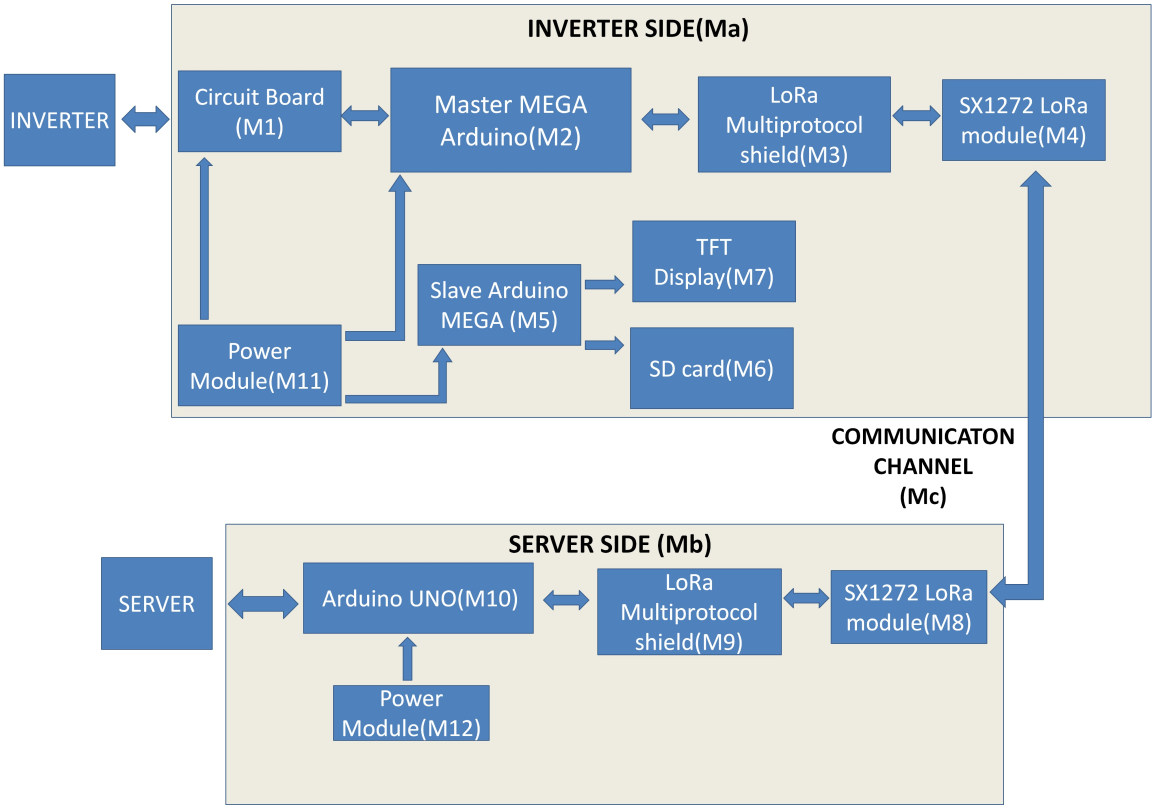


Figure 4.1: Structure of the communication link

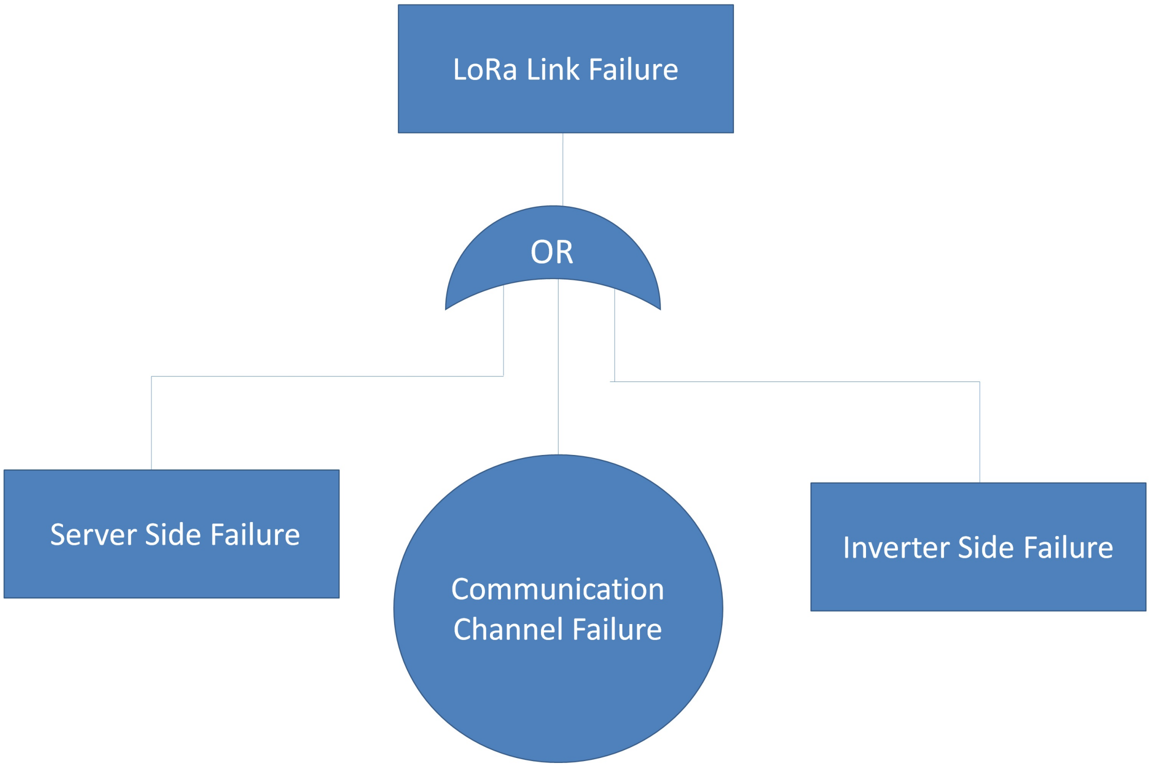


Figure 4.2: Sub reliability model for LoRa link

Carlo simulation. Then the reliability model for the local storage has been added to the primary reliability model to check the modification in reliability indices of the system by adding the local storage. During the construction of reliability model for the LoRa link, the total system has been divided into two primary subsystems to reduce the complexity. Therefore, sub reliability models have been constructed for each subsystem. And in each subsystem, there are some necessary functional modules. Finally, those subsystem models have been combined to build the reliability model for the LoRa link.

### 4.3.1 Sub-reliability model for LoRa link

The main function of the communication link which the data transmission from inverter side to the server side is carried out by this LoRa link using the server side and inverter side LoRa based communication modules. This LoRa link (G1) has been divided into two subsystems as server side(Ma) and inverter side (Mb) other than the communication channel (Mc) as shown in the reliability model for the LoRa link in Figure 4.2. In this sub-reliability model there two other basic sub-reliability models for server-side (Ma) and inverter side (Mb) as shown in the figure 4.3 and figure 4.4 respectively. Here $\lambda$Mc, $\mu$Mc denotes the failure rate and the repair rate of the communication channel which is the other module in this sub-reliability model.

#### Sub-reliability model for server side

Server-side (Mb) represents the modules on the server side of the loRa communication link which receive the data coming from the inverter side and send those to the server computer while sending the commands coming from the server computer to the inverter side through the communication channel. This consists four function modules: Power module (M12), LoRa module (M8), LoRa multiprotocol shield (M9) and Arduino UNO board (M10). The sub reliability model for the server side is shown in the figure 4.3 in which $\lambda$M12, $\lambda$MHW8, $\lambda$MSW8, $\lambda$M9, $\lambda$MHW10, $\lambda$MSW10, $\mu$M12 $\mu$MHW8, $\mu$MSW8, $\mu$HWM9, $\mu$MHW10, $\mu$MSW10 denote the failure rate and the repair rate of each module M12, M8, M9 and M10 in the sub-system (Mb). In this subsystem failure of any of the module can cause the failure of the total sub-system which can be considered as in series reliability. Therefore each module is connected to subsystem through OR gate.

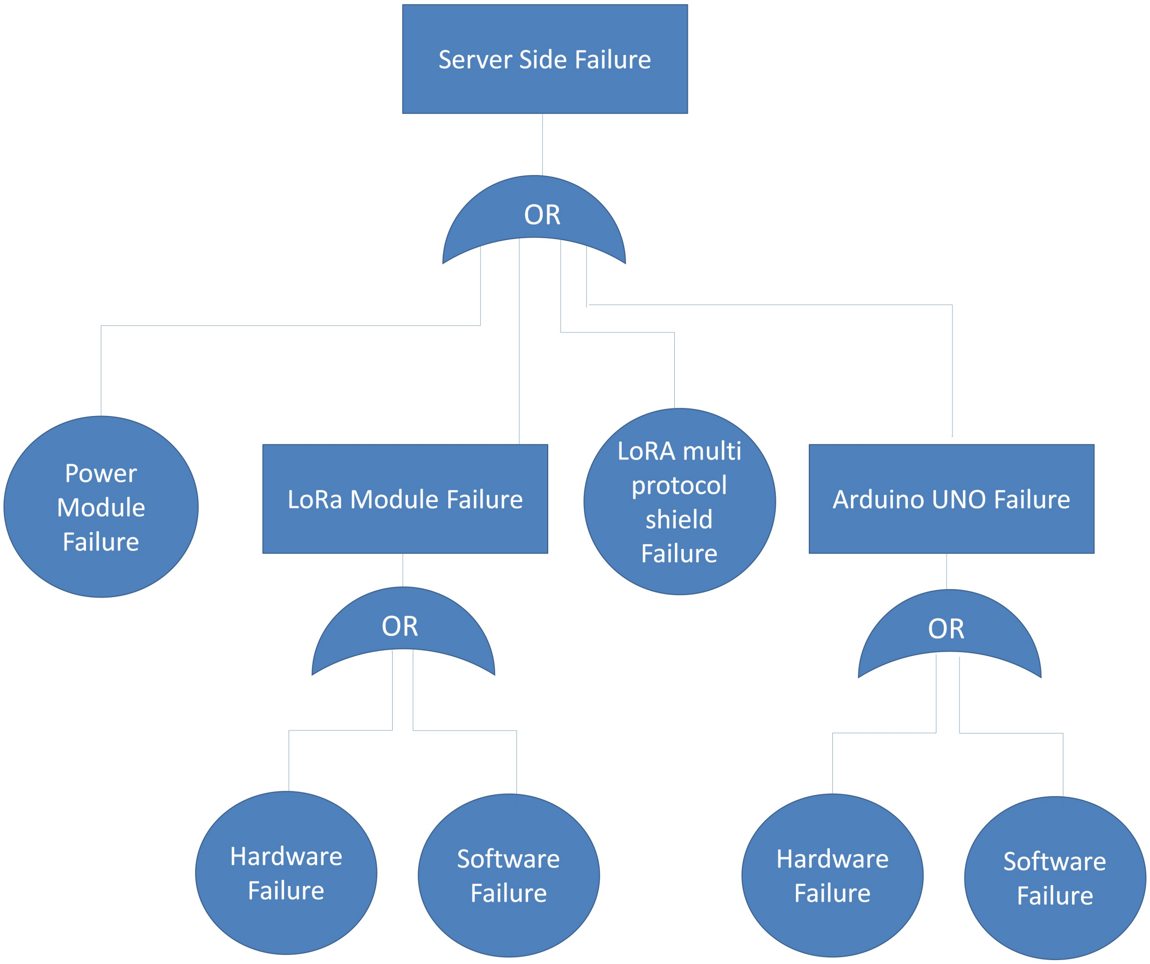


Figure 4.3: Sub-reliability model for server side

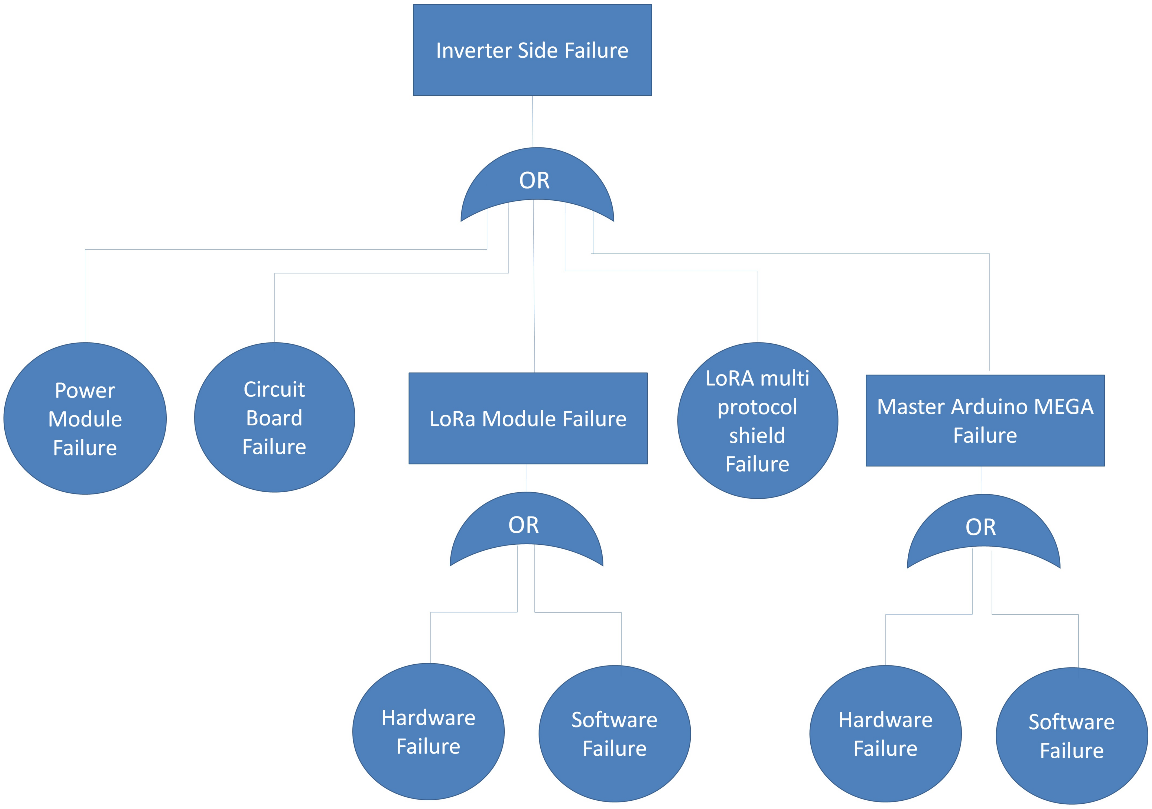


Figure 4.4: Sub-reliability model for inverter side

**Sub-reliability model for inverter side**

This is another key subsystem in this reliability model which takes the data from the inverter and send it to the server via loRa communication link. Before sending the signal via the communication channel, this subsystem encrypts the message for security purposes. At the same time it accepts the signals from the server side, and after decrypting them, it sends it to the inverter. Figure 4.4 show the sub-reliability model for this inverter side (Ma) and as it shows there are five function modules as such power supply module(M11), circuit board module (M1), loRa module (M4), loRa multi protocol shield (M3) and the Master Arduino Mega board (M2). Moreover, $\lambda$M11, $\lambda$M1, $\lambda$MHW4, $\lambda$MSW4, $\lambda$M3, $\lambda$M2, $\mu$M2, $\mu$M3, $\mu$MSW4, $\mu$MHW4, $\mu$M11, $\mu$M1 denote the failure rate and repair rate of those modules in the reliability model.

As explained on the server side, failure of any of these modules will directly cause a failure in the subsystem, therefore to the point of reliability these function modules are connected in series. Due to that, the gate OR has been used to denote that relationship in a logical manner.

### 4.3.2 Sub-reliability model for local storage

Other than the communication from inverter side to the server side, there is another subsystem in this communication link to enhance the user-friendliness and the reliability of data. Physically this system also acts as the part of the inverter side of this communication link. It has two main functional components such as SD card and the LCD where SD card is used to save the data locally, and LCD is used to display the values of the critical parameters like current, voltage, power, date and time. The data from the inverter is sending into this subsystem, before the encryption of the data. Then data will be saved in each 5 second in CSV format for seven days, and it

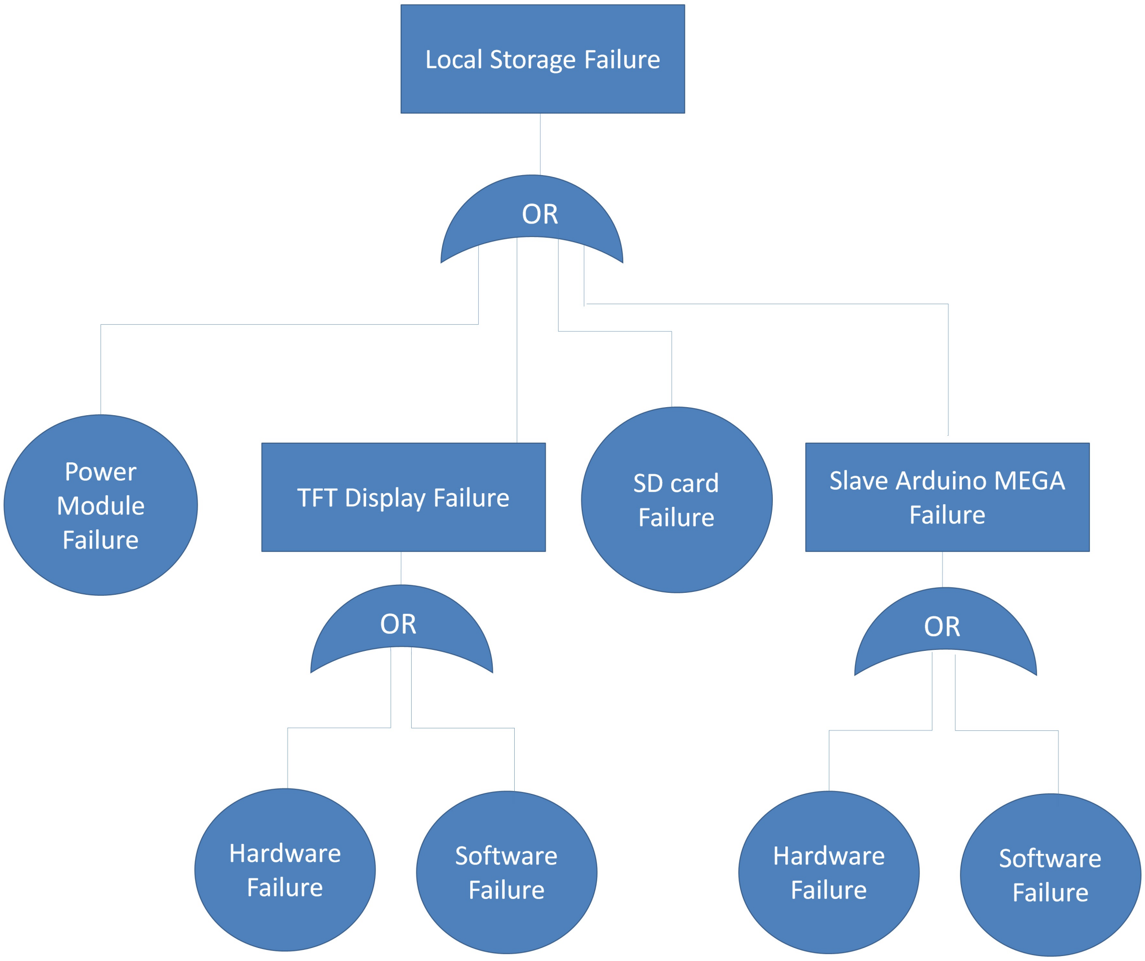


Figure 4.5: Sub-reliability module for local storage

will overwrite after seven days to keep the capacity. Since this data can be directly uploaded to the server, any data loss while the communication via loRa link can get recovered through this method. Three main function modules in this subsystem are power module (M11) and Slave Arduino MEGA board (M5) as shown in the figure 4.5. $\lambda$M11, $\lambda$MHW5, $\lambda$MSW5, $\lambda$M6, $\lambda$MHW7, $\lambda$MSW7, $\mu$M11, $\mu$MHW5, $\mu$MSW5, $\mu$M6, $\mu$MHW7, $\mu$MSW7 are the failure rates and the repair rates of those modules respectively. In this subsystem also the basic logic gate OR is used to depict the relationship between the function modules.

### 4.3.3 Complete Reliability model of communication link

As it can be seen from the figure 4.2 failure of any side of the communication link can cause the breakdown in the total system. Therefore to depict that relationship in reliability point of view, the primary OR gate is used in the reliability model for the entire system as shown in the figure 4.6. However, to represent the parallel operation of the local storage subsystem and the loRa link subsystem, the basic logic gate AND is used in the complete reliability model in figure 4.6. Once the complete reliability model constructed, the system has been analyzed through the Monte Carlo simulation to determine the reliability indices for the system. During the system investigating it is completed in two steps as such firstly without the local storage subsystem and secondly with that subsystem to observe the effect on that particular subsystem to the system reliability indices.

## 4.4 Fault Tree Analysis of the system

Fault tree analysis can be quantitative and qualitative, or both are depending on the scope of review. In this chapter qualitative analysis of fault tree is carried out through Monte Carlo simulation, and that has been executed on the model developed in section 4.3 according to the Monte Carlo simulation steps explained in section 4.5. In the quantitative evaluation of the fault tree, the first step is to develop the structural representation which has been done in section 4.5. In this section, the Boolean representation of this system model which is required for the quantitative analysis is presented.

For the total system ;

$$T = G_1.G_2 \tag{4.1}$$

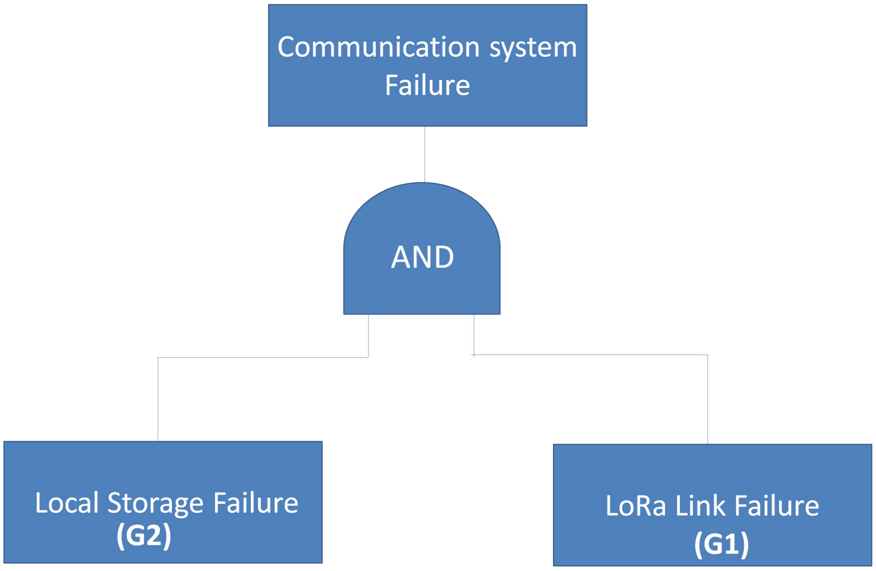


Figure 4.6: Complete Reliability model of communication link

For the LoRa link ;

$$G1 = M_a + M_b + M_c \tag{4.2}$$

For the inverter side;

$$M_a = M_1 + M_2 + M_3 + M_4 + M_{11} \tag{4.3}$$

For the server side;

$$M_b = M_8 + M_9 + M_{10} + M_{12} \tag{4.4}$$

For the local storage sub system;

$$G_2 = M_{11} + M_5 + M_6 + M_7 \tag{4.5}$$

To enhance the logical analysis, this qualitative study is being used. However, to execute a numerical evaluation of a fault tree, the probabilistic data of the basic events are essential. Therefore, reliability and availability prediction techniques use actual test or field use data to set up the quantitative values [29].

## 4.5 Monte Carlo Analysis of reliability model of communication link

In qualitative fault tree analysis determining minimal cut set and minimal path sets and finding common cause failure is important. To find the minimal cut set for the fault tree, one of the primary approaches is Monte Carlo Simulation [37]. Thus in this chapter, this simulation method is adopted to establish the reliability indices of the communication link using the developed system model which described in section 4.3. The procedure which followed to find the reliability indices for the communication link by Monte Carlo simulation is illustrated in the following steps.

**Step 1:** The probability density functions (PDF) were identified for the basic components of the total system. Then according to the reliability distribution of the physical elements, their failure rates, and the repair rates are either collected or derived from [24]. Reliability parameters of the basic components which are used for the simulation are shown in table 1.

**Step 2:** Based on the developed reliability model for the system, system representation blocks are generated in the "Minitab-Companion" software. Then appropriate reliability distributions were selected for basic components, and required parameters for those elements were given into the software.

**Step 3:** Determined the convergence factor evaluate the validity of the reliability indices developed to form the Monte Carlo simulation. To determine the convergence factor, the developed simulation model in the software has been run for different iteration values.

**Step 4:** Once the convergence factor is found, with the aim of finding the reliability indices, the simulation model runs for iteration which satisfies the convergence value. And data of the simulation is adapted to the "Minitab" software to find out the best fit probability distribution for the entire system model. Using the selected PDF, reliability indices were found for the system by taking the parameters needed.

**Step 5:** then step 4 is carried out to find the best fit distribution for subsystems and to find out reliability indices for subsystems as well.

**Step 6:** finally, the local storage subsystem is also added to the "Minitab companion" software model and run the system to find the reliability indices for the complete system as explained in the above steps.

Table 4.1: **Failure and Repair rates of the components in the system [54], [55]**

| **Component** | **Module No.** | **Failure rate $\lambda$** | **Repair rate $\mu$** |
|---|---|---|---|
| Communication channel | $M_c$ | 4.7297 | 1166.27 |
| LoRa module-SS- Hardware | $M_HW8$ | 0.4963 | 3735.42 |
| LoRa module-SS- Software | $M_SW8$ | 0.9771 | 16248.25 |
| LoRa multi protocol shield-SS | $M_9$ | 0.3298 | 2874.83 |
| Arduino UNO-SS-Hardware | $M_HW10$ | 0.3201 | 1688.22 |
| Arduino UNO-SS-Software | $M_SW10$ | 2.0091 | 33265.86 |
| Power module-SS | $M_12$ | 0.6803 | 0.6803 |
| Power module-IS | $M_11$ | 0.6803 | 1846.53 |
| Circuit board-IS | $M_1$ | 0.3171 | 1316.56 |
| LoRa module-IS-Hardware | $M_HW4$ | 0.4963 | 3735.42 |
| LoRa module-IS-Software | $M_SW4$ | 0.9771 | 16248.25 |
| LoRa multi protocol shield-IS | $M_3$ | 0.3298 | 2874.83 |
| Arduino Mega-IS-Hardware | $M_HW2$ | 0.3201 | 1522.58 |
| Arduino Mega-IS-Software | $M_SW2$ | 2.0092 | 33265.86 |
| Power Module, local storage | $M_11$ | 0.6803 | 1846.53 |
| Slave Arduino Mega-Hardware | $M_HW5$ | 0.3201 | 1522.58 |
| Slave Arduino Mega-Software | $M_SW5$ | 2.0092 | 33265.86 |
| SD card-Local storage | $M6$ | 0.2153 | 16224.33 |
| TFT display-Local storage-Hardware | $M_HW7$ | 0.4938 | 4097.65 |
| TFT display-Local storage-Software | $M_SW7$ | 2.0083 | 5782.79 |

### 4.5.1 Reliability Distributions

To find out the probability density functions for basic components, the standard probability distributions are analyzed [38].

- **Exponential:**this distribution is used for constant failure rate $\lambda$ cases. This $\lambda$

is the inverse of the mean time to failure to occur.

$$f(t) = \lambda^{(-\lambda t)} \tag{4.6}$$

- **Weibull:** this distribution can be used for constant, increasing or decreasing failure rate scenarios by selecting corresponding shape parameter $\beta$. Shape parameter is $\beta$ and characteristic life parameter is $\eta$ in the function.

$$f(t) = \frac{\beta}{\eta}(\frac{t}{\eta})^{\beta-1}e^{-(\frac{t}{\eta})^{\beta}} \tag{4.7}$$

- **Normal:** In the distribution function $\sigma$ is the Standard deviation or the scale parameter and $\mu$ is the mean or the location parameter.

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(t-\mu)^2}{2\sigma^2}} \tag{4.8}$$

- **Log normal:** this distribution is a continuous distribution in which the logarithm of a variable has a normal distribution. In the distribution function, $\alpha$ is the scale parameter and $\mu$ is the location parameter as in the normal distribution.

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(\ln(t)-\mu)^2}{2\sigma^2}} \tag{4.9}$$

In this system, mostly there are electronic components, and in the electronic components, the preferred reliability data is the failure rate. And in general, failure rate data for electronic elements are referred to the phase constant failure rate[41].Therefore exponential distribution is considered for all the essential components of the system during the simulation for failure rates. However, for the repair times, the most commonly used distribution is the log normal model since it replicates short duration repair-time, a large number of observations tightly grouped with some modal value [40].

### 4.5.2 Convergence assessment

Since Monte Carlo simulation is a computer simulation method based on the probability theory, to evaluate the reliability indices from this approach, the accuracy of the simulation is required to be assessed. For that purpose, this convergence assessment can be used to evaluate the precision of the simulation results [24].

The convergence factor is defined as,

$$\beta = \frac{\sigma}{\sqrt{N}} \leq \epsilon_r \tag{4.10}$$

where; $\beta$ = convergence factor $\sigma$ = standard deviation N= failure count $\epsilon_r$= prespecified fraction

### 4.5.3 Reliability indices

Reliability of the performance is defined as the ability of an item to perform a required function under given conditions for a given time interval as per the International Standard IEC 60050 (191)[41]. Thus to measure the performance of the system several reliability indices are being used. Using the reliability standards, system reliability can be evaluated as well as the system can be analyzed to identify the suitable reliability improvements which can be taken to enhance the reliability of the system. In this chapter, MTBF, MTTR, availability, unavailability have been computed under the reliability evaluation.

**Mean Time Between Failure (MTBF) :** The expectation of the operating time between failures is indicated by the MTBF. If constant failure rate component is considered MTBF can be represented as the inverse of the failure rate [41].

**Mean Time To Repair (MTTR):** The expectation of the time to restoration is represented by the MTTR. This factor conveys the active corrective maintenance time required for an item to restore up to expected performance level [41].

**Availability:** Availability may be interpreted as the probability that a system is operational at a given point in time or over the period of time [54].

$$Availability(A) = \frac{uptime}{uptime + downtime} = \frac{MTBF}{MTBF + MTTR} \tag{4.11}$$

$$Unavailability = 1 - Availability(A) = \frac{MTTR}{MTBF + MTTR} \tag{4.12}$$

## 4.6 Numerical study

### 4.6.1 Convergence Assessment

The plot against convergence factor ($\beta$) and the Monte Carlo simulation number is shown in the figure 4.7. According to that is can be seen the convergence factor seems to remain unchanged once the simulation count is above 500,000. And the corresponding convergence factor 0.35% is then selected as the termination condition for the Monte Carlo simulation. In other words, the validity of the reliability indices and the following analysis carried out on this system ensures once the simulation count exceeds 500,000 with particular convergence factor.

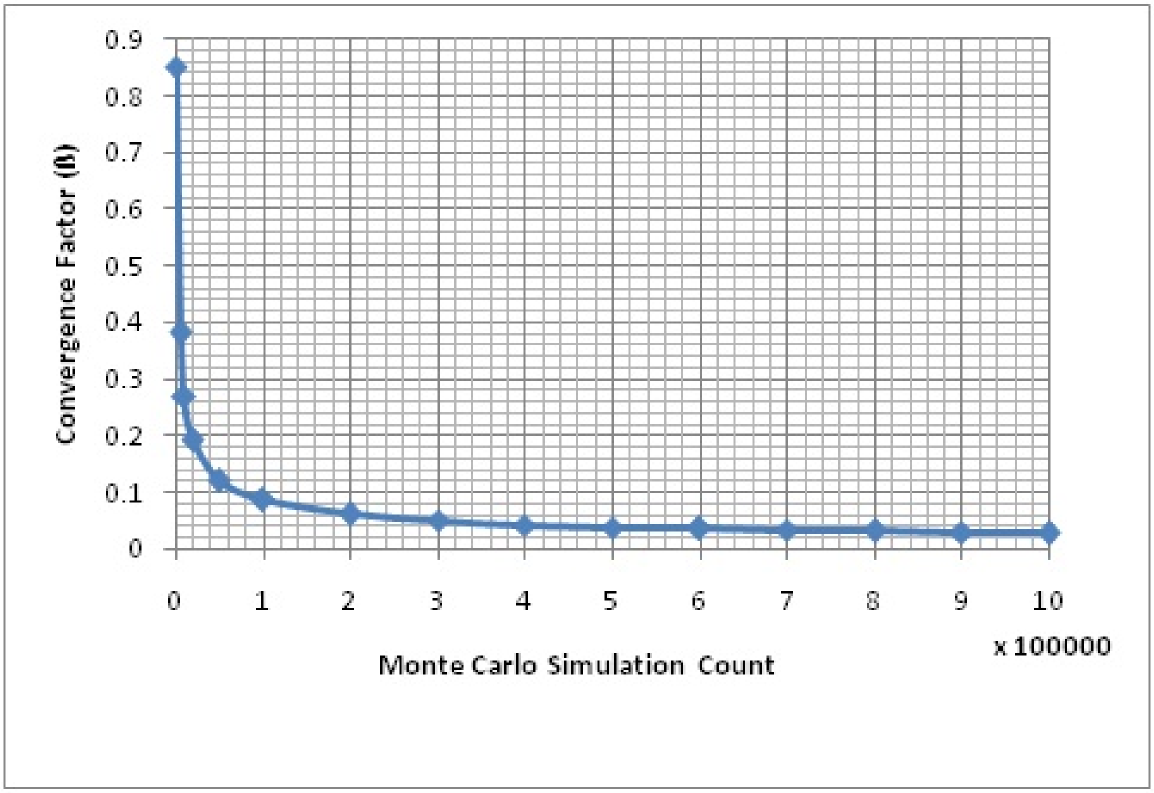


Figure 4.7: Convergence factor against Monte Carlo simulation count

### 4.6.2 Reliability indices for the system

Using available data and assumed data values for MTBF, MTTR, Availability, and Unavailability reliability indices were taken from the simulation for each function modules and the subsystems in the communication link as well as the values for the entire communication system. According to the availability calculations, the communication channel has been identified as the most critical module. Since the communication channel is vulnerable to many external factors, its availability is lower compared to other modules in the system.

The local storage system is added to the LoRa link in this design to complete the communication link with the purpose of enhancing the reliability of data transmission. In other words, even data loss occurred during the transfer of data via LoRa link, since that data will be stored locally in the SD card on the inverter side, missing data can be recovered. That strategy seems proved by the reliability evaluation since the availability of the communication link seems improved, where the availability of the system without local storage is 0.994288, and the with the local storage is 0.99946 when the local storage subsystem is added to the communication link.

## 4.7 Conclusion and Future Works

The communication link is a crucial component in SCADA system, especially for a remotely located inverter. Since this system facilitates the need of passing data from inverter to the SCADA as well as SCADA to inverter, to control and monitor the inverter in an efficient manner, the reliability of this communication link is also outstanding. In this chapter, a comprehensive static fault tree based Monte Carlo simulation method is proposed to evaluate the reliability of the communication link. Fault tree method is one of the most utilized methods to construct the reliability

Table 4.2: **Reliability indices of components of the system**

| **Component** | **No.** | **MTBF (hr)** | **MTTR (hr)** | **Avail-ability** | **Unavail-ability** |
|---|---|---|---|---|---|
| LoRa link | $G_1$ | 888.4741 | 5.543 | 0.99379 | 0.00620 |
| Server side | $M_b$ | 1820.1576 | 17.6188 | 0.99041 | 0.00959 |
| Inverter side | $M_a$ | 1707.6422 | 6.765 | 0.99605 | 0.00395 |
| Communication channel | $M_c$ | 1852.1180 | 7.50694 | 0.995963 | 0.00404 |
| LoRa module-SS | $M_8$ | 5945.3112 | 4.3807 | 0.999263 | 0.00074 |
| LoRa multi protocol shield-SS | $M_9$ | 26561.5524 | 3.04567 | 0.999885 | b 0.00011 |
| Arduino UNO-SS | $M_1$0 | 3760.83494 | 5.45225 | 0.998552 | 0.00145 |
| Power module-SS | $M_1$2 | 12877.23992 | 4.74339 | 0.999631 | 0.00037 |
| Power module-IS | $M_1$1 | 12877.23992 | 4.74339 | 0.999631 | 0.00037 |
| Circuit board-IS | $M_1$ | 27624.4836 | 6.65109 | 0.99975 | 0.00024 |
| LoRa module-IS | $M_4$ | 5945.3112 | 4.38243 | 0.999263 | 0.00074 |
| LoRa multi protocol shield-IS | $M_3$ | 26561.5524 | 3.04259 | 0.999885 | 0.00011 |
| Arduino Mega-IS | $M_2$ | 3760.8349 | 6.01822 | 0.998402 | 0.00160 |
| Local storage sub system | $G_2$ | 1529.6019 | 14.5216 | 0.990595 | 0.00940 |
| Communication link | $T$ | 155.1403 | 0.0838 | 0.99946 | 0.00054 |

model for a particular system. Therefore, in this research also fault tree method is used to build the reliability model for the system in which relationship between the essential components of the communication link is accurately depicted using basic logic gates. Then the Monte Carlo simulation approach is employed to analyze the static fault tree to evaluate the reliability indices for the communication link system. The analysis shows that the communication link has the lowest availability before adding the local storage. Addition of a data storage parallel to the system has increased the availability of the system.

During the formation of the fault tree which is the protection against surges or lightning have not been considered. Even though the reliability of each component is high with given data from taken in laboratory environment protection against overloads and overvoltages will become a significant factor in reliability. Therefore, in the future, these components need to be tested for few years and obtain reliability data to get exact numbers.

# Chapter 5

# Summary and Future Recommendations

Due to the increasing number of energy storage systems that are connected to the power system it has become a necessity to monitor and control energy storage systems. Monitoring and control are done through a SCADA system, and each component would be needed to be connected to a SCADA, and each shall be needed to be in a network. Therefore, there is a requirement for secure and reliable communication method. To fulfill this gap, this research conducted based on three primary objectives,

1. to identify a low cost, long range, secure and reliable communication method for monitoring and control of a grid-connected inverter.
2. to develop the selected communication method regarding security and reliability.
3. to develop a model to analyze the reliability of the developed communication method

## 5.1 Summary

### 5.1.1 Research Summary Based on Objective 1

Initially, a thorough literature review has been conducted to find out available low-cost communication systems with more than two-kilometers range. Based on the review three low-cost wireless communication methods have been identified such as LoRa, Radioteletype, and UHF/ VHF data communication. Then those methods were tested and compared concerning cost, range, data rate and the security.

Based on the study LoRa has been recognized as the most suitable communication method for the selected application.

### 5.1.2 Research Summary Based on Objective 2

The second phase of the study was conducted to develop the selected communication method further, to be used in monitoring and control of inverters. In the application, it is a requirement to have a secure and reliable communication link. Therefore, the selected communication method has been improved concerning security with an addition of encryption algorithm. To improve the reliability, another wired communication link is parallelly added to the wireless connection. Then the challenge was to improve the data redundancy and then it has been overcome by adding a local data storage for the developed system.

### 5.1.3 Research Summary Based on Objective 3

Finally, the third objective of the research is to develop a model to analyze the reliability of communication method. A novel approach which combines both static Fault Tree Analysis and Monte Carlo simulation method has been used to achieve the third objective. The system has been divided into several primary subsystems to analyze the reliability, and static fault tree is developed based on that. Then the Monte Carlo simulation has been conducted to calculate the reliability indices of the system. Finally, sensitivity analysis has been performed using the Monte Carlo simulation approach.

## 5.2 Significant Contributions

To summarize, this thesis has made following key contributions in the field of SCADA systems by fulfilling all of the outlined research objectives,

1. A framework has been developed to compare communication technologies:

   Under the first objective, the thesis developed a framework to compare available communication technologies.

2. Improved a LoRa communication link regarding security and the reliability:

3. Successfully demonstrated a combination of LoRa and PLC:

4. Reliability model has been developed to analyze the reliability of the developed communication method:

## 5.3 Directions for Future Work

- **Develop directional antenna:**

  The range of the LoRa link can be improved with a directional antenna, based on features of the application usage of an omnidirectional antenna can be omitted. Therefore, for monitoring and control of grid-connected inverters, directional antennas can be used, and it will allow reducing the energy consumption for data transmission as well. Therefore it would be vital to developing directional antenna for LoRa based on the application.

- **Develop a LoRaWAN network for Inverters:**

  Since the population of inverters are getting high and the requirement for the monitoring and control of inverters are also getting high Long Range Wide Area Network can be developed specially for the monitoring and control of inverters.

- **High secure encryption for low power processors:**

  This is another research wing that needs to be developed the use of encryption methods such as AES; Elliptic Curve cryptography would take high processing power from a small processor. Therefore, there is a requirement in the field of cryptography to develop encryption method that can be implemented in a small processor.

- **Reduce power consumption of the developed system:**

  Phantom load of the developed system is high compared to the power consumed by the communication link itself. It is in the other way in a conventional system. Therefore there is a requirement to redesign the system to reduce the power consumption.

## 5.4 List of Publications

All the contributions in the thesis are published in the following technical papers and posters:

**Journal Articles**

1. Kumbalatara Arachchige, M. T. Iqbal, G. Mann, Data logging and control of

a remote inverter using LoRa with local storage, submitted with IEEE Access 2017

### Peer-reviewed Conference Articles

2. Lasagani K A. Terashmila, Tariq Iqbal, George Mann, A comparison of low-cost wireless communication methods for remote control of grid-tied converters, presented at CCECE 2017, Windsor ON Canada.

### Abstract-reviewed Conference Articles

3. Terashmila Lasagani, Tariq Iqbal, George Mann, Reliability Analysis of a Communication Link Developed for a SCADA System Using Monte-Carlo Simulation Approach, accepted to present at 26th IEEE NECEC conference 2017.

4. Terashmila Lasagani, Tariq Iqbal, George Mann, Data Logging and Control of a Remote Inverter Using LoRa with A Local Storage, accepted to present at 26th IEEE NECEC conference 2017.

5. Terashmila Lasagani, Tariq Iqbal, George Mann, the Best communication method for remote control of grid-tied converter for an energy storage system, presented at 25th IEEE NECEC conference 2016.

### Poster presentations

6. L. K. A. Terashmila, T. Iqbal and G. Mann, Data logging and control of a remote inverter using LoRa, Arduino, RS232 and SD card, Poster session presented at: NESTNet Technical Conference. 1st Annual conference. 21-22 July, 2017; Toronto, ON.

# Bibliography

[1] H. Kanchev, D. Lu, F. Colas, V. Lazarov and B. Francois,. 2011 Oct. "Energy Management and Operational Planning of a Microgrid With a PV-Based Active Generator for Smart Grid Applications." IEEE Transactions on Industrial Electronics 4583-4592.

[2] P. Palensky, D. Dietrich. 2011. "Demand side management: Demand response, intelligent energy systems, and smart loads,." IEEE Trans. Ind. Inf. 7 (3): 381-388.

[3] Q.Yang, J. A. Barria, and T. C. Green,. 2011. "Communication infrastructures for distributed control of power distribution networks." IEEE Trans. Ind. Inform. 316-327.

[4] V. C. Gungor, B. Lu, G. P. Hancke,. 2010 . "Opportunities and challenges of wireless sensor networks in smart grid." IEEE Trans. Ind. Electron 57 (10): 35573564.

[5] M. Y. Zhai. 2011. "Transmission Characteristics of Low-Voltage Distribution Networks in China Under the Smart Grids Environment." IEEE Transactions on Power Delivery 26 (1): 173-180.

[6] Sara Bavarian, Lutz Lampe. 2012. "Communications and access technologies for smart grid." In Smart Grid Communications and Networking, by Zhu Han, H. Vincent Poor Ekram Hossain, 111-146. Cambridge University Press.

[7] Zhiming Wang, Weichun Ge, Chenggang Wang and Chaoming Zeng,. 2012. "The applications of networking of consumption data acquisition system by combining broadband powerline communication and wireless communication." IEEE PES Innovative Smart Grid Technologies. Tianjin. 1-4.

[8] Hossain, Zhu Han, H. Vincent Poor. 2012. Smart Grid Communications and Networking. Cambridge University Press.

[9] Zhang Donglai, Fan Hong, Wang Chao and Zhou Ying,. 2005. "Low Cost and High Performance Power System Telemetry Data Transmission System Based on Embedded Ethernet and ADSL." IEEEPES Transmission and Distribution Conference and Exposition: Asia and Pacific. Dalian. 1-5.

[10] H. J. Zhou, C. X. Guo and J. Qin,. 2010. ”Efficient application of GPRS and CDMA networks in SCADA system.” IEEE PES General Meeting. Minneapolis. 1-6.

[11] C. Fu and Z. Ni. 2015. ”The application of embedded system in Supervisory Control and Data Acquisition System (SCADA) over wireless sensor and GPRS networks.” 2015 IEEE 9th International Conference on Anti-counterfeiting, Security, and Identification (ASID). Xiamen. 81-85.

[12] M. Conti, D. Fedeli and M. Virgulti,. 2011. ”B4V2G: Bluetooth for electric vehicle to smart grid connection.” 011 Proceedings of the Ninth International Workshop on Intelligent Solutions in Embedded Systems. Regensburg. 13-18.

[13] G. Eason, B. Noble, and I.N. Sneddon, On certain integrals of Lipschitz-Hankel type involving products of Bessel functions, Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955.

[14] J. So, D. Kim, H. Kim, H. Lee and S. Park, ”LoRaCloud: LoRa platform on OpenStack,” 2016 IEEE NetSoft Conference and Workshops (NetSoft), Seoul, 2016, pp. 431-434. doi: 10.1109/NETSOFT.2016.7502471

[15] D. H. Kim, J. Y. Lim and J. D. Kim, ”Low-Power, Long-Range, High-Data Transmission Using Wi-Fi and LoRa,” 2016 6th International Conference on IT Convergence and Security (ICITCS), Prague, 2016, pp. 1-3. doi: 10.1109/ICITCS.2016.7740351

[16] Y. Yao, X. Chen, L. Rao, X. Liu and X. Zhou,. 2017. ”LORA: Loss Differentiation Rate Adaptation Scheme for Vehicle-to-Vehicle Safety Communications.” IEEE Transactions on Vehicular Technology 2499-2512.

[17] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen and M. Pettissalo, ”On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology,” 2015 14th International Conference on ITS Telecommunications (ITST), Copenhagen, 2015, pp. 55-59. doi: 10.1109/ITST.2015.7377400

[18] Q. Yang, J. A. Barria and T. C. Green. 2011. ”Communication Infrastructures for Distributed Control of Power Distribution Networks.” IEEE Transactions on Industrial Informatics 316-327.

[19] D. Dzung, M. Naedele, T. P. Von Hoff and M. Crevatin. 2005. ”Security for Industrial Communication Systems.” Proceedings of the IEEE 1152-1177

[20] K. Moslehi and R. Kumar. 2010. ”A Reliability Perspective of the Smart Grid.” IEEE Transactions on Smart Grid 57-64.

[21] Gungor, V. C. 2011. ”Smart Grid Technologies: Communication Technologies and Standards.” IEEE Transactions on Industrial Informatics 529-539.

[22] H. Li and W. Zhang. 2010. "QoS Routing in Smart Grid." IEEE Global Telecommunications Conference GLOBECOM 2010. Miami. 1-6.

[23] Ou, Yong, and Joanne Bechta Dugan. "Sensitivity analysis of modular dynamic fault trees." Computer Performance and Dependability Symposium, 2000. IPDS 2000. Proceedings. IEEE International. IEEE, 2000.

[24] P. Zhang and K. W. Chan, "Reliability Evaluation of Phasor Measurement Unit Using Monte Carlo Dynamic Fault Tree Method," in IEEE Transactions on Smart Grid, vol. 3, no. 3, pp. 1235-1243, Sept. 2012.

[25] Chun-Lien Su and Ya-Chin Chang, "A SCADA system reliability evaluation considering performance requirement," 2004 International Conference on Power System Technology, 2004. PowerCon 2004., 2004, pp. 574-579 Vol.1.

[26] Dai, Zhi-Hui, Zeng-Ping Wang, and Yan-Jun Jiao. "Reliability evaluation of the communication network in wide-area protection." IEEE Transactions on Power Delivery 26.4 (2011): 2523-2530.

[27] L. Wang, P. P. Gelberger and N. Ramani, "Reliability assessment of the operational functions of a power system control center," 1991 Third International Conference on Probabilistic Methods Applied to Electric Power Systems, London, 1991, pp. 229-234.

[28] Y. Zhang, M. Larsson, B. Pal and N. F. Thornhill, "Simulation approach to reliability analysis of WAMPAC system," 2015 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, 2015, pp. 1-5.

[29] Y. Chen, X. Q. He and P. Lai, "The application of fault tree analysis method in electrical component," Proceedings of the 20th IEEE International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA), Suzhou, 2013, pp. 658-661.

[30] Maosheng, Ding, Wang Gang, and Li Xiaohua. "Reliability analysis of digital relay." (2004): 268-271.

[31] Billinton, Roy, and Ronald Norman Allan. Reliability evaluation of engineering systems. New York: Plenum press, 1992.

[32] Sperandio, Mauricio, and Jorge Coelho. "Constructing Markov models for reliability assessment with self-organizing maps." Probabilistic Methods Applied to Power Systems, 2006. PMAPS 2006. International Conference on. IEEE, 2006.

[33] Buzacott, J. A., and G. J. Anders. "Reliability evaluation of systems with after fault switching." IEEE transactions on power systems 2.3 (1987): 601-607.

[34] Brown, Richard Eric, et al. ”Distribution system reliability assessment using hierarchical Markov modeling.” IEEE Transactions on Power Delivery 11.4 (1996): 1929-1934.

[35] P. Liggesmeyer and M. Rothfelder, ”Improving system reliability with automatic fault tree generation,” Digest of Papers. Twenty-Eighth Annual International Symposium on Fault-Tolerant Computing (Cat. No.98CB36224), Munich, Germany, 1998, pp. 90-99.

[36] Ren, Yansong, and J. Bechta Dugan. ”Design of reliable systems using static and dynamic fault trees.” IEEE Transactions on Reliability 47.3 (1998): 234-244.

[37] W. S. Lee, D. L. Grosh, F. A. Tillman and C. H. Lie, ”Fault Tree Analysis, Methods, and Applications and A Review,” in IEEE Transactions on Reliability, vol. R-34, no. 3, pp. 194-203, Aug. 1985.

[38] L. Kolek, M. Y. Ibrahim, I. Gunawan, M. A. Laribi and S. Zegloul, ”Evaluation of control system reliability using combined dynamic fault trees and Markov models,” 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), Cambridge, 2015, pp. 536-543.

[39] Hou, W., B. Bagen, and A. M. Gole. ”Including reliability index distributions in HVdc system adequacy assessment using Monte Carlo simulation.” (2017): 27-6.

[40] https://engineer.jpl.nasa.gov/practices/at2.pdf

[41] https://www.epsma.org/MTBF/202005.pdf

[42] Hilber, Patrik, and Lina Bertling. ”A method for extracting reliability importance indices from reliability simulations of electrical networks.” 15th Power Systems Computation Conference, PSCC 2005, Liege, Belgium, 22 August 2005 through 26 August 2005. Power Systems Computation Conference (PSCC), 2005

[43] Wang, Wendai, James Loman, and Pantelis Vassiliou. ”Reliability importance of components in a complex system.” Reliability and Maintainability, 2004 Annual Symposium-RAMS. IEEE, 2004.

[44] K. Mikhaylov, .. Juha Petaejaejaervi and T. Haenninen, ”Analysis of Capacity and Scalability of the LoRa Low Power Wide Area Network Technology,” European Wireless 2016; 22th European Wireless Conference, Oulu, Finland, 2016, pp. 1-6.

[45] O. Georgiou and U. Raza, ”Low Power Wide Area Network Analysis: Can LoRa Scale?,” in IEEE Wireless Communications Letters, vol. 6, no. 2, pp. 162-165, April 2017. doi: 10.1109/LWC.2016.2647247

[46] A. J. Wixted, P. Kinnaird, H. Larijani, A. Tait, A. Ahmadinia and N. Strachan, ”Evaluation of LoRa and LoRaWAN for wireless sensor networks,” 2016 IEEE SENSORS, Orlando, FL, 2016, pp. 1-3. doi: 10.1109/ICSENS.2016.7808712

[47] V. C. Gungor et al. 2011. ”Smart Grid Technologies: Communication Technologies and Standards.” IEEE Transactions on Industrial Informatics 529-539.

[48] T. Sauter, M. Lobashov,. 2011. ”End-to-end communication architecture for smart grids.” IEEE Transactions on Industrial Informatics 529-539.

[49] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen and M. Pettissalo,. 2015. ”On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology.” 2015 14th International Conference on ITS Telecommunications (ITST). Copenhagen. 55-59.

[50] n.d. What is LORA? SEMTECH. http://www.semtech.com/wireless-rf/internet-of-things/what-is-lora/.

[51] Corporation, Semtech. 2015. AN1200.22 LoRa Modulation Basics.

[52] Cass, Stephen. 2016. A ham radio for makers. IEEE Spectrum, 21-23

[53] A. Goldsmith Wireless Communications Cambridge University Press 2005

[54] Charles E. Ebeling ”An Introduction to Reliability and Maintainability Engineering”, Waveland Pr, 2009

[55] Cetinkaya E. K., 2001, Reliability Analysis of SCADA Systems Used in Oil and Gas Industry, Retrieved from www.ittc.ku.eduresilinetspapersEKC_MSThesis.pdf

# Appendix A

# Inverter Side Arduino Code

```
#include <Wire.h>

// Cooking API libraries
#include <arduinoUtils.h>

// Include the SX1272 and SPI library:
#include ”arduinoLoRa.h”
#include <SPI.h>

int e;

char message1 [60];

char message_encrypt [60];
char message_recieved [60];
char message_decrypted [60];
char my_packet[60];


void LoRa_transmit();
void encryptt(String key);
void LORAconfig();
void serialreading();
void LoRa_recieve();
void decryptt(String key);

void setup() {
    Serial.begin(9600);
    Serial1.begin(9600);
    Serial2.begin(9600);

 // Print a start message
  Serial.println(F(”SX1272 module and Arduino: receive packets without ACK”));

  // Power ON the module
  e = sx1272.ON();
  Serial.print(F(”Setting power ON: state ”));
  Serial.println(e, DEC);

  // Set transmission mode and print the result
  e |= sx1272.setMode(4);
  Serial.print(F(”Setting Mode: state ”));
  Serial.println(e, DEC);

  // Set header
  e |= sx1272.setHeaderON();
  Serial.print(F(”Setting Header ON: state ”));
  Serial.println(e, DEC);

  // Select frequency channel
  e |= sx1272.setChannel(CH_10_900);
  Serial.print(F(”Setting Channel: state ”));
  Serial.println(e, DEC);
  // Set CRC
  e |= sx1272.setCRC_ON();
  Serial.print(F(”Setting CRC ON: state ”));
  Serial.println(e, DEC);

  // Select output power (Max, High or Low)
  e |= sx1272.setPower(’M’);
  Serial.print(F(”Setting Power: state ”));
  Serial.println(e, DEC);

  // Set the node address and print the result
  e |= sx1272.setNodeAddress(8);
  Serial.print(F(”Setting node address: state ”));
  Serial.println(e, DEC);

  // Print a success message
  if (e == 0)
    Serial.println(F(”SX1272 successfully configured”));
  else
    Serial.println(F(”SX1272 initialization failed”));
}

void loop() {
  String key1 =”l1wRtg6BhjU890crFdeRDsjiKJHbgUhn98NqA3z$c1b&c0mPd2c”;
  String key2 =”n81AvFyJ9liz11BmdFewimNlOpFtrRG67Hrt14eRDS2nbUgLdeFs4Cv5N6W52”;
  // put your main code here, to run repeatedly:
    //if(Serial.available()>0){
  serialreading();
  Serial1.println(message1);
  encryptt(key2);
  LoRa_transmit();
  //  }

  LoRa_recieve();
  decryptt(key1);
  Serial.println(message_decrypted);
  Serial2.println(message_decrypted);

}

void LORAconfig(){
  /*************LoRa setup
**********************/

  // Power ON the module
  e = sx1272.ON();
  Serial.print(F(”Setting power ON: state ”));
  Serial.println(e, DEC);

  // Set transmission mode and print the result
  e |= sx1272.setMode(4);
  Serial.print(F(”Setting Mode: state ”));
  Serial.println(e, DEC);

  // Set header
  e |= sx1272.setHeaderON();
  Serial.print(F(”Setting Header ON: state ”));
  Serial.println(e, DEC);

  // Select frequency channel
  e |= sx1272.setChannel(CH_10_900);
  Serial.print(F(”Setting Channel: state ”));
  Serial.println(e, DEC);

  // Set CRC
  e |= sx1272.setCRC_ON();
  Serial.print(F(”Setting CRC ON: state ”));
  Serial.println(e, DEC);

  // Select output power (Max, High or Low)
  e |= sx1272.setPower(’H’);
  Serial.print(F(”Setting Power: state ”));
  Serial.println(e, DEC);

  // Set the node address and print the result
  e |= sx1272.setNodeAddress(8);
  Serial.print(F(”Setting node address: state ”));
  Serial.println(e, DEC);

  // Print a success message
  if (e == 0)
    Serial.println(F(”SX1272 successfully configured”));
  else
    Serial.println(F(”SX1272 initialization failed”));

  }

void serialreading(){
```

```
int incomingByte = 0;
char x=’a’;

        int i=0;
      //  message1[0] = ””;

//Read serial from the inverter
        while(Serial2.available() > 0) {
                // read the incoming byte:
                incomingByte = Serial2.read();
                x = incomingByte;
                message1[i]= x;
                i++;
        }
  Serial.println(”I got the following messege”);
  Serial.println(message1);
  }

void LoRa_transmit(){
char message2 [60];

    int i =0;
    while(i<60){
    message2[i]= message_encrypt[i];
    i++;
    }
   Serial.print(”i = ”);
   Serial.println(i);
   Serial.println(message2);
   Serial.println(”About to send”);

  e = sx1272.sendPacketTimeout(8, message2);
  Serial.println(”Sent this is the message”);
  Serial.println(message_encrypt);
  Serial.println(”Holly mother of god”);

  Serial.print(F(”Packet sent, state ”));
  Serial.println(e, DEC);
}


void LoRa_recieve(){
    // Receive message
  e = sx1272.receivePacketTimeout(1000);
  if ( e == 0 )
  {
    Serial.print(F(”Receive packet, state ”));
    Serial.println(e, DEC);

    for (unsigned int i = 0; i < sx1272.packet_received.length; i++)
    {
      my_packet[i] = (char)sx1272.packet_received.data[i];
    }
    Serial.print(F(”Message: ”));
    Serial.println(my_packet);
  }
  else {
    Serial.print(F(”Receive packet, state ”));
    Serial.println(e, DEC);
  }
}

void encryptt(String key){
 // message_encrypt =””;
  String cipher=””;
  int message1length = 60;
  int intmessage[message1length];
  int intkey[message1length];
  char x=’A’;

  int i=0;

  Serial.println(”Entered to the encryption loop”);

  while(i<message1length){
    intmessage[i]=message1[i]+21+key[i]-65;
    x=intmessage[i];
    message_encrypt[i] = x;

    i++;
    }
   Serial.print(”i = ”);
   Serial.println(i);
}

void decryptt(String key){
 // message_encrypt =””;
  String cipher=””;
  int message1length = 60;
  int intmessage[message1length];
  int intkey[message1length];
  char x='A';

  int i=0;

  while(i<message1length){
    intmessage[i]=my_packet[i]-21-key[i]+65;
    x=intmessage[i];
    message_decrypted[i] = x;
    i++;
    }
}
```

# Appendix B

# Inverter Side TFT data processing Arduino Code

```
#include <stdio.h>
#include <errno.h>
#include <stdlib.h>
#include <math.h>
#include <Adafruit_GFX.h> // Hardware-specific library
#include <MCUFRIEND_kbv.h>
#include <SPI.h>
#include <SD.h>
#include <EEPROM.h>
#include <avr/io.h>
#include <avr/wdt.h>

MCUFRIEND_kbv tft;

#define BLACK   0x0000
#define BLUE    0x001F
#define RED     0xF800
#define GREEN   0x07E0
#define CYAN    0x07FF
#define MAGENTA 0xF81F
#define YELLOW  0xFFE0
#define WHITE   0xFFFF

const int ok_buttonPin = 40;
int ok_buttonState = 0;

const int left_buttonPin = 41;
int left_buttonState = 0;

const int right_buttonPin = 42;
int right_buttonState = 0;

const int chipSelect = 10;
unsigned long miliseconds=0;
unsigned long miliseconds_pre=0;
long int index_id=0;

char message1 [100] = "";
String sync_word="";
String   status_on = "";
String   status_off = "";
char   current[8] = "";
char   voltage[8] = "";
char   power[8] = "";
char   number_of_samples[4] = "";
char   current_max[4] = "";
char   voltage_max[4] = "";
String   reserved_bytes = "";
```

```
char   version_number[4] = ””;
String   crc_bits = ””;
String date_time_save= ””;

long int status_on_dec = 1;
long int status_off_dec = 1;
long int current_dec = 1;
long int  voltage_dec = 1;
long int power_dec = 1;
long int number_of_samples_dec = 1;
long int current_max_dec = 1;
long int voltage_max_dec = 1;
long int reserved_bytes_dec = 1;
long int version_number_dec =1;
long int crc_bits_dec = 1;

double voltage_actual =0;
double current_actual =0;
float power_actual = 0;

int year_d=2017;
int month_d =7;
int date_d =28;
int hour_d =21;
int minute_d =40;
int second_d =22;

long int dectohex(char x[], int y);
void displayValues(String datavalues);
void sdSave(String dataString);
void sdsetup();
void sdRemove();

void set_clock_diplay_values(int year, int month, int date, int hour, int minute,int second );
void setup_display_clock();
void set_datetime();
void update_time();

void split_number();
void setup_index();

void setup() {
    wdt_disable();
    Serial.begin(9600);
    Serial1.begin(9600);
    tft.reset();
    uint16_t identifier = tft.readID();
    Serial.print(”ID = 0x”);
    Serial.println(identifier, HEX);
    if (identifier == 0xEFEF) identifier = 0x9486;
    tft.begin(identifier);

    // set time
    pinMode(ok_buttonPin, INPUT);
    pinMode(left_buttonPin, INPUT);
    pinMode(right_buttonPin, INPUT);

    setup_index();
    setup_display_clock();
    delay(2000);
    set_datetime();
    delay(1000);

    // setup SD card
    tft.setRotation(0);
    tft.fillScreen(0x0000);
    tft.setCursor(0, 0);
    tft.setTextSize(2);
    tft.println(”Setup Started”);
    delay(1000);
    tft.setCursor(0, 30);
    tft.print(”Initializing SD card...”);
  // see if the card is present and can be initialized:
     delay(1000);
     tft.setCursor(0, 120);

    if (!SD.begin(chipSelect)) {
    tft.println(”Card failed, or not present”);
    // don’t do anything more:
    delay(1000);
    return;
  }
    tft.println(”card initialized.”);
   delay(1000);

}

void loop() {
```

```
String savedataString ="";
char message1 [100] = "";

  // put your main code here, to run repeatedly:
    int incomingByte = 0;
    char x='a';

    int i=0;
    //  message1[0] = "";

    // put your main code here, to run repeatedly:
    while(Serial1.available() > 0) {
                // read the incoming byte:
                incomingByte = Serial1.read();
                x = incomingByte;
                message1[i]= x;
                i++;

    }

    if(i>50){
     status_on = message1[4]+message1[5]+message1[6]+message1[7];
     status_off = message1[8]+message1[9]+message1[10]+message1[11];
      current[0] = message1[18];
      current[1] = message1[19];
      current[2] = message1[16];
      current[3] = message1[17];
      current[4] = message1[14];
      current[5] = message1[15];
      current[6] = message1[12];
      current[7] = message1[13];



      voltage[0] = message1[26];
      voltage[1] = message1[27];
      voltage[2] = message1[24];
      voltage[3] = message1[25];
      voltage[4] = message1[22];
      voltage[5] = message1[23];
      voltage[6] = message1[20];
      voltage[7] = message1[21];

      Serial.println(voltage[0]);
      Serial.println(voltage[1]);
      Serial.println(voltage[2]);
      Serial.println(voltage[3]);
      Serial.println(voltage[4]);
      Serial.println(voltage[5]);
      Serial.println(voltage[6]);
      Serial.println(voltage[7]);
      Serial.println(voltage[8]);
      Serial.println(voltage[9]);

      power[0] = message1[34];
      power[1] = message1[35];
      power[2] = message1[32];
      power[3] = message1[33];
      power[4] = message1[30];
      power[5] = message1[31];
      power[6] = message1[28];
      power[7] = message1[29];


   number_of_samples[0] = message1[38];
   number_of_samples[1] = message1[39];
   number_of_samples[2] = message1[36];
   number_of_samples[3] = message1[37];


   current_max[0] = message1[42];
   current_max[1] = message1[43];
   current_max[2] = message1[40];
   current_max[3] = message1[41];


   voltage_max[0] = message1[46];
   voltage_max[1] = message1[47];
   voltage_max[2] = message1[44];
   voltage_max[3] = message1[45];

   reserved_bytes = message1[50]+message1[51]+message1[48]+message1[49];

   version_number[0] = message1[54];
   version_number[1] = message1[55];
   version_number[2] = message1[52];
   version_number[3] = message1[53];
   voltage_max_dec = dectohex(voltage_max ,4);
   current_max_dec = dectohex(current_max ,4);
   number_of_samples_dec = dectohex(number_of_samples ,4);
   version_number_dec= dectohex(version_number ,4);

   crc_bits = message1[56]+message1[57]+message1[58]+message1[59];

   Serial.println(current);
   current_dec = dectohex(current ,8);
   Serial.println(current_dec);

   Serial.println(voltage);
   voltage_dec = dectohex(voltage ,8);
   Serial.println(voltage_dec);

   Serial.println(”power”);
   power_dec = dectohex(power ,8);
   Serial.println(power_dec);

   Serial.println(”version number”);
   Serial.println(version_number);
   Serial.println(version_number_dec);

   voltage_actual = (sqrt(voltage_dec/number_of_samples_dec)*voltage_max_dec)/4096;
   current_actual = (sqrt(current_dec/number_of_samples_dec)*current_max_dec)/4096;

  Serial.println(power_dec);
  power_actual = power_dec/(4096);
  Serial.println(power_actual);

  power_actual = (power_actual*current_max_dec)/number_of_samples_dec;
  Serial.println(power_actual);

  power_actual = (power_actual*voltage_max_dec)/4096;
  Serial.println(power_actual);

  long int dumpvalue =0;
  dumpvalue = number_of_samples_dec*4096;
  long double dumpvalue1= 0;

  Serial.println( power_actual);
  Serial.println(power_dec);
  Serial.println(dumpvalue);
  Serial.println(voltage_max_dec);
  Serial.println(current_max_dec);
  Serial.println(number_of_samples_dec);

 Serial.println(message1);
 update_time();
 date_time_save = String(year_d)+”-” + String(month_d) +”-” + String(date_d) +”  ” ;
 date_time_save = date_time_save + String(hour_d) +”:” + String(minute_d) +”:” + String(second_d);

 index_id++;
 savedataString = date_time_save+”,”+ String(index_id)+”,”+ String(current_dec)+ ”,” +String(voltage_dec);
 savedataString = savedataString + ”,” + String(power_dec)+ ”,” + String(version_number_dec);
 sdSave( savedataString);

 split_number();
 displayValues(”valid”);
 delay(5000);

    }else{
         Serial.println(message1);
         displayValues(”not valid”);
         update_time();
         split_number();
         delay(500);
        }
    sdRemove();
}


long int dectohex(char x[], int y){
   long int dec_number =0;
   int z[y];
   Serial.println(”entered to hex to dec”);

  for(int i=0; i<y; i++){
    if(x[i]==’0’){
      z[i]=0;
      }else if(x[i]==’1’){
          z[i]=1;
        }else if(x[i]==’2’){
              z[i]=2;
        }else if(x[i]==’3’){
              z[i]=3;
        }else if(x[i]==’4’){
              z[i]=4;
        }else if(x[i]==’5’){
              z[i]=5;
        }else if(x[i]==’6’){
              z[i]=6;
        }else if(x[i]==’7’){
              z[i]=7;
        }else if(x[i]==’8’){
              z[i]=8;
        }else if(x[i]==’9’){
              z[i]=9;
        }else if(x[i]==’a’){
              z[i]=10;
        }else if(x[i]==’b’){
              z[i]=11;
        }else if(x[i]==’c’){
              z[i]=12;
        }else if(x[i]==’d’){
              z[i]=13;
        }else if(x[i]==’e’){
              z[i]=14;
        }else if(x[i]==’f’){
              z[i]=15;
        }

  long  int result = pow(16, y-1-i)+1;

       if(result >2){
        dec_number=dec_number+result*z[i];
       }else {
        dec_number=dec_number+z[i];
        }
    }

  Serial.println(dec_number);
  return dec_number;
  }


void displayValues(String datavalues){
    if(datavalues==”valid”){
    tft.setRotation(0);
    tft.fillScreen(0x0000);
    tft.setCursor(0, 0);
    tft.setTextSize(2);
    tft.println(”Inverter SCADA”);

    tft.setCursor(0, 30);
    tft.println(”ON / OFF”);
    tft.setCursor(0, 75);
    tft.println(”Current:”);
    tft.setCursor(30, 100);
    tft.println(current_actual);
    tft.setCursor(0, 150);
    tft.println(”Voltage:”);
    tft.setCursor(30, 175);
    tft.println(voltage_actual);
    tft.setCursor(0, 225);
    tft.println(”Power:”);
    tft.setCursor(30, 250);
    tft.println(power_actual);
    tft.setCursor(0, 280);
    tft.setTextSize(1);
    tft.println(”Inverter Version ”);
    tft.setCursor(30, 290);
    tft.println(version_number_dec);
    tft.setCursor(0, 300);
    tft.println(date_time_save);
      }else{

    tft.setRotation(0);
    tft.fillScreen(0x0000);
    tft.setCursor(0, 150);
    tft.setTextSize(2);
    tft.println(”Not a valid string”);
    update_time();
    date_time_save = String(year_d)+”-” + String(month_d) +”-” + String(date_d) +”  ” ;
    date_time_save=date_time_save + String(hour_d) +”:” + String(minute_d) +”:” + String(second_d);
    tft.setCursor(0, 250);
    tft.println(date_time_save);
    split_number();
        }
}

void sdSave(String dataString){
     long int data_per_day = 17280;
    String lineone =”created_at,entry_id,field1,field2,field3,field4”;
    File myFile;
```

```
if(index_id ==0){
      if (SD.exists(”1.CSV”)) {
          Serial.println(”1.CSV exists.”);
          SD.remove(”1.CSV”);
        }
      myFile = SD.open(”1.CSV”, FILE_WRITE);
      myFile.close();
      File dataFile = SD.open(”1.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(lineone);
        dataFile.println(dataString);
        dataFile.close();
      }

  }else if(index_id>0 && index_id< data_per_day){
      File dataFile = SD.open(”1.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(dataString);
        dataFile.close();
      }
  }else if(index_id == data_per_day){

      if (SD.exists(”2.CSV”)) {
          Serial.println(”2.CSV exists.”);
          SD.remove(”2.CSV”);
        }
      myFile = SD.open(”2.CSV”, FILE_WRITE);
      myFile.close();
      File dataFile = SD.open(”2.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(lineone);
        dataFile.println(dataString);
        dataFile.close();
      }


  }else if(index_id>data_per_day && index_id< 2*data_per_day){
      File dataFile = SD.open(”2.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(dataString);
        dataFile.close();
      }

  }else if(index_id == 2*data_per_day){
      if (SD.exists(”3.CSV”)) {
          Serial.println(”3.CSV exists.”);
          SD.remove(”3.CSV”);
        }
      myFile = SD.open(”3.CSV”, FILE_WRITE);
      myFile.close();
      File dataFile = SD.open(”3.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(lineone);
        dataFile.println(dataString);
        dataFile.close();
      }

  }else if(index_id >2*data_per_day && index_id< 3*data_per_day){
      File dataFile = SD.open(”3.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(dataString);
        dataFile.close();
      }

  }else if(index_id == 3*data_per_day){
      if (SD.exists(”4.CSV”)) {
          Serial.println(”4.CSV exists.”);
          SD.remove(”4.CSV”);
        }
      myFile = SD.open(”4.CSV”, FILE_WRITE);
      myFile.close();
      File dataFile = SD.open(”4.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(lineone);
        dataFile.println(dataString);
        dataFile.close();
      }
    }else if(index_id >3*data_per_day && index_id< 4*data_per_day){
      File dataFile = SD.open(”4.CSV”, FILE_WRITE);
      if (dataFile) {
        dataFile.println(dataString);
        dataFile.close();
      }
    }else if(index_id == 4*data_per_day){
      if (SD.exists(”5.CSV”)) {
          Serial.println(”5.CSV exists.”);
          SD.remove(”5.CSV”);
        }
        myFile = SD.open(”5.CSV”, FILE_WRITE);
        myFile.close();
        File dataFile = SD.open(”5.CSV”, FILE_WRITE);
        if (dataFile) {
          dataFile.println(lineone);
          dataFile.println(dataString);
          dataFile.close();
        }
       }else if(index_id >4*data_per_day && index_id< 5*data_per_day){
        File dataFile = SD.open(”5.CSV”, FILE_WRITE);
        if (dataFile) {
          dataFile.println(dataString);
          dataFile.close();
        }
    }else if(index_id == 5*data_per_day){
        if (SD.exists(”6.CSV”)) {
            Serial.println(”6.CSV exists.”);
           SD.remove(”6.CSV”);
          }
        myFile = SD.open(”6.CSV”, FILE_WRITE);
        myFile.close();
        File dataFile = SD.open(”6.CSV”, FILE_WRITE);
        if (dataFile) {
          dataFile.println(lineone);
          dataFile.println(dataString);
          dataFile.close();
        }
    }else if(index_id >5*data_per_day && index_id< 6*data_per_day){

        File dataFile = SD.open(”6.CSV”, FILE_WRITE);
        if (dataFile) {
          dataFile.println(dataString);
          dataFile.close();
        }
    }else if(index_id == 6*data_per_day){

        if (SD.exists(”7.CSV”)) {
            Serial.println(”7.CSV exists.”);
           SD.remove(”7.CSV”);
          }
        myFile = SD.open(”7.CSV”, FILE_WRITE);
        myFile.close();
        File dataFile = SD.open(”7.CSV”, FILE_WRITE);
        if (dataFile) {
          dataFile.println(lineone);
          dataFile.println(dataString);
          dataFile.close();
        }
    }else if(index_id >6*data_per_day && index_id< 7*data_per_day){
        File dataFile = SD.open(”7.CSV”, FILE_WRITE);
        if (dataFile) {
          dataFile.println(dataString);
          dataFile.close();
        }

     } else if(index_id >7*data_per_day ){
      index_id =0;
      }
}

void sdsetup(){
    tft.setRotation(0);
    tft.fillScreen(0x0000);
    tft.setCursor(0, 60);
    tft.print(”Initializing SD card...”);
  // see if the card is present and can be initialized:
  delay(1000);
      tft.setCursor(0, 120);
  if (!SD.begin(chipSelect)) {
    tft.println(”Card failed, or not present”);
    delay(5000);
    // don’t do anything more:
    return;
  }
    tft.println(”card initialized.”);

  delay(1000);


  }

void sdRemove(){
    ok_buttonState = digitalRead(ok_buttonPin);
      if (ok_buttonState == HIGH) {
              tft.setRotation(0);
              tft.fillScreen(0x0000);
              tft.setCursor(0, 30);
              tft.setTextSize(2);
              tft.println(”Safe to remove the SD card”);
              //  tft.setTextSize(2);
              tft.setCursor(0, 60);
              tft.println(”Press the button to continue”);
              delay(2000);
                  ok_buttonState = digitalRead(ok_buttonPin);
                  while(ok_buttonState != HIGH)
                    {
                      ok_buttonState = digitalRead(ok_buttonPin);
                      }
              wdt_enable(WDTO_30MS);
              while(1) {};
             // sdsetup();
              return;
          }

  }

void setup_display_clock(){
    tft.setRotation(0);
    tft.fillScreen(0x0000);
    tft.setTextSize(2);
    tft.fillRect(30, 0, 168, 18, BLUE);

    tft.setCursor(30, 0);
    tft.println(”Inverter SCADA”);

    tft.setCursor(0, 30);
    tft.println(”Set Date and Time”);

    tft.setCursor(10, 60);
    tft.println(”Year”);

    tft.setCursor(10, 90);
    tft.println(”Month”);

    tft.setCursor(10, 120);
    tft.println(”Date”);

    tft.setCursor(10, 150);
    tft.println(”HH”);

    tft.setCursor(10, 180);
    tft.println(”MM”);

    tft.setCursor(10, 210);
    tft.println(”SS”);

  }


void set_clock_diplay_values(int year, int month, int date, int hour, int minute,int second ){

    tft.fillRect(100, 60, 60, 18, BLUE);
    tft.setCursor(100, 60);
    tft.println(year);

    tft.fillRect(100, 90, 60, 18, BLUE);
    tft.setCursor(100, 90);
    tft.println(month);

    tft.fillRect(100, 120, 60, 18, BLUE);
    tft.setCursor(100, 120);
    tft.println(date);

    tft.fillRect(100, 150, 60, 18, BLUE);
    tft.setCursor(100, 150);
    tft.println(hour);

    tft.fillRect(100, 180, 60, 18, BLUE);
    tft.setCursor(100, 180);
    tft.println(minute);

    tft.fillRect(100, 210, 60, 18, BLUE);
    tft.setCursor(100, 210);
    tft.println(second);

    tft.fillRect(0, 270, 300, 18, GREEN);
    tft.setCursor(0, 270);
    tft.println(String(year)+”/” + String(month) +”/” + String(date) +”    ” + String(hour)
    +”:” + String(minute) +”:” + String(second));
}

void set_datetime(){
    int verticle=1;
    int up =1;
    int down=1;
```

```
   int counter =0;
   ok_buttonState = digitalRead(ok_buttonPin);
   tft.setTextSize(2);
   while(ok_buttonState == LOW){
                tft.fillRect(0, 300, 300, 18, GREEN);
                tft.setCursor(0, 300);
                tft.println(”Press Ok to set year”);
                delay(500);
                ok_buttonState = digitalRead(ok_buttonPin);
                if(counter>10){
                 return;
                 }
                counter++;

     }
             delay(1000);
                ok_buttonState = digitalRead(ok_buttonPin);

         if(verticle==1){
              while(ok_buttonState == LOW){
              tft.fillRect(0, 30+30*(verticle+5), 10, 18, BLACK);
              tft.fillRect(0, 30+30*verticle, 5, 18, GREEN);
//              tft.setCursor(2, 30+30*verticle);
              tft.println(”|”);
              ok_buttonState = digitalRead(ok_buttonPin);
               while(ok_buttonState == LOW){
                 left_buttonState = digitalRead(left_buttonPin);
                 right_buttonState = digitalRead(right_buttonPin);
                     if (left_buttonState == HIGH) {
                       year_d=year_d+1;
                       }
                     if (right_buttonState == HIGH) {
                       year_d=year_d-1;
                       }

                tft.fillRect(0, 300, 300, 18, GREEN);
                tft.setCursor(0, 300);
                tft.println(”Press Ok to set month”);
                ok_buttonState = digitalRead(ok_buttonPin);
                set_clock_diplay_values(year_d,month_d,date_d,hour_d,minute_d,second_d);
              }
            }
           delay(1000);
           ok_buttonState = digitalRead(ok_buttonPin);
           verticle++;
           }
         if(verticle==2){
           while(ok_buttonState == LOW){
              tft.fillRect(0, 30+30*(verticle-1), 10, 18, BLACK);
              tft.fillRect(0, 30+30*verticle, 5, 18, GREEN);
              tft.setCursor(2, 30+30*verticle);
              ok_buttonState = digitalRead(ok_buttonPin);
               while(ok_buttonState == LOW){
                 left_buttonState = digitalRead(left_buttonPin);
                 right_buttonState = digitalRead(right_buttonPin);
                     if (left_buttonState == HIGH) {
                       month_d=month_d+1;
                       }
                     if (right_buttonState == HIGH) {
                       month_d=month_d-1;
                       }
                tft.fillRect(0, 300, 300, 18, GREEN);
                tft.setCursor(0, 300);
                tft.println(”Press Ok to set date”);
                ok_buttonState = digitalRead(ok_buttonPin);
                set_clock_diplay_values(year_d,month_d,date_d,hour_d,minute_d,second_d);
              }
           }
           delay(1000);
           ok_buttonState = digitalRead(ok_buttonPin);
           verticle++;

         }

         if(verticle==3){
           while(ok_buttonState == LOW){
              tft.fillRect(0, 30+30*(verticle-1), 10, 18, BLACK);
              tft.fillRect(0, 30+30*verticle, 5, 18, GREEN);
              tft.setCursor(2, 30+30*verticle);
      //        tft.println(”|”);
              ok_buttonState = digitalRead(ok_buttonPin);
               while(ok_buttonState == LOW){
                 left_buttonState = digitalRead(left_buttonPin);
                 right_buttonState = digitalRead(right_buttonPin);
                     if (left_buttonState == HIGH) {
                       date_d=date_d+1;
                       }
                     if (right_buttonState == HIGH) {
                       date_d=date_d -1;
                       }
                tft.fillRect(0, 300, 300, 18, GREEN);
                tft.setCursor(0, 300);
                tft.println(”Press Ok to set hour”);
                ok_buttonState = digitalRead(ok_buttonPin);
                set_clock_diplay_values(year_d ,month_d ,date_d ,hour_d ,minute_d ,second_d );
              }
           }
           delay(1000);
           ok_buttonState = digitalRead(ok_buttonPin);
           verticle++;

           }
         if(verticle==4){
            while(ok_buttonState == LOW){
              tft.fillRect(0, 30+30*(verticle -1), 10, 18, BLACK);
              tft.fillRect(0, 30+30*verticle , 5, 18, GREEN);
              tft.setCursor(2, 30+30*verticle );
              ok_buttonState = digitalRead(ok_buttonPin);
               while(ok_buttonState == LOW){
                 left_buttonState = digitalRead(left_buttonPin);
                 right_buttonState = digitalRead(right_buttonPin);
                     if (left_buttonState == HIGH) {
                       hour_d=hour_d+1;
                       }
                     if (right_buttonState == HIGH) {
                       hour_d=hour_d -1;
                       }
                tft.fillRect(0, 300, 300, 18, GREEN);
                tft.setCursor(0, 300);
                tft.println(”Press Ok to set minute”);
                ok_buttonState = digitalRead(ok_buttonPin);
                set_clock_diplay_values(year_d ,month_d ,date_d ,hour_d ,minute_d ,second_d );
              }
            }
           delay(1000);
           ok_buttonState = digitalRead(ok_buttonPin);
           verticle++;
         }
         if(verticle==5){
           while(ok_buttonState == LOW){
              tft.fillRect(0, 30+30*(verticle -1), 10, 18, BLACK);
              tft.fillRect(0, 30+30*verticle , 5, 18, GREEN);
              tft.setCursor(2, 30+30*verticle );
              ok_buttonState = digitalRead(ok_buttonPin);
               while(ok_buttonState == LOW){
                 left_buttonState = digitalRead(left_buttonPin);
                 right_buttonState = digitalRead(right_buttonPin);
                     if (left_buttonState == HIGH) {
                       minute_d=minute_d+1;
                       }
                     if (right_buttonState == HIGH) {
                       minute_d=minute_d -1;
                       }
                tft.fillRect(0, 300, 300, 18, GREEN);
                tft.setCursor(0, 300);
                tft.println(”Press Ok to set minute”);
                ok_buttonState = digitalRead(ok_buttonPin);
                set_clock_diplay_values(year_d ,month_d ,date_d ,hour_d ,minute_d ,second_d );
              }
           }
           delay(1000);
           ok_buttonState = digitalRead(ok_buttonPin);
           verticle++;

           }
         if(verticle==6){
           while(ok_buttonState == LOW){
              tft.fillRect(0, 30+30*(verticle -1), 10, 18, BLACK);
              tft.fillRect(0, 30+30*verticle , 5, 18, GREEN);
              tft.setCursor(2, 30+30*verticle );
              ok_buttonState = digitalRead(ok_buttonPin);
               while(ok_buttonState == LOW){
                 left_buttonState = digitalRead(left_buttonPin);
                 right_buttonState = digitalRead(right_buttonPin);
                     if (left_buttonState == HIGH) {
                       second_d=second_d+1;
                       }
                     if (right_buttonState == HIGH) {
                       second_d=second_d -1;
                       }
                tft.fillRect(0, 300, 300, 18, GREEN);
                tft.setCursor(0, 300);
                tft.println(”Press Ok to set minute”);
                ok_buttonState = digitalRead(ok_buttonPin);
                set_clock_diplay_values(year_d ,month_d ,date_d ,hour_d ,minute_d ,second_d );
```

```
               }
            }
            delay(1000);
            ok_buttonState = digitalRead(ok_buttonPin);
            verticle++;
            }
          if(verticle==7){
                 tft.fillRect(0, 30+30*(verticle-1), 10, 18, BLACK);
                 tft.fillRect(0, 300, 300, 25, BLACK);
                 tft.setCursor(0, 300);
                 tft.setTextSize(1);
                 tft.println(”Press Ok exit or anyother button to set time again”);
                 ok_buttonState = digitalRead(ok_buttonPin);
                     while(ok_buttonState == LOW){
                            left_buttonState = digitalRead(left_buttonPin);
                            right_buttonState = digitalRead(right_buttonPin);
                                if (left_buttonState == HIGH) {
                                  set_datetime();
                                  }
                                if (right_buttonState == HIGH) {
                                  set_datetime();
                                  }
                                verticle==0;
                               ok_buttonState = digitalRead(ok_buttonPin);
                         }
            }
                 tft.fillRect(0, 300, 300, 25, BLACK);
                 tft.setTextSize(2);

  }

void update_time(){

int year_u=2017;
int year_pre=2017;

int month_u =6;
int month_pre =6;

int date_u =2;
int date_pre =2;

int hour_u =21;
int hour_pre =21;

int minute_u =14;
int minute_pre =14;

unsigned long second_u =22;
unsigned long second_pre =22;

   second_u =  (millis() -miliseconds );
   if(second_u<0){
    miliseconds=0;
    second_u =  (millis() -miliseconds );
    }

   miliseconds = millis();
   Serial.print(”SS u: ”);
   Serial.println(second_u);
   miliseconds_pre =  second_u + miliseconds_pre;

   second_d = miliseconds_pre/1000;
   Serial.print(”SS: ”);
   Serial.println(second_d);

   if(second_d>59){
      minute_u = second_d/60;
      second_d =second_d - minute_u*60;
      miliseconds_pre = second_d*1000;
      minute_d = minute_u+minute_d;
    }
      if(minute_d>59){
        minute_d= minute_d- 60;
        hour_d = hour_d+1;
    }
       if(hour_d>23){
        hour_d = hour_d-24;
        date_d = date_d +1;
    }
      if(date_d==28&&month_d==2){
          year_pre = year_d%4;
          if(year_pre ==0){
            }else{
              month_d ++;
              date_d = 1;
              }
          }else if(date_d==29&&month_d==2){
             month_d ++;
             date_d = 1;

            }else if(date_d ==30){
              if (month_d==4||month_d==6||month_d==9||month_d==11){
            month_d ++;
            date_d = 1;
            }}else if(date_d ==31){
                month_d ++;
                date_d = 1;
              }
      if(month_d==12){
        month_d = 1;
        year_d++;
        }
}

void split_number(){

  int x1=0;
  int x2=0;
  int x3=0;
  int x4=0;
  int x5=0;
  int x6=0;

  int y1=0;
  int y2=0;
  int y3=0;
  int y4=0;

  int m1=0;
  int m2=0;

  int d1=0;
  int d2=0;

  int hh1=0;
  int hh2=0;

  int mm1=0;
  int mm2=0;

  int ss1=0;
  int ss2=0;

// save index id to EEPROM

  x1 = (index_id/100000)%10;
  x2 = (index_id/10000)%10;
  x3 = (index_id/1000)%10;
  x4 = (index_id/100)%10;
  x5 = (index_id/10)%10;
  x6 = (index_id)%10;

   EEPROM.write(0,x1);
   EEPROM.write(1,x2);
   EEPROM.write(2,x3);
   EEPROM.write(3,x4);
   EEPROM.write(4,x5);
   EEPROM.write(5,x6);


// Save year to EEPROM
  y1 = (year_d/1000)%10;
  y2 = (year_d/100)%10;
  y3 = (year_d/10)%10;
  y4 = (year_d)%10;

   EEPROM.write(6,y1);
   EEPROM.write(7,y2);
   EEPROM.write(8,y3);
   EEPROM.write(9,y4);

// save month to EEPROM
  m1 = (month_d/10)%10;
  m2 = (month_d)%10;

   EEPROM.write(10,m1);
   EEPROM.write(11,m2);

// Save date to EEPROM
  d1 = (date_d/10)%10;
  d2 = (date_d)%10;

   EEPROM.write(12,d1);
   EEPROM.write(13,d2);
```

```
// Save hour to EEPROM
  hh1 = (hour_d/10)%10;
  hh2 = (hour_d)%10;

   EEPROM.write(14,hh1);
   EEPROM.write(15,hh2);

// Save minute to EEPROM
  mm1 = (minute_d/10)%10;
  mm2 = (minute_d)%10;

   EEPROM.write(16,mm1);
   EEPROM.write(17,mm2);

// Save second to EEPROM
  ss1 = (second_d/10)%10;
  ss2 = (second_d)%10;

   EEPROM.write(18,ss1);
   EEPROM.write(19,ss2);

}

void setup_index(){

  int x1=0;
  int x2=0;
  int x3=0;
  int x4=0;
  int x5=0;
  int x6=0;

  int y1=0;
  int y2=0;
  int y3=0;
  int y4=0;

  int m1=0;
  int m2=0;

  int d1=0;
  int d2=0;

  int hh1=0;
  int hh2=0;

  int mm1=0;
  int mm2=0;

  int ss1=0;
  int ss2=0;

// read index id to EEPROM
   x1 =EEPROM.read(0);
   x2 =EEPROM.read(1);
   x3 =EEPROM.read(2);
   x4 =EEPROM.read(3);
   x5 =EEPROM.read(4);
   x6 =EEPROM.read(5);

  index_id = x1*100000+x2*10000+x3*1000+x4*100+x5*10+x6;

// raed year to EEPROM

   y1 =EEPROM.read(6);
   y2 =EEPROM.read(7);
   y3 =EEPROM.read(8);
   y4 =EEPROM.read(9);

  year_d = y1*1000+y2*100+y3*10+y4;

// save month to EEPROM

   m1 =EEPROM.read(10);
   m2 =EEPROM.read(11);

  month_d = m1*10+m2;

// Save date to EEPROM
   d1 =EEPROM.read(12);
   d2 =EEPROM.read(13);

  date_d = d1*10+d2;

// Save hour to EEPROM
   hh1 =EEPROM.read(14);
   hh2 =EEPROM.read(15);
  hour_d = hh1*10+hh2;

// Save minute to EEPROM
   mm1 =EEPROM.read(16);
   mm2 =EEPROM.read(17);

   minute_d = mm1*10+mm2;

// Save second to EEPROM
   ss1 =EEPROM.read(18);
   ss2 =EEPROM.read(19);

   second_d = ss1*10+ss2;

  }
```

# Appendix C

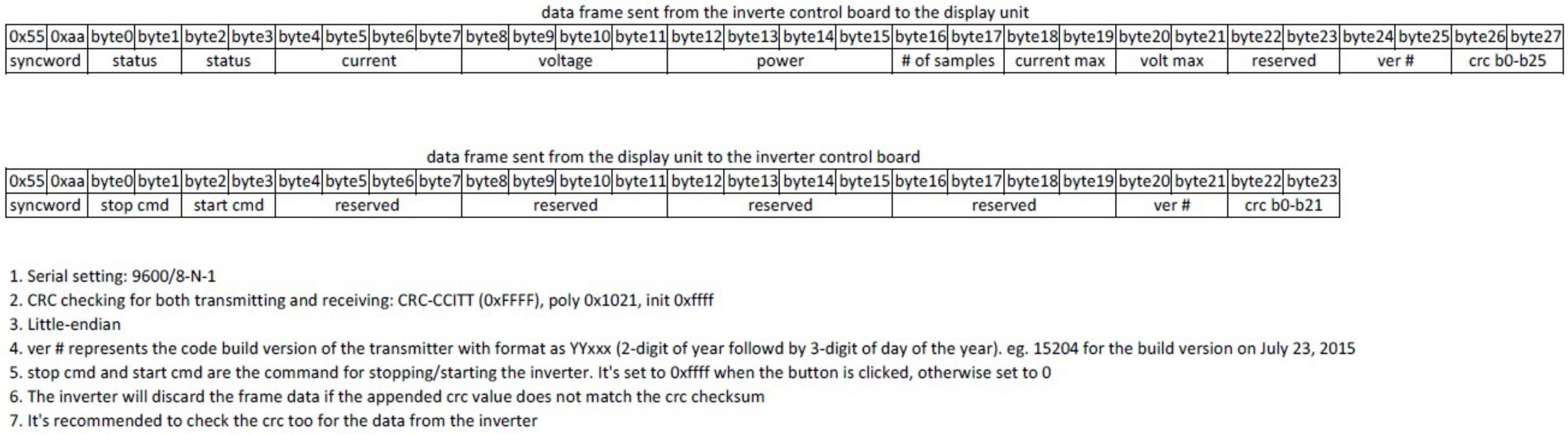
# Server Side Arduino Code

```
#include <Wire.h>

// Cooking API libraries
#include <arduinoUtils.h>

// Include the SX1272 and SPI library:
#include "arduinoLoRa.h"
#include <SPI.h>

int e;
char message1 [60];
char message_encrypt [60];
char message_recieved [60];
char message_decrypted [60];
char my_packet[60];


void LoRa_transmit();
void encryptt(String key);
void LORAconfig();
void serialreading();
void LoRa_recieve();
void decryptt(String key);

void setup() {
  // put your setup code here, to run once:
    Serial.begin(9600);
  // Print a start message
  Serial.println(F("SX1272 module and Arduino: receive packets without ACK"));

  // Power ON the module
  e = sx1272.ON();
  Serial.print(F("Setting power ON: state "));
  Serial.println(e, DEC);

  // Set transmission mode and print the result
  e |= sx1272.setMode(4);
  Serial.print(F("Setting Mode: state "));
  Serial.println(e, DEC);

  // Set header
  e |= sx1272.setHeaderON();
  Serial.print(F("Setting Header ON: state "));
  Serial.println(e, DEC);

  // Select frequency channel
  e |= sx1272.setChannel(CH_10_900);
  Serial.print(F("Setting Channel: state "));
  Serial.println(e, DEC);

  // Set CRC
  e |= sx1272.setCRC_ON();
  Serial.print(F("Setting CRC ON: state "));
  Serial.println(e, DEC);
  // Select output power (Max, High or Low)
  e |= sx1272.setPower('M');
  Serial.print(F("Setting Power: state "));
  Serial.println(e, DEC);

  // Set the node address and print the result
  e |= sx1272.setNodeAddress(8);
  Serial.print(F("Setting node address: state "));
  Serial.println(e, DEC);

  // Print a success message
  if (e == 0)
    Serial.println(F("SX1272 successfully configured"));
  else
    Serial.println(F("SX1272 initialization failed"));
}

void loop() {
  String key1 ="l1wRtg6BhjU890crFdeRDsjiKJHbgUhn98NqA3z$c1b&c0mPd2c";
  String key2 ="n81AvFyJ9liz11BmdFewimNlOpFtrRG67Hrt14eRDS2nbUgLdeFs4Cv5N6W5";
  // put your main code here, to run repeatedly:

  LoRa_recieve();
  //Serial.println(message_decrypted);
  decryptt(key2);
  if(e!=1){
  Serial.println("Recileved messege after decryption");
  Serial.println(message_decrypted);
  } else{
      Serial.println("Not recived a packet");

    }

  if(Serial.available()>0){
  serialreading();
  encryptt(key1);
  LoRa_transmit();
    }
}



void LORAconfig(){
/*********************************************************
LoRa setup
*********************************************************/


  // Print a start message
  Serial.println(F("SX1272 module and Arduino: send packets without ACK"));

  // Power ON the module
  e = sx1272.ON();
  Serial.print(F("Setting power ON: state "));
  Serial.println(e, DEC);

  // Set transmission mode and print the result
  e |= sx1272.setMode(4);
  Serial.print(F("Setting Mode: state "));
  Serial.println(e, DEC);

  // Set header
  e |= sx1272.setHeaderON();
  Serial.print(F("Setting Header ON: state "));
  Serial.println(e, DEC);

  // Select frequency channel
  e |= sx1272.setChannel(CH_10_900);
  Serial.print(F("Setting Channel: state "));
  Serial.println(e, DEC);

  // Set CRC
  e |= sx1272.setCRC_ON();
  Serial.print(F("Setting CRC ON: state "));
  Serial.println(e, DEC);

  // Select output power (Max, High or Low)
  e |= sx1272.setPower('H');
  Serial.print(F("Setting Power: state "));
  Serial.println(e, DEC);

  // Set the node address and print the result
  e |= sx1272.setNodeAddress(3);
  Serial.print(F("Setting node address: state "));
  Serial.println(e, DEC);

  // Print a success message
  if (e == 0)
    Serial.println(F(”SX1272 successfully configured”));
  else
    Serial.println(F(”SX1272 initialization failed”));
}

void serialreading(){
int incomingByte = 0;
char x='a';


        int i=0;
      //  message1[0] = ””;

//Read serial from the inverter
        while(Serial.available() > 0) {
                // read the incoming byte:
                incomingByte = Serial.read();
                x = incomingByte;
                message1[i]= x;
                i++;
        }
}



void LoRa_transmit(){
  // Send message1 and print the result
  e = sx1272.sendPacketTimeout(8, message_encrypt);
  Serial.print(F(”Packet sent, state ”));
  Serial.println(e, DEC);
}


void LoRa_recieve(){
    // Receive message
  e = sx1272.receivePacketTimeout(10000);
  if ( e == 0 )
  {
    Serial.print(F(”Receive packet, state ”));
    Serial.println(e, DEC);

    for (unsigned int i = 0; i < sx1272.packet_received.length; i++)
    {
      my_packet[i] = (char)sx1272.packet_received.data[i];
    }
    Serial.print(F(”Message: ”));
    Serial.println(my_packet);
  }
  else {
    Serial.print(F(”Receive packet, state ”));
    Serial.println(e, DEC);
  }
}

void encryptt(String key){
 // message_encrypt =””;
  String cipher=””;
  int message1length = 60;
  int intmessage[message1length];
  int intkey[message1length];
  char x='A';

  int i=0;

  while(i<message1length){
    intmessage[i]=message1[i]+21+key[i]-65;
    x=intmessage[i];
    message_encrypt[i] = x;
    i++;
    }
}

void decryptt(String key){
 // message_encrypt =””;
  String cipher=””;
  int message1length = 60;
  int intmessage[message1length];
  int intkey[message1length];
  char x='A';

  int i=0;

  while(i<message1length){
    intmessage[i]=my_packet[i]-21-key[i]+65;
    x=intmessage[i];
    message_decrypted[i] = x;
    i++;
```

```
        }
}
```

# Appendix D

# Protocol



data frame sent from the inverte control board to the display unit

| 0x55 | 0xaa | byte0 | byte1 | byte2 | byte3 | byte4 | byte5 | byte6 | byte7 | byte8 | byte9 | byte10 | byte11 | byte12 | byte13 | byte14 | byte15 | byte16 | byte17 | byte18 | byte19 | byte20 | byte21 | byte22 | byte23 | byte24 | byte25 | byte26 | byte27 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| syncword | | status | | status | | current | | | | voltage | | | | power | | | | # of samples | | current max | | volt max | | reserved | | ver # | | crc b0-b25 | |

data frame sent from the display unit to the inverter control board

| 0x55 | 0xaa | byte0 | byte1 | byte2 | byte3 | byte4 | byte5 | byte6 | byte7 | byte8 | byte9 | byte10 | byte11 | byte12 | byte13 | byte14 | byte15 | byte16 | byte17 | byte18 | byte19 | byte20 | byte21 | byte22 | byte23 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| syncword | | stop cmd | | start cmd | | reserved | | | | reserved | | | | reserved | | | | reserved | | | | ver # | | crc b0-b21 | |

1. Serial setting: 9600/8-N-1
2. CRC checking for both transmitting and receiving: CRC-CCITT (0xFFFF), poly 0x1021, init 0xffff
3. Little-endian
4. ver # represents the code build version of the transmitter with format as YYxxx (2-digit of year followd by 3-digit of day of the year). eg. 15204 for the build version on July 23, 2015
5. stop cmd and start cmd are the command for stopping/starting the inverter. It's set to 0xffff when the button is clicked, otherwise set to 0
6. The inverter will discard the frame data if the appended crc value does not match the crc checksum
7. It's recommended to check the crc too for the data from the inverter