A METERING NETWORK FOR A SMART PREPAID SCHEME UNDER AN ELECTRICAL UTILITY ORGANIZATION

By

Md. Ashrafur Rahman
13321010
Tridib Roy
12121047
Syeda Afra Saiara
16321165
Sheikh Tajnur Mubassir
13321024

A Thesis Submitted to the Department of Electrical and Electronic Engineering of BRAC University in Partial Fulfillment of the Requirement of the Degree of Bachelor of Science in Electrical and Electronic Engineering

Department of Electrical and Electronic Engineering

BRAC University
October 2020

Declaration

It is hereby declared that

1. The thesis submitted is our original work while completing a degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all of the main sources of help.

Student's Full Name & Signature:

MD ASHRAFUR RAHMAN	**TRIDIB ROY**
13321010	12121047
SYEDA AFRA SAIARA	**SHEIKH TAJNUR MUBASSIR**
16321165	13321024

Approval

The thesis/project titled “[Thesis/Project Title]” submitted by

1. Md. Ashrafur Rahman (13321010)
2. Tridib Roy (12121047)
3. Syeda Afra Saiara(16321165)
4. Sheikh Tajnur Mubassir(13321024)

of Summer, 2020 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Department of Electrical and Electronic Engineering on 07-09-2020.

Examining Committee:

Supervisor:
(Member)

____

Dr. S. M. Lutful Kabir
Professor,IICT
BUET

Program Coordinator:
(Member)

____

Dr.Abu S.M. Mohsin
Assistant Professor,EEE
Brac University

Departmental Head:
(Chair)

____

Dr. Shahidul Islam Khan
Chairperson & Professor, EEE
BRAC University

Abstract

The present prepaid metering scheme is connected to server mainly using the GSM network. It creates an inherent dependence on mobile companies. Moreover, there is always an operational (recurring) cost to maintain different kinds of information exchange. Therefore, it will be beneficial for utility organizations if they could develop a low-cost network of their own.

The objective proposed thesis work is mainly to develop a backbone metering network which once installed can be used for information exchange almost free of recurring cost. The establishment cost of the network may be high, but the benefit can justify that the utility organization can derive from it.

The backbone may be composed of mixed technologies. Some portion of the network may be established with optical fibre, and some may be with copper wire, and also some may be wireless. All options and their suitability for different portions will be studies and tested to set up a standard.

The work will finally be tested, and the proposed network will be up for suggestion to be implemented in the smart meter networking scheme.

Keywords: Smart Prepaid Meter, Optical Fiber,

Dedication

With immense love and gratitude, we would like to dedicate this work to our parents, friends and mentors whose creative intellect have inspired us to take up this task, and whose kind words motivated us to complete what we had started.

Acknowledgement

All praises and gratefulness to Almighty Allah, the Lord of the world, for giving us the strength to proceed and persevere to accomplish our task.

With gracious privilege we express our deep sense of thankfulness to our honourable supervisor Dr. S. M. Lutful Kabir, Professor, Institute of Information and Communication Technology, BUET; for his immensely knowledgeable guidance and support which led us to the completion of this dissertation. His generosity and helpful demeanour were the keys to make work a wholesome one.

We would like to offer our gratitude to Engr. Rashedul Hasan, for his continuous cooperation throughout the research.

We would also like to use this opportunity to express our thanks to all the utility personnel who provided the necessary data to make this work a reality.

Our gratitude extends to all the staff of the Department of Electrical and Electronic Engineering of BUET and BRAC University.

Finally, we wish to remember our families, friends and loved ones one more time for being our pillars of strength.

Table of Contents

List of Figures

List of Acronyms

GSM	Global System for Mobile
PLC	Power Line Communication
RF	Radio Frequency
GPRS	General Packet Radio Service
EMF	Electromagnetic Fields
SMETS2	Smart Metering Equipment Technical Specification Standard 2
M2M	Machine to Machine

Chapter1

Introduction

1.1 Introduction

Electricity is a vital element required for economic growth, poverty reduction and social development. Efficient and sustainable usage of electricity has become a global concern. Smart Grids, for many, the next big technological revolution since the invention of the Internet, will play an important role in tomorrow's societies. In most of the developing countries, the idea of the prepaid entering scheme has been introduced.

Smart meters, along with the prepaid system, have become popular since it provides better facilities for both consumers and utility. For consumers, there is no hassle of disconnection and re-connection. Moreover, the utility receives payment earlier and keeping records are not needed. The objective of this thesis work is to develop a backbone metering network mainly, which once installed, can be used for efficient information exchange. Serial communication is a communication technique where data transfer occurs by transmitting data one at a time in sequential order. In our system, serial communication has been used for a combined network so that servers are not required for communication which reduces cost. Optical fiber has been used for communication as regular maintenance is not required, and it is also waterproof.

Moreover, they can increase the reliability of the electricity supply (reduced blackout rate) by real-time measurements, monitoring and control of the generation, and transmission and distribution networks. Further, they can render the utilization of base-load power plants and electricity transport infrastructure more efficient, deploying dynamic pricing and demand response strategies.

1.2 Objective

At present, in Bangladesh, there are around only 20,000 smart meters in use. While it is the ultimate goal of the government to switch to the smart metering technology, the main obstacle faced by them is the mode of communication that will be used to connect this vast number of devices. While there has been some experimentation done with PLC and RF technology, the majority of these 20,000 smart meters are connected using the GPRS technology.

However, there have been problems associated with this existing system of communication using GPRS technology. These issues will be discussed in greater detail in the following pages. Nevertheless, to state primarily, the objective of this thesis is to develop a backbone metering network which will address those difficulties of the previous technologies and try to put forward a new and unique metering network system.

With the present GPRS module of Bangladesh, the billing information of rural areas is not sent and appropriately received as the network towers cannot retrieve the messages from the existing smart meter.

To develop a sustainable and efficient smart meter, this can provide a stable networking system. Moreover, it will be more reliable and cost-effective than the existing smart metering system. A low-cost meter is essential for the third world countries that are unable to afford to connect the whole country to the smart power grid.

To replace the present wired network with optical fibre cable for fastest data transfer with less maintenance which in turn will provide low costing service in future.

1.3 Motivation

This study motivated us to do some work in this line to help the utility companies to switch to the smart metering scheme so they could prevent this loss from taking place.

Since the power system is not affordable yet, electricity does not cover the whole of Bangladesh adequately. As our smart meter is cheap, consumers can afford it very quickly. Moreover, the whole of Bangladesh can be connected to a smart power grid through our system.

1.4 Literature Review

Several remarkable works have been done on smart meter networking scheme ranging from the design of a simple meter to a master plan for an entire feasible smart grid system for a nation.

In [1] a detailed explanation has been presented of what a prepaid meter is, and some of the drawbacks of the existing prepaid metering system. Furthermore, a framework for a smart metering system is proposed, followed by the design of a prototype of the prepaid meter in the discussion.

[2] Discusses the various Wireless technologies that may be used in a smart meter networking scheme. It presented the various advantages and disadvantages associated with each technology and was credible and relevant to our research because the write was done in the context of India, which is a similar economy to us in terms of population pressures and resource scarcity.

The primary analysis of [3] revolves around the growing popularity of IoT (Internet of Things) and how this technology can be used as the backbone of the smart metering network.

[4] Brings in light a rather pressing issue of RF and EMF radiation emitted from smart meters, which are well above the permitted figures. It was eye-opening to us because we understood from this paper that while designing our scheme, we had to take into account the fact that it must be low emitting and less harmful to the environment.

[5] Gives a comprehensive insight into a smart grid system.

1.5 Organization of the Thesis

This book is organized in six chapters which are given as follows:

- Chapter-1 introduces the reader to our work through a summary of specific notable related works which have been done in this field by some previous groups, and the objective and motivation behind our work.
- Chapter-2 is divided into three different sub-topics; the first explains the basic definition of what a smart metering network is, the latter ones discuss some categories of this network and describes the present scenario of the smart metering network in Bangladesh respectively.
- Chapter-3 establishes the detailed foundation of the four basic hardware components to our readers that were used in our project.
- Chapter-4 examines the work we have done; it unfolds the hardware setup and interprets the impenetrable programming language of the software codes used in the most convenient style so that it is easily comprehensible to our readers that what the program is executing.
- Chapter-5 extends the outcome of our hardware implementation to and provides evidence of successful completion of our task.
- Chapter-6 closes with how our work may be applied to the smart metering scheme to make it more effective.

Chapter 2

Some General Discussion on the Topic

2.1 Basic Concepts and Definitions of Smart Meter and Metering Network

The smart meter has become a vital part of our life. It is capable of handling many vital issues that are needed for electric supply. Moreover, it sends data to the electricity provider for observing and changing. It also records hourly consumption daily. Between the meter and the central system, two-way communication is created by the smart meter.

Smart meters work differently, depending on whether it is used for measuring electricity or gas. A smart meter is connected to the mains and controls how much Power has been consumed.

The smart meter is still a new concept in Bangladesh. However, local production of the smart the meter will save foreign currency which in turn can create scopes of employment.

Moreover, the chances of exporting the smart meter after fulfilling local demand will increase.

Sources working in Power Division informed that the current price of a prepaid meter in the market is around Taka 4,500/ and the locally produced meter will be around Taka 3,780/ saving around Taka 720/ in foreign currency.

The Dhaka Electric Supply Company Limited (DESCO) has initiated a move to implement a

Taka 1.86 billion projects to supply and install some 0.2 million smart prepayment meters. The main objective behind this is to improve client services provided by DESCO through digitalization as well as land controlling and improving the revenue realization management.

It will also reduce the technical losses and misuse of electricity.

The primary purpose of a smart meter is to quantify the voltage and current flow at periodic interim. It then sums this up to determine the Power used in a half-hour period. Similarly, for gas, the flow is measured at periodic interim. These details can be sent to both the In-Home Display and service provider. Various technologies will be implemented in different areas to allow WAN to send and receive data from both ends. Moreover, energy calculation and power consumption will be more consistently monitored by their place. There are several benefits of using smart meters.

<u>For consumers</u>

-Consumers can be informed remotely or locally on energy costs

-energy consumption of electrical equipment at home can be displayed on the appliance

-allows electrical appliances to be automatically controlled

-allows the consumer to reduce costs by increasing energy consumption during off-peak cheaper tariff periods

For utilities

-gain first-class data

-influence the energy consumption of their users

-a reduction in 'costs to serve.'

-more effective grid management

-help for revenue protection

-a new communication channel to customers

For national

-helps to convince customers to manage their consumption better and reduce usage leading the way to improved service levels through the better billing system

-implement liberalization of energy markets

-allow the full realization of the Energy Services Directive

2.2 Categories of Metering Network

1. Fibre Optics Network

A fiber optical network seems to be the popular solution for Smart Grid. Optical fibre is the fastest and highest bandwidth network available, supporting a good range of communication protocols and services. Fiber optics are commonly used in telecommunication services such as Internet, television and telephones. They are mostly used for long-distance and high -performance data networking. Fiber optic cables are mainly used for their advantages over copper cables as they support higher bandwidth capacities. They are also stronger, thinner and lighter than copper wire cables and regular maintenance is not needed. Military and space industries also make use of optical fiber as a means of communication and signal transfer, in addition to its ability to provide temperature sensing. Furthermore, Fiber to the Home (FTTH) networks is future proof. However, this communication infrastructure does not play a notable role within the many broadband markets yet. The low penetration of fibre as an access technology relates to the expenses of deploying FTTH-networks. The prices range between € 500 in dense urban areas and € 2000 in rural areas (Casier et al. 2008).

Considerably increasingly strengthened participation among utilities and media transmission administrators (Tahon et al. 2013) cannot change the prediction that FTTH systems will not be an alternative within the up and coming time to assist end-client oriented Smart Grid apparatuses to a significant degree. The equivalent applies to network-oriented apparatuses within the electricity grid. Here, the network elements (e.g. substations) are usually not equipped with access to fibre networks. From an expenses point of view, it is far too expensive to deploy fibre networks just for Smart Grid purposes. The low penetration of fiber as an access technology relates to the expenses of deploying FTTH networks

2. DSL (Digital Subscriber Line)

Digital Subscriber Lines (DSLs) are high-speed Internet connections which used standard telephone lines. Most of the broadband connections are supported by DSL. However, DSL is prevalent in urban areas, mostly whereas in rural areas the infrastructure remained less developed. These lines were initially developed for voice signals and are restricted in bandwidth and data rate. As a result, the utilization of DSL [22] faces significant challenges in terms of access line regulation, dependence on the customer and installation cost. A DSL connection does not require any upgrading cost to the phone system as it makes use of existing telephone wiring.

Moreover, users can use both the telephone line and Internet simultaneously as voice, and digital signals are transferred at different frequencies. There is also a wide range of choice for consumers according to connection speeds and pricing from various providers. On the other hand, DSL only works over a limited physical distance and remains unavailable in many areas where the local telephone infrastructure does not support DSL technology and so the service is not available everywhere. The connection is faster for receiving data than it is for sending over the Internet.

DSL also will likely incur higher installation costs compared to the power line or certain wireless technologies. As a rule, the wired infrastructure needs an additional wired or remote connection to the Smart Meter for which the assent of the proprietor of the premises is required. Both the additional framework and also the necessity to incorporate the proprietor increment the one-time expenses of the organization. Concerning one's interest to avoid a lock-in effect, it can be a shortcoming that DSL providers usually offer a product bundle comprising connectivity and IT services, like data [23] management which is principally focused on littler utilities which do not have their facilities.

3. Power Line Communication

Power line communications (PLCs) use low voltage power lines as a communication medium for data. Some utilities have already used PLC for load control and remote metering. It is simply integrated into the smart metering system since the power lines already reach the meter. The employment of power line is inspired by both the lacking of commercial offerings and the concern of one to exercise regulation over the communication infrastructure given that they have responsibility for grid operation. The power line technology works by imparting modulated carrier signals on the power transmission wires. Interaction is limited within each line section between transformers because data signals cannot penetrate through transformers (Wang et al. 2011). Utilities mainly employed single carrier narrowband techniques until recently. The narrowband power line is now being upgraded to broadband systems operating in higher frequency bands. With this, data rates up to 200 Mb/s can be obtained. While power line innovation could not contend with wired broadband advances (e.g., DSL, link) in the mass market for broadband associations, this is not the situation in the Smart Grid presenting new market chances to this innovation. Promoters of the power line (Schönberg 2012) feature the fact that the physical lines required for PLC [24] already exist. It could lead to lower (operating) costs, although we have to bear in mind that the cost structure depends on the number of connected devices. As referenced above, because of lower outer expense and general line accessibility, the power line innovation is frequently utilized in smart meter pilots. Furthermore, the power line can be utilized for home zone arrange (HAN) purposes.

Moreover, the data rates required for the various applications are inside the broadband power line capabilities. The weaknesses of the customary narrowband power line are as per the following: An essential element of Smart Grid is the ability to transfer data in real-time. That applies basically to the frequency stability and power flow status in low and medium voltage grids. Here, the office needs credible data in order to keep the network stable. When narrowband power line is utilized, the technology does not always fulfil this specification. Pilots have revealed this observation. Another shortcoming of the power line relates to interference. If power line technology is used nationwide to connect up to 80 per cent of smart meters, it is not clear how the power line will affect radio applicants (e.g., broadcasting services). According to the German security profile characterized by BSI, the limit of the narrowband power line is not adequate to convey the metering traffic with required security highlights. As a result, broadband power line is by all accounts the more applicable innovation for Smart Grid. Be that as it may, the broadband electrical cable uses a similar frequency spectrum as an in-house power line for LANs and some radio applications (i.e., emergency services). In this way, interference on same grid communications brought about by other utilization just as aggravations of different clients can happen, mainly on account of broad utilization of broadband power line for smart metering. Since broadband power line uses high frequencies, the signal attenuation is a lot higher than for the narrowband power line. Besides, the regulatory emission limits for radio applications' security limit power line activity to generally small signal levels because line amplifiers are frequently required if the gap between substations and smart meters is too lengthy. Additionally, the signal coupling between the power line modem and power grid, and therefore signal strength, is highly unforeseeable because of unspecified and time-varying impendency of the power grid for higher frequencies. Likewise can prompt usage of extra line amplifiers for higher accessibility.

What is more, in case of a blackout, correspondence over the power line is not accessible, a profoundly unfortunate circumstance. Subsequently, the power line is not effectively prepared for sophisticated applications in the power grid. Lastly, the broadband power line is a generally early-stage and non-normalized innovation. Picking broadband power line facilities for smart metering and basic network-oriented applications show up in any event unsafe.

4. Mobile Network

Most of the people assume that "mobile network" refers to a wireless network.

However, a mobile network and a wireless network are two different types of networks. A mobile network is mainly a communication network scattered through a considerable area around the world and is connected wirelessly by transceivers at fixed locations known as cell sites or base stations.

A mobile network [25] provides access to the distributed networks through a portable device that can be carried anywhere as long as it is under the network coverage area.

On the other hand, a wireless network provides a fixed or mobile endpoint to access a distributed network.

5. GSM (Global System for Mobile)

GSM (Global System for Mobile communication) is a digital mobile network widely used by mobile phone users in Europe and other parts of the world. GSM uses a variation [26] of time division multiple access (TDMA) and is the most widely used of the three digital wireless telephony technologies: TDMA, GSM, and code-division multiple access (CDMA). GSM digitizes and compresses data, then sends it down a channel with two other user data streams, each in its time slot. It operates at either the 900 megahertz (MHz) or 1,800 MHz frequency band. Other than GSM/GPRS in the 900 MHz band, most mobile operators have not changed to another innovation (like UMTS or LTE).

Consequently, Smart Grid in the 900 MHz frequencies utilizes GSM/GPRS. Different mechanization may be put in use after 2020. Presently, preliminaries in Germany and the other EU Member States use GSM/GPRS as the WAN for the Smart Grid. 9 The utilization of this innovation reflects how GPRS is the leading portable technology is open wherever in the nation.

Moreover, GSM/GPRS systems at 900 MHz can give sensible indoor inclusion, and data services are accessible at a sensible expense. In any case, watching the prerequisites illustrated above and considering preliminaries (Ernst and Young 2013, 49), it becomes clear that GPRS has some critical constraints; specific presentation necessities are not met by GPRS (e.g., inactivity). The indoor coverage of GSM systems is presumably the best of any commercial networks, given its 900 MHz range band and far-reaching territory inclusion, yet pilots have demonstrated that the inclusion is regularly lacking for Smart Meters. Since in the 900 MHz bands, all frequencies committed for portable communication (or mobile access to the web) have just been dispensed, there is no likelihood of conveying a private system. As an outcome, GSM/GPRS in 900 MHz is just accessible as help on commercial networks with setbacks regarding security, control, accessibility, and strength. Here we need to hold up under the main priority that GSM/GRPS systems have been planned, dimensioned, and conveyed to offer portable voice communication to end clients. The networks are ill-prepared to handle millions of devices' signaling traffic, which occurs with the advanced metering infrastructure on top of mass-market voice services. Service guarantees and dedicated quality of service to ensure mobile operators cannot guarantee priority message delivery. Besides, it is expected that other technologies will substitute GSM/GPRS in the midterm, which is more efficient. Thus, it can be assumed that mobile operators will not invest in this technology to improve the network quality for just one class of service. In synopsis, even though the utilization of GSM/GPRS is the least complicated option accessible to utilities, as it keeps away from any other choice on correspondence arrangements and related ventures and organizations, there is a mismatch between the utilities' prerequisites and the abilities and lifecycle of

GSM/GPRS. While GSM/GPRS is predominant in current preliminaries in light of its complete accessibility, apparently GSM/GPRS is by and large observed as a mediocre arrangement when an undeniable turn out of the AMI is thought.

6. LTE (Long Term Evaluation)

LTE (Long Term Evaluation) LTE as another worldwide standard for portable (broadband) information has pulled in significant consideration with regards to Smart Grid (Brown/Kahn 2013). LTE [27] is an outstanding, easily deploy able network technology, offering high speeds and low latency over long distances. LTE systems are at present worked in the recurrence groups at 800 MHz, 1800 MHz, and 2,6 GHz. Considering the expanding interest for mobile data services, it is accepted that mobile operators will request extra frequencies in a mid-term point of view (Kürner 2013). Because of proliferation attributes, just LTE at 800 MHz offers the chance to give the essential indoor coverage by and large. Be that as it may, for now, LTE at 800 MHz does not give across the nation inclusion in most European nations. Besides, in individual districts, LTE is or will be the central technology that offers access to the Internet.

Consequently, network capacity is limited. Although LTE 800 fulfills the performance requirements broadly (e.g., latency), we have to consider that LTE 800 is firstly being rolled-out by commercial operators only10 and secondly, will not be designed optimized for M2M traffic. The current use instances of LTE vary fundamentally from the utilization case we have distinguished in Smart Grid. The opportunity costs emerging in the occasion Smart Grid traffic swarms out mass-market applications in the retail market will affect valuing the Smart Grid traffic, making it progressively costly. The better LTE prevails in the market, the higher the opportunity costs, which in this way must be reflected in the discount costs for Smart Grid services. The telecommunication operator has to bear in mind that M2M traffic with many installed devices, like the Smart Meter, uses a large part of network resources. Congestion in commercial networks can only be avoided if the network concerned operates with sufficient frequency bands below 1 GHz and is optimized for M2M traffic patterns. In summary, although LTE 800 fulfils some requirements, there are doubts whether it is technically and economically feasible for mobile operators to enter the market for Smart Gird communication. Furthermore, choosing commercial LTE 800 services for Smart Grid communication faces several of the shortcomings of GSM/GPRS 900: chiefly no control over the network and, therefore, a lock-in to providers, which will mainly service the majority user industry for income return motives.

7. CDMA (Code Division Multiple Access)

Code Division Multiple Access (CDMA) is a sort of multiplexing that facilitates various signals [28] to occupy a single transmission channel. It optimizes the use of available bandwidth. CDMA allows up to 61 concurrent users in a 1.2288 MHz channel by processing each voice packet with two PN codes. There are 64 Walsh codes available to differentiate between calls and theoretical limits. Operational limits and quality issues will reduce the maximum number of calls somewhat lower than this value. The technology is commonly used in ultra-high-frequency (UHF) cellular telephone systems, ranging between the 800-

MHz and 1.9-GHz.The 450 MHz frequency band is accessible in numerous European nations (being either underutilized or unassigned) and has in correlation with 800 MHz and 900 MHz a unique advantage. Because of its lower frequency range, the propagation characteristics empower much-preferred building penetration over GSM 900 and LTE 800. Mobile networks in the 450 MHz range also require roughly multiple times fewer base stations than mobile networks at 800 and 900 MHz (independent of the technology used), subsequently offering much better financial prospects. As of now, the standardized CDMA (3G) technology is accessible at 450 MHz. In the mid-term, the LTE mechanism will turn into another option. Standardization of LTE for 450 MHz is in transit, and the first providers have LTE gear accessible. Almost certainly, operators in Brazil will use LTE in this frequency band beginning in 2014. Notwithstanding, LTE presently cannot seem to be advanced for M2M use cases. CDMA EV-DO Rev. A gives a data throughput of 1.8 Mb/s in the uplink and 3.1 Mb/s in the downlink.

The latency criteria of network-oriented applications are likewise met. Existing CDMA450 organizes over 60 nations, and a well-developed supply ecosystem guarantees longer-term gear accessibility. CDMA innovation is as a rule additionally upgraded for M2M use by new standard improvements, for example, CDMA 1X Rev F., An ongoing investigation of the financial aspects of a Greenfield CDMA450, organize uncovered low unit cost making the CDMA450 arrangement unrivaled as well as more financially feasible with the power line and GSM/GPRS (Sörries 2013). In rundown, as spectrum is accessible in numerous nations, CDMA (or then again LTE) at 450 MHz is a legitimate chance for the application in Smart Grid correspondence. Contrasted with the choices of GSM/GPRS 900 and LTE 800, the 450MHz band is not broadly utilized in Western Europe for the mass market organization for mobile services. The range is either unassigned or right now underutilized in numerous nations, making it accessible for Smart Grid correspondence. It is not just in fact, yet besides a spectrum/network accessibility outlook, very appropriate for conveying/working systems that are entirely or in part devoted to Smart Grid.

2.3 Present Scenario of Bangladesh's Metering Network

Prepaid meter users pay for their energy before using it - usually by adding money to a 'key' or smart card, which is then inserted into the meter. At present, Bangladesh has around more than 22 lac prepaid meters in use throughout the country. Our country is slowly advancing towards replacing all the electricity meters with prepaid meters. The reason behind this is that prepaid meters benefit consumers in that it helps them manage their resources more efficiently and reduce wastage of electricity. Financially they benefit that there is no minimum charge associated with prepaid meters.

A significant part of the electricity generated by power companies is lost or remains accounted for. This is partly due to the power plants' technical losses and the transmission and distribution lines. There is also another 5-7% loss to various non-technical issues at the customer level. However, various efforts have been made to address and solve non-technical issues such as contracting out meter reading and billing, authorized billing system. Unfortunately, none of the systems was much successful. However, the prepaid metering system has been one of the most cost-effective and hassle-free system. Power distribution companies will set up 22, 26,600 smart prepayment layers in the financial year 2020-2021. Unified prepayment System is now generating prepaid energy token through third-party bending operator and BPDB's bending station. It minimizes the need for a considerable number of vending stations to deal with the vast number of consumers, which requires a large workforce for operation. In this system, BPDB appointed Grameenphone and Robi to provide vending service to the prepaid meter consumer of BPDB through Mobile USSD and Mobile Apps. This system's main objectives are to reduce the setup cost of massive vending stations and improve customer service. Moreover, the system becomes more sustainable as the payment is secure, and the user can recharge his prepaid meter anytime from anywhere.

The procedure of manual reading is shown in the picture below.

Figure 2.1: ‘Meter Readers’ collecting readings off post-paid meters.

On the other hand, the scenario becomes much simplified with a prepaid meter, as shown in the following picture.

Figure 2.2: A digital Prepaid Meter used by DESCO, being recharged by a consumer.

There is no hassle of reading or billing in prepaid meters, as was in the previous case.

The next technology that follows prepaid meters is the smart meter. Smart" indicates information exchange and programmable capabilities. Using smart metering technologies, the utility companies can monitor the data of meters (electricity consumption details, demand, Power, and Tamper information, if any. Remotely, they can also control the meters remotely, which includes disconnection of Power to any consumer, reducing the permissible load to any consumer, or any other relevant programming at the consumer end. At present, in Bangladesh, there are around only 20000 smart meters in use. While it is the government's ultimate goal to switch to the smart metering technology, the main obstacle faced by them is the mode of communication that will be used to connect this vast number of devices. While there has been some experimentation done with PLC and RF technology, most of these 20000 smart meters are connected using the GPRS technology.

Few main problems discovered by the GPRS technology users are that it creates an inevitable dependency of the utility providers on the telecommunication companies. The technology has proven to be quite expensive since the private telecommunication companies charge high rent. This process's reliability is also to question because, in many rural areas, it has been observed that the connectivity is relatively low.

Chapter 3

Description of the Basic Components

3.1 Arduino Board

Arduino is an open-source electronics platform supported by easy-to-use hardware and software. Arduino boards can read inputs and create a corresponding output.

An Arduino Uno board, together with its key points, is given below for the readers' convenience.

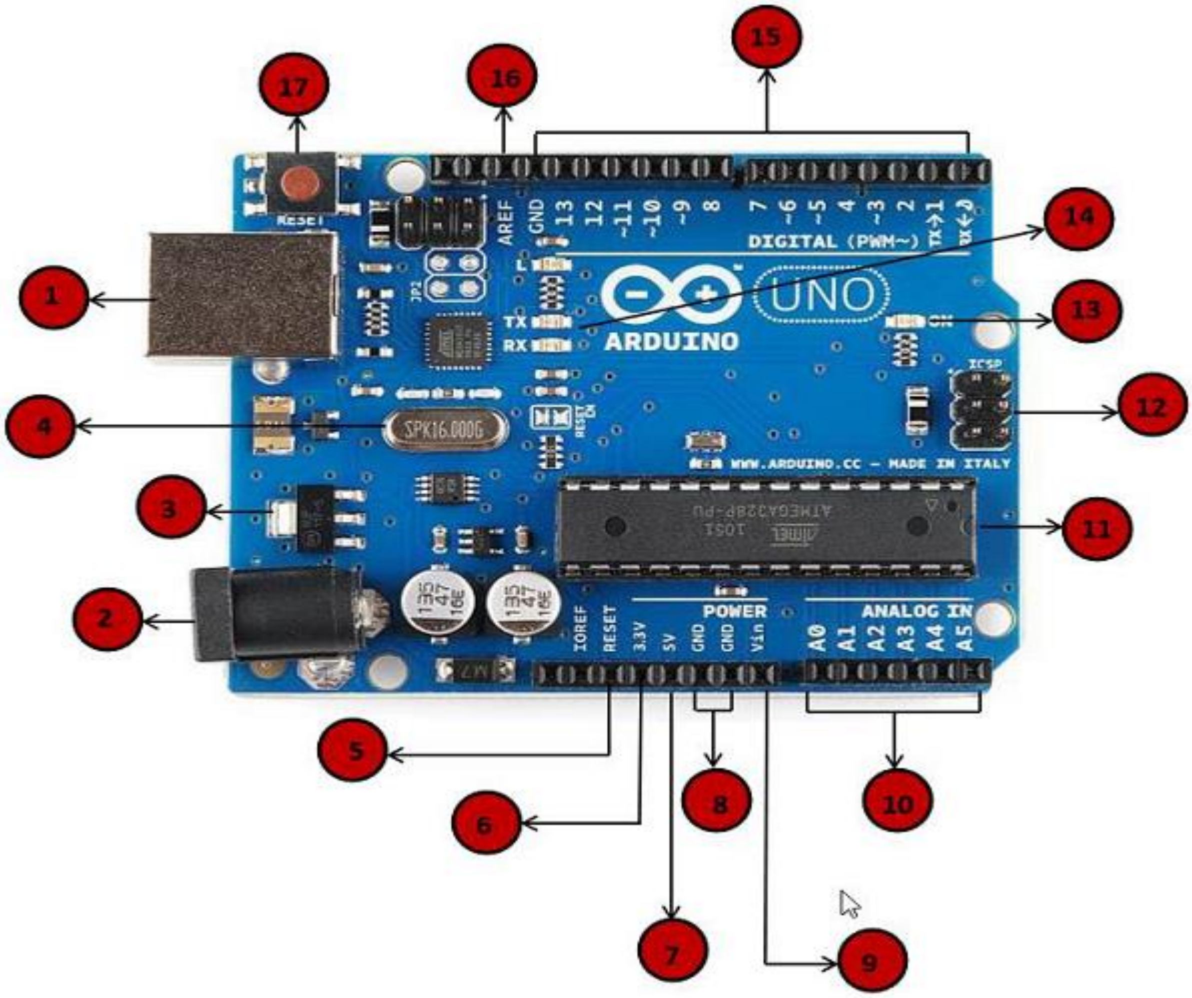

Figure 3.1: An Arduino Uno Board, along with its key points marked.

1	Power USB Arduino board is activated by using the USB cable from the computer. All required is to attach the USB cable to the USB connection (1).
2	Power (Barrel Jack) Arduino boards are activated straight from the AC mains power supply by attaching it to the Barrel Jack (2).
3	Voltage Regulator The voltage regulator's purpose is to regulate the voltage given to the Arduino board and form the DC voltages utilized by the processor and other elements stable.
4	Crystal Oscillator The crystal oscillator helps Arduino in handling the time concerns. How does Arduino measure time? By using the crystal oscillator. The digit printed on top of the Arduino crystal is 16.000H9H. It means the frequency is 16,000,000 Hertz or 16 MHz

5,17	Arduino Reset One can reset the Arduino board, i.e., begin the program from the start. There are two ways to reset a UNO board. Firstly, the reset button (17) on the board may be used to do this. Secondly, one can attach an external reset button to the Arduino pin labeled RESET (5).
6,7 8,9	Pins (3.3, 5, GND, Vin) • 3.3V (6) − Supply 3.3 output volt • 5V (7) − Supply 5 output volt • Most of the components used with the Arduino board works well with 3.3 volt and 5 volts. • GND (8)(Ground) − There are several GND pins on the Arduino, any of which might be accustomed to ground your circuit. • Vin (9) − This pin can also be accustomed to activate the Arduino board from an external power source, like an AC mains power supply.
10	Analog pins A0 through A5, the Arduino UNO board, has six analog input pins. These pins can read the signal from an analog sensor just like the humidity sensor or temperature sensor and convert it into a digital value that may be read by the microprocessor.

11	Main microcontroller Each Arduino board has its microcontroller (11). One can presume it to be the brain of the board. The main IC (integrated circuit) on the Arduino is slimly contrasting from board to board. The microcontrollers are generally of the ATMEL Company. One must know what IC the board has before loading up a brand-new program from the Arduino IDE. This fact is accessible on the highest point of the IC. For more information about the IC fabrication and roles, one can look into the datasheet.
12	ICSP pin A small programming header for the Arduino comprising MOSI, MISO, SCK, RESET, VCC, and GND is called the ICSP (12). It is an AVR. It is usually called an SPI (Serial Peripheral Interface), which may well be assumed as an "expansion" of the output. In reality, the output device to the master of the SPI bus has slaved.
13	Power LED indicator LED should glow after connecting Arduino into a power source to signal that the board is activated in the right way. If this light does not activate, then there is something wrong with the connection.
14	TX and RX LEDs On the board, one will see two labels: TX (transmit) and RX (receive). They emerge in two spots

	on the Arduino UNO board. Firstly, at the digital pins 0 and 1, the pins are accountable for serial communication. Secondly, the TX and RX led (13). The TX led glows with different speeds while sending the serial data. The speed of flashing relates to the baud rate employed by the board. RX flashes during the receiving process.
15	Digital I/O The Arduino UNO board has 14 digital I/O pins (15) (of which 6 render PWM (Pulse Width Modulation) output. These pins may be set up to work as digital input pins to read logic values (0 or 1) or as digital output pins to operate different modules like LEDs, relays, and other devices. The pins labeled "~" may be employed to generate PWM.
16	AREF AREF stands for Analog Reference. Setting an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins **is a** particular purpose of this pin.

3.2 Ethernet Module

The Ethernet switch network module happens to be a high-density voice network module. It is mainly employed across Ethernet ports to facilitate Layer 2 switching.

The Ethernet module that was utilized during this setup is that the ENC28J60 Ethernet Module.

A short specification of this module is as follows:

- SPI Interface Clock Speed: up to twenty MHz
- Buffer Size: 8 KB

- Operating Voltage: 3.1 V to 3.6 V (3.3 V typical)
- 5 V Tolerant Inputs

The pin diagram of the connection between the Arduino board and the ethernet module is shown below:

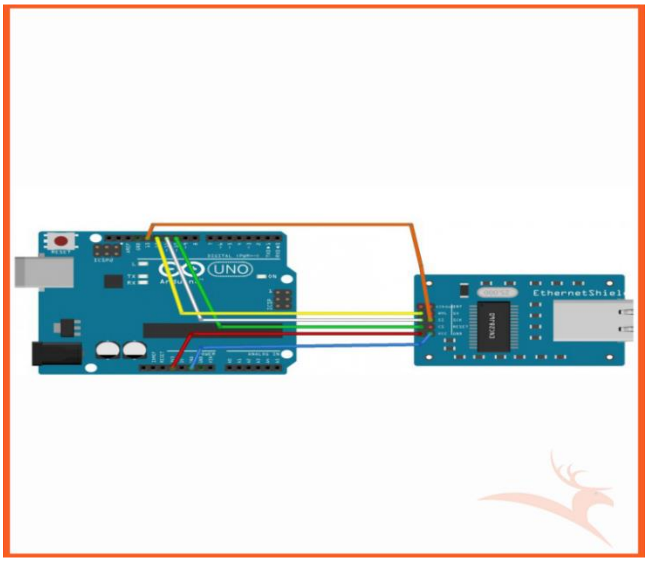

Figure 3.2: The pin diagram of the connection between the Arduino board and the ethernet module.

3.3 Media Converter

A fiber optic media converter is a networking accessory used to make an association in the middle of two unique media types, such as twisted pair with fiber optic cabling. A conventional media converter involves two transceivers that can transmit data to and receive data from one another and a power supply. Each of the transceivers has various connectors that are good with various media. One of the media goes in, and the other type of media comes out.

This media converter works at 5V DC and 1A current.

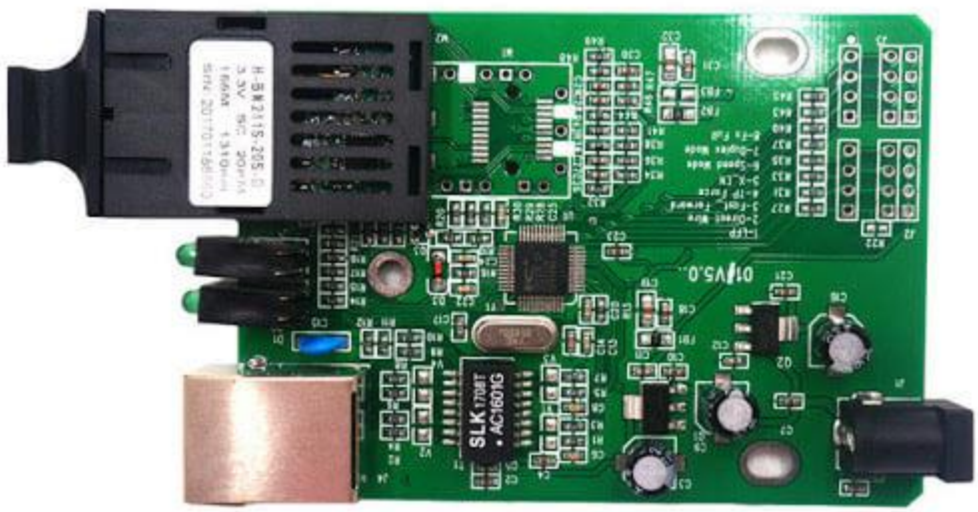

Figure 3.3: Media Converter.

3.4 Optical Fiber

Fiber optics (optical fibers) are extensive, thread-like strings of very pure glass about human hair's width. They are ordered in bundles called optical cables. The purpose of optical cables is to carry light signals over long distances.

Upon looking closely at a single optical fiber, one may find that it has the following parts:

- Core - Thin glass center of the fiber where the light propagates.
- Cladding - Outer optical material enclosing the core that reflects the light into the core.
- Buffer coating - Plastic coating that secures the fiber from harm and humidity

Optical cables are hundreds or thousands of these optical fibers arranged in bundles. An exterior covering, called a jacket, shields the bundles in the cable.

Fiber cables offer several benefits over long-distance copper cabling. Such as:

- Fiber optics endorse a higher capacity. A fiber cable exceeds the network bandwidth that a copper cable can carry with the same thickness. The standard ratings of fiber cables are 10 Gbps, 40 Gbps, and 100 Gbps.
- Because light can travel for much longer distances over a fiber cable without losing its strength, signal boosters' need is lessened.
- A fiber optic cable is less susceptible to interference. For protecting it from electromagnetic interference, a copper network cable requires shielding. While this shielding aids, it is not adequate to prevent interference when many cables are strung together close to one another. The physical properties of fiber optic cables solve most of these problems.

Two views of Fiber Optic cables are shown below:

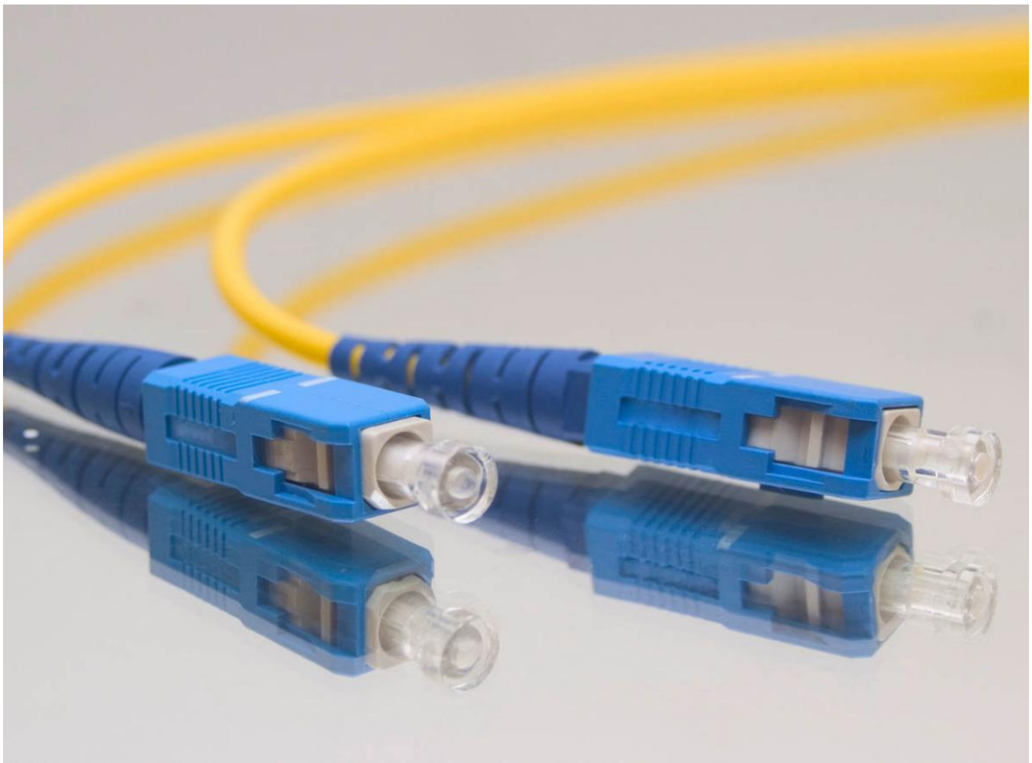

Figure 3.4: Optical Fiber cable (Exterior View).

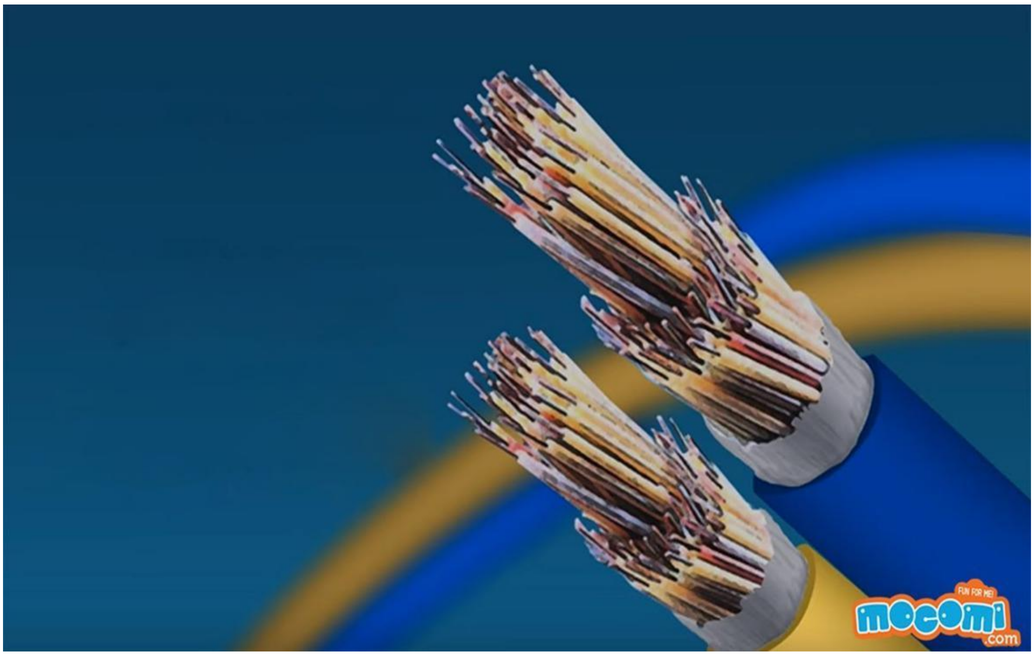

Figure 3.5: Optical Fiber Cable (Interior View).

Chapter 4

Explanation of the Working Principle of the System

4.1 The Detailed Hardware Setup Explanation

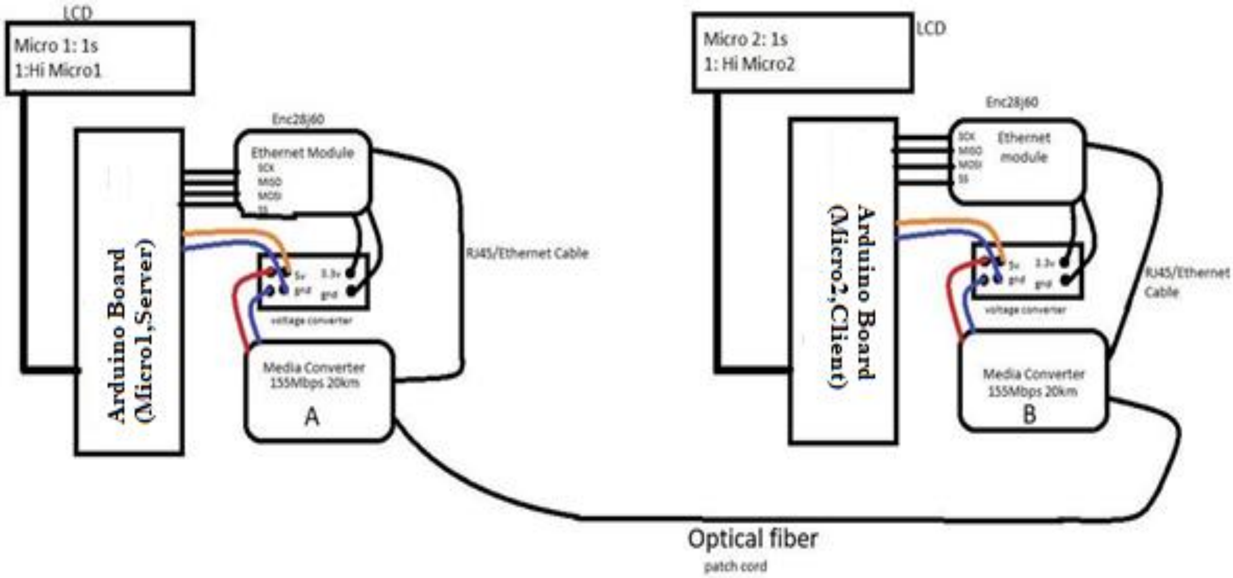

Figure 4.1: Hardware Setup

The three main components, in this case, are the Arduino Boards, Media Converters and Ethernet Modules. Pins of acs721 current sensor Vcc, output, and ground are connected to pins of Arduino A0, A1, and A2. Through SPI protocol, Arduino PINs 11, 12 and 13, which are MOSI, MISO, and SCK, respectively, and CS pins are connected to pins SCK, MISO, MOSI, and CS of Ethernet module encj860. From the Ethernet module, data has been sent to the media converter through rj45 protocol and then passed to the optical fiber. Then PIN of Arduino 2, which is DB7, and PIN 7 is connected to LCD.

At first, the signal passes from Arduino to Ethernet module j2860 through the SPI communication protocol. Data has been sent from the Ethernet module to the media converter through serial communication rj45, and voltage has been converted to 3V. From the media converter, an optical signal is passed to the meter through the patch cord. Meter sends data to the Ethernet module through media converter, and then data is passed to Arduino. Finally, we get output on display through the digital output signal

On one side, the Udpclient code is uploaded, and on the other, the Udpserver code is given. Mainly the client-side transmits data to the server in the form of UDP packets. It sends the string 'HI Micro 1" to our server-side.

This is done at an interval of 5 seconds. After successfully sending the packet it waits for up to 5 seconds for a response on the local port that has been implicitly opened when sending the packet.

The Ethernet Module provides the network which helps to connect all these different devices.

This setup's prominent uniqueness is an optical fiber has been used to connect the two sides, client and server, respectively. The media converters used here help to convert the electrical signals into optical signals and vice versa. A media converter is needed because two different media are being used in this setup.

On the server-side, it is checked whether a packet is received. If a packet arrives,

‘Hi Micro 2’ is written in reply.

4.2 The Code Explanation:

4.2.1 The Client-Side Code:

Explanation of Main Code (Appendix A):

Two libraries were included at the beginning. These libraries will be discussed in detail in the Appendix Sections.

In the beginning, an unsigned long is assigned as 'next.'

The LCD connection settings are done.

A boolean 'displayCountFlag' is made 'true,' which means that for any non-zero value at the Boolean 'displayCountFlag,' it will be considered as correct.

An integer termed as ‘displayCount’ is introduced with the initial value to be zero.

Next, under the ‘void setup’ function, we set the baud rate at 9600. The 'void setup' function runs only once throughout the entire time the program is running.

An ‘uint8_t’ array of 6 hexadecimal digits is made.

The ethernet library is initialized with the ‘Ethernet.begin’ function.

Next, the 'millis' function is used. It returns the number of milliseconds that have been passed since the Arduino board has been powered up. We add 5000 to this value and name it as our variable 'next.'

In the next two lines, we name this Arduino board as 'Micro 2:' and display this name on our LCD.

Next, under the 'void loop' function in the first three lines, we convert the output of the 'millis' function into seconds from milliseconds and display this value on our LCD. The 'void loop' function runs repeatedly.

Next, two other integer variables are introduced as ‘success' and 'len.'

Now one of the main executions of the begins. An 'if' function is used. The condition of this structure ensures that the following tasks are done at an interval of 5 seconds. After 5 seconds has elapsed after the Arduino board is powered, and then after every 5-second interval, using the 'udp.beginPacket' function, a connection is set up to write UDP data to a remote connection.

Next, a ‘while’ loop function is introduced. This section mainly describes how the program will behave during the intervals of 5 the second time. In these two lines of the code, it is instructed to keep the connection off during these interval times by using the 'goto stop' command.

Next, using the 'udp.write' function, the message "Hi Micro 1" is written onto the remote connection. This message is sent when the next function, 'udp.endPacket' is called.

Now when the client has sent a message, next, it waits for 5 seconds to receive a response. The 'udp.parsePacket’ function is used to check for any received packet.

If no response is received within this interval, the program is instructed to disconnect.

The variable 'displayCount' is the number of times that the client has sent a message and received a response in return successfully.

Next, an ‘if’ loop is used. The purpose of this block of codes is to display the number of times the communication has taken place both ways successfully on the LCD. Notice, the Boolean flag ‘displayCountFlag’ is made ‘False’ here to ensure that the count variable is increased by one only if the communication has been completed successfully.

Next, an integer ‘c’ is introduced. Using the ‘udp.read’ function, the characters in the buffer are read and stored in integer 'c' It is very crucial to understand that this output that is stored in integer 'c' is the response that is sent by the other side, i.e. the server-side (Micro 1) in our case. This response is evidence that communication is occurring, both ways, successfully. Thus, we have chosen this output to be shown on our LCD as our ultimate confirmation of successful communication of our project. In the LCD of 'Micro 2' (Client-side Arduino Board), “Hi Micro 2” will be visible and vice versa. This will be further discussed in detail in the ‘Results’ Chapter of out write up.

In the last section of our Client-side Code, ‘udp.available’ function is used to determine the bytes of data available to read. Next, the 'udp.flush' function has been used to discard these bytes that have not been read. Using 'udp.stop' the connection is terminated. Finally, the ‘displayCountFlag’ Boolean is made true once again so for the convenience of the program when the ‘void loop’ loop function repeats itself all over again.

4.2.2 The Server-side Code:

Explanation of Main Code (Appendix B):

The same two libraries are included, and the same LCD connection setup is done as was done in the client-side.

A similar Boolean flag and integer ‘displayCount’ are created.

In the 'void setup' function block, a similar baud rate is set, and the ethernet library is initialized.

We name this server-side Arduino board as ‘Micro 1:’

In the ‘void loop’ block, we use the ‘millis’ function as previously, convert the output from millisecond to seconds and display on our LCD.

We introduce a new integer named ‘packetSize’ in this server-side code. We store the output of the function ‘udp.parsePacket’ in this integer, i.e. we store the size of the received packet here.

In the following lines of the code, if we find any received packet, we proceed by incrementing the value of the integer ‘displayCount’ and displaying this number on our LCD.

In the next section, a character is introduced named 'msg'. This character stores the space in memory using the function 'malloc' of the size 'packetsize+1'.

In the next line, an integer is introduced, namely ‘len’. This integer stores the characters that have been read from the buffer, using the ‘udp.read; function.

In the next line, we use the NULL pointer to direct the program to an empty location in the memory.

Next, we display the message received on our LCD and finally free up the memory.

In this section, it is critical to understand that we will receive the message sent from our Client-side, i.e., "Hi Mico 1". It will display on the server-side (Micro 1) and act as evidence that our communication is working successfully.

In the following section, we recognize the number of bytes available to read and discard them.

Next, we implement the 'udp.beginPacket' to establish a connection to write UDP data to the remote connection.

Using 'udp.println,' a new function, we send back a message, "HI Micro 2" to the client-side.

It is essential to notice that this same message, "Hi Micro 2," is displayed on the LCD of the client-side Arduino board (Micro 2).

In conclusion, it may be brought to the light of the discussion that the server-side sends data and checks for any received data throughout the program is running. The connection is never disconnected throughout this entire time.

On the contrary, the Client-side keeps the connection shut off all the time. It just connects at an interval of every 5 seconds to send data. After sending it, it disconnects again. After the next 5-second interval, it again establishes the connection to send new data and checks for the response of the data sent 5 seconds earlier.

Chapter 5

Results

To test the program, they set up the hardware as follows:

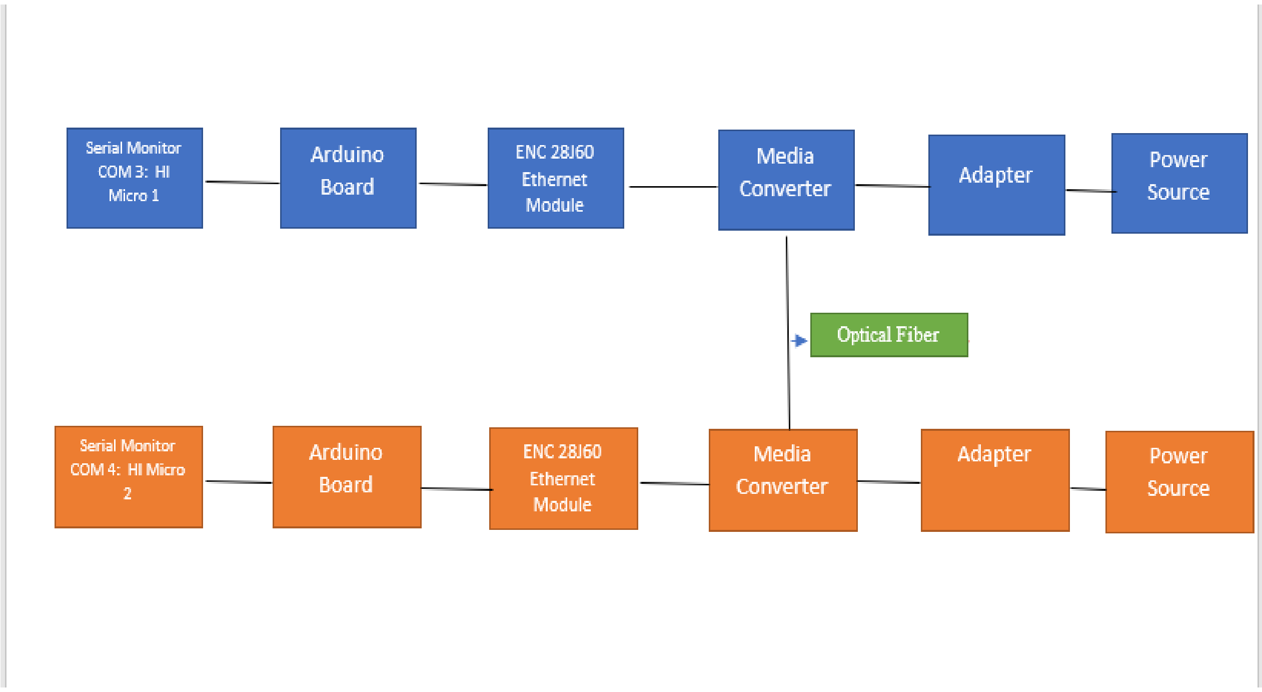

Figure 5.1: How the Hardware Components have been connected for our Project Implementation.

As discussed in the code explanation section, we sent strings of data between the two optically connected Arduino boards. The communication was done in duplex mode since both the boards were sending and receiving data. A greeting is sent to Micro:1 to micro:2. When micro:2 receives this greeting, it sends back a deferential acknowledgment in response.

So Micro:1 will receive a "Hi micro:1" from Micro:2.

This will only happen when Micro:2 receives some message from Micro:1.

So our target was to obtain “Hi Micor:1” in the LCD of our Micro:1 Arduino.

This was achieved successfully.

The following picture shows the successful result of our real-time testing of the hardware.

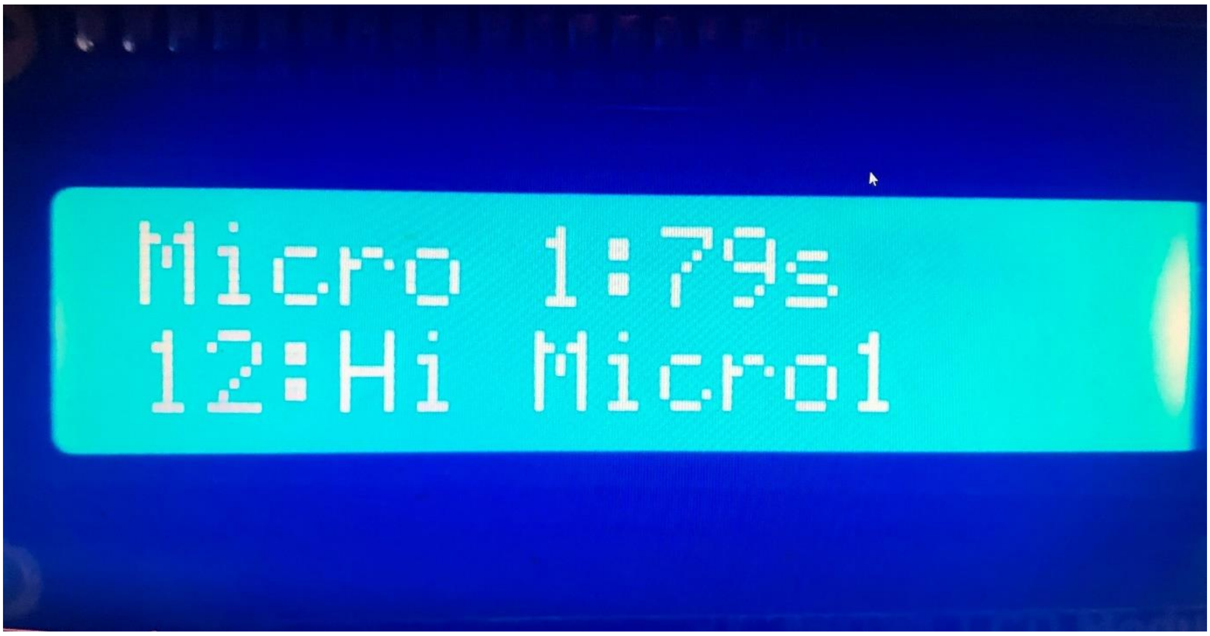

Figure 5.2: The successful output of our hardware implementation.

Chapter 6

Conclusions

Future Work:

The objective proposed thesis work is mainly to develop a backbone metering network that can be used for information exchange almost free of recurring cost once installed.

The uniqueness of this was that it would be composed of mixed technologies. Some portion of the network may be established with optical fiber, and some may be with copper wire, and some may be wireless.

So, we tested out this program and implemented the hardware to show that this is a relatively simple and effective way of communicating.

The benefits that the users will achieve from this setup are that

- copper-based Ethernet connections are limited to a data transmission distance of 100 meters when using UTP cable. The link distance may be increased by up to 80 kilometers or more by using Ethernet to fiber conversion.
- Electromagnetic interference, or EMI, can cause corruption of data over copper-based Ethernet links. Optimal data transmission and good network performance are ensured because data transmitted through fiber optic cable is entirely immune to this type of noise.
- Optical fibers allow converting link speeds from 10 Mbps to 100 Mbps or 100 Mbps to 1000 Mbps.

So, this idea may be applied in the smart metering network infrastructure. This network can be used for electricity, Internet, gas, and all other utilities. So, this would prove to be a very cost-effective and efficient use of resources for the economy.

Reference

1. Parvin, Mst & Kabir, S (2015). A framework of a smart system for the prepaid electric metering scheme. 1-5. 10.1109/ICIEV.2015.7334061

2. M, Patel & Modi, Mitul. (2015). A Review on Smart Meter System. IJIREEICE. 3. 70-73. 10.17148/IJIREEICE.2015.31215

3. urajudeen-Bakinde, Nazmat & AYODELE, Sunday & Oloruntoba, Timilehin & Otuoze, Abdulrahaman & Faruk, Nasir. (, 2017). Development of an Internet-Based Prepaid Energy Meter. 10.1109/AFRCON.2017.8095681.

4. Aerts, Sam & Bossche, Matthias & Vergara, Ximena & Odie, Simon & Verloock, Leen & Martens, Luc & Joseph, Wout. (, 2020). Spatial and temporal assessment of radiofrequency electromagnetic fields emitted by smart meters and smart meter banks in urban environments. Environmental research. 183. 109196. 10.1016/j.envres.2020.109196.

5. Weranga, Kasun & Kumarawadu, Sisil & Chandima, D.P.. (2014). Smart Grid and Smart Metering. 10.1007/978-981-4451-82-6_1.

6. https://www.arduino.cc/en/guide/introduction

7. https://www.tutorialspoint.com/arduino/arduino_board_description.htm

8. https://www.cisco.com/c/en/us/td/docs/routers/access/interfaces/nm/hardware/installation/guide/connswh.html

9. www.openimpulse.com/enc28j60-ethernet-module

10. https://chitraexpress.com.bd/wp-content/uploads/2019/05/1404-large_default-new-enc28j60-ethernet-lan-network-module-for-arduino-robotics-bangladesh.jpg 9ethernet

11. https://www.computercablestore.com/what-is-a-fiber-optic-media-converter
12. https://ryanscomputers.com/detail/d-link-htb-3100a-25km-media-converter

13. https://www.lifewire.com/fiber-optic-cable-817874

14. https://computer.howstuffworks.com/fiber-optic1.html

15. https://www.ul.com/sites/g/files/qbfpbp251/files/styles/hero_boxed_width/public/2019-03/GettyImages-173677758-2664px.jpg?itok

16. https://i.ytimg.com/vi/o5t6evogJbg/maxresdefault.jpg

17. https://www.letsstartcoding.com/syntax/

18. https://www.quora.com/What-is-the-difference-between-a-smart-meter-and-prepaid-meter

19. https://powerdivision.gov.bd/site/page/ae49fd24-a37e-4644-a2c1-638bfc3bf537/প্রিপেইড-মিটারিং

20. https://www.smartme.co.uk/how-they-work.html#gsc.tab=0

21. www.cdg.org

22. https://www.geeksforgeeks.org/digital-subscriber-line-dsl/

23. https://www.sciencedirect.com/topics/engineering/digital-subscriber-lines

24. https://www.sciencedirect.com/topics/engineering/power-line-communication

25. https://study.com/academy/lesson/mobile-networking-definition-components-comparison.html

26. https://searchmobilecomputing.techtarget.com/definition/GSM

27. https://www.extremetech.com/mobile/110711-what-is-lte

28. https://www.tutorialspoint.com/cdma/cdma_technology.htm

Appendix A

The Client-Side Main Code:

```
#include <UIPEthernet.h>
#include <LiquidCrystal.h>
EthernetUDP udp;
unsigned long next;
const int rs = 6, en = 7, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
boolean displayCountFlag = true;
int displayCount = 0;
void setup()
{
  Serial.begin(9600);
  uint8_t mac[6] = {0x00,0x01,0x02,0x03,0x04,0x05};
  Ethernet.begin(mac,IPAddress(192,168,0,108));
  next = millis()+5000;
  lcd.begin(16, 2);
  lcd.print("Micro 2:");
}
void loop()
{
  lcd.setCursor(8, 0);
  lcd.print(millis() / 1000);
  lcd.print("s");
  int success;
```

```
int len = 0;
if (((signed long)(millis()-next))>0)
{
  do{success = udp.beginPacket(IPAddress(192,168,0,109),5000);}
  while(!success && ((signed long)(millis()-next))<0);
  if (!success ){goto stop;}
  success = udp.write("Hi Micro1");
  success = udp.endPacket();
  do{success = udp.parsePacket();}
  while(!success && ((signed long)(millis()-next))<0);
  if(!success ){goto stop;}
  do
  {
    if(displayCountFlag)
    {
      lcd.setCursor(0, 1);
      lcd.print(displayCount);
      lcd.print(":");
      displayCountFlag = false;
      displayCount++;
    }
    int c = udp.read();
    lcd.write(c);
    Serial.write(c);
    len++;
  }
  while((success = udp.available())>0);
```

```
    udp.flush();
    stop:
    udp.stop();
    next = millis()+5000;
   }
   displayCountFlag = true;
}
```

Appendix B

The Server-Side Main Code:

```
#include <UIPEthernet.h>

#include <LiquidCrystal.h>

EthernetUDP udp;

const int rs = 6, en = 7, d4 = 5, d5 = 4, d6 = 3, d7 = 2;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

boolean displayCountFlag = true;

int displayCount = 0;

void setup()

{

  Serial.begin(9600);

  uint8_t mac[6] = {0x01,0x02,0x03,0x04,0x05,0x06};

  Ethernet.begin(mac,IPAddress(192,168,0,109));

  udp.begin(5000);

  lcd.begin(16, 2);

  lcd.print("Micro 1:");

}

void loop()

{

  lcd.setCursor(8, 0);

  lcd.print(millis() / 1000);

  lcd.print("s");

  int packetSize = udp.parsePacket();//check for new udp-packet:

  if (packetSize > 0)
```

```
{
 do
 {
  if(displayCountFlag)
  {
   lcd.setCursor(0, 1);
   lcd.print(displayCount);
   lcd.print(":");
   displayCountFlag = false;
   displayCount++;
  }
  char* msg = (char*)malloc(packetSize+1);//Memory Allocation
  int len = udp.read(msg,packetSize+1);//length of the massage
  msg[len]=0;//NULL char for print() termination
  lcd.print(msg);
  Serial.print(msg);//Print Massage
  free(msg);//Free memory
 }
 while((packetSize = udp.available())>0);
 udp.flush();
 int success;
 do{success = udp.beginPacket(udp.remoteIP(),udp.remotePort());}
 while(!success);
 udp.println("Hi Micro2");
 udp.endPacket();
 udp.stop();
 udp.begin(5000);
```

```
  }
  displayCountFlag = true;
}
```

Appendix C

UDP Library (Client)

```
#if defined(__MBED__)
  #include <mbed.h>
  #include "mbed/millis.h"
  #define delay(x) wait_ms(x)
  #define PROGMEM
  #include "mbed/Print.h"
#endif
#include <UIPEthernet.h>
#include "utility/logging.h"
EthernetUDP udp;
unsigned long next;
#if defined(ARDUINO)
void setup() {
#endif
#if defined(__MBED__)
int main() {
#endif
  #if ACTLOGLEVEL>LOG_NONE
    #if defined(ARDUINO)
      LogObject.begin(9600);
    #endif
    #if defined(__MBED__)
      Serial LogObject(SERIAL_TX,SERIAL_RX);
    #endif
  #endif
  uint8_t mac[6] = {0x00,0x01,0x02,0x03,0x04,0x05};
```

```
  Ethernet.begin(mac,IPAddress(192,168,0,6));
  next = millis()+5000;
#if defined(ARDUINO)
}
void loop() {
#endif
#if defined(__MBED__)
while(true) {
#endif
  int success;
  int len = 0;
  if (millis()>next)
    {
      do
        {
          success = udp.beginPacket(IPAddress(192,168,0,1),5000);
          #if ACTLOGLEVEL>=LOG_INFO
            LogObject.uart_send_str(F("beginPacket: "));
            LogObject.uart_send_strln(success ? "success" : "failed");
          #endif
          //beginPacket fails if remote ethaddr is unknown. In this case an
          //arp-request is send out first and beginPacket succeeds as soon
          //the arp-response is received.
          #if defined(ESP8266)
            wdt_reset();
          #endif
        }
      while (!success && (millis()<next));
      if (success)
        {
        success = udp.write("hello world from arduino");
        #if ACTLOGLEVEL>=LOG_INFO
```

```
  LogObject.uart_send_str(F("bytes written: "));
  LogObject.uart_send_decln(success);
#endif
success = udp.endPacket();
#if ACTLOGLEVEL>=LOG_INFO
  LogObject.uart_send_str(F("endPacket: "));
  LogObject.uart_send_strln(success ? "success" : "failed");
#endif
do
  {
  //check for new udp-packet:
  success = udp.parsePacket();
  #if defined(ESP8266)
    wdt_reset();
  #endif
  }
while (!success && (millis()<next));
if (success)
  {
  #if ACTLOGLEVEL>=LOG_INFO
    LogObject.uart_send_str(F("received: '"));
  #endif
  do
    {
    char c = udp.read();
    #if ACTLOGLEVEL>=LOG_INFO
      #if defined(ARDUINO)
        LogObject.write(c);
      #endif
      #if defined(__MBED__)
        LogObject.printf("%c",&c);
      #endif
```

```
        #endif
        len++;
        }
       while ((success = udp.available())>0);
       #if ACTLOGLEVEL>=LOG_INFO
        LogObject.uart_send_str(F("', "));
        LogObject.uart_send_dec(len);
        LogObject.uart_send_strln(F(" bytes"));
       #endif
       //finish reading this packet:
       udp.flush();
       }
      }
    udp.stop();
    next = millis()+5000;
  }
}
#if defined(__MBED__)
}
#endif
```

Appendix D

UDP Library (Server)

```
#include <UIPEthernet.h>
#include <LiquidCrystal.h>
EthernetUDP udp;
const int rs = 6, en = 7, d4 = 4, d5 = 5, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
boolean displayCountFlag = true;
int displayCount = 0;
double adcVoltage = 0;
double currentValue = 0;
long privMillis = 0;
void setup()
{
  //ADC
  Serial.begin(9600);
  pinMode(A0,OUTPUT);
  pinMode(A2,OUTPUT);
  digitalWrite(A0,HIGH);
  //Ethernet
  uint8_t mac[6] = {0x01,0x02,0x03,0x04,0x05,0x06};
  Ethernet.begin(mac,IPAddress(192,168,0,109));
  udp.begin(5000);
  lcd.begin(16, 2);
```

```
  lcd.print("(S)Unit1:");
  privMillis = millis();
}

void loop()
{
  lcd.setCursor(8, 0);
  //lcd.print(millis() / 1000);
  //lcd.print("s");
  //ADC
  if((millis()-privMillis)>=500)
  {
    adcVoltage = (analogRead(A1) / 1024.0) * 5000;
    currentValue = ((adcVoltage - 2485) / 66);//Sensitivity==66 for 30A
    lcd.print(currentValue);
    lcd.write('A');
    privMillis = millis();
  }
  //Ethernet
  int packetSize = udp.parsePacket();//check for new udp-packet:
  if (packetSize > 0)
  {
    do
    {
      if(displayCountFlag)
      {
        lcd.setCursor(0, 1);
```

```
      lcd.print(displayCount);
      lcd.print(":");
      displayCountFlag = false;
      displayCount++;
    }
    char* msg = (char*)malloc(packetSize+1);//Memory Allocation
    int len = udp.read(msg,packetSize+1);//length of the massage
    msg[len]=0;//NULL char for print() termination
    lcd.print(msg);
    Serial.print(msg);//Print Massage
    free(msg);//Free memory
  }
  while((packetSize = udp.available())>0);
  udp.flush();
  int success;
  do{success = udp.beginPacket(udp.remoteIP(),udp.remotePort());}
  while(!success);
  udp.println("Hi from unit1");
  udp.endPacket();
  udp.stop();
  udp.begin(5000);
}
displayCountFlag = true;
}
```

UDP Library (Client)

```
#include <UIPEthernet.h>

#include <LiquidCrystal.h>

EthernetUDP udp;

const int rs = 6, en = 7, d4 = 4, d5 = 5, d6 = 3, d7 = 2;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

boolean displayCountFlag = true;

int displayCount = 0;

double adcVoltage = 0;

double currentValue = 0;

long privMillis = 0;

void setup()

{

  //ADC

  Serial.begin(9600);

  pinMode(A0,OUTPUT);

  pinMode(A2,OUTPUT);
```

```
  digitalWrite(A0,HIGH);

  //Ethernet

  uint8_t mac[6] = {0x01,0x02,0x03,0x04,0x05,0x06};

  Ethernet.begin(mac,IPAddress(192,168,0,109));

  udp.begin(5000);

  lcd.begin(16, 2);

  lcd.print("(S)Unit1:");

  privMillis = millis();

}


void loop()

{

  lcd.setCursor(8, 0);

  //lcd.print(millis() / 1000);

  //lcd.print("s");

  //ADC

  if((millis()-privMillis)>=500)

  {
```

```
  adcVoltage = (analogRead(A1) / 1024.0) * 5000;
  currentValue = ((adcVoltage - 2485) / 66);//Sensitivity==66 for 30A
  lcd.print(currentValue);
  lcd.write('A');
  privMillis = millis();
}
//Ethernet
int packetSize = udp.parsePacket();//check for new udp-packet:
if (packetSize > 0)
{
  do
  {
    if(displayCountFlag)
    {
      lcd.setCursor(0, 1);
      lcd.print(displayCount);
      lcd.print(":");
      displayCountFlag = false;
```

```
    displayCount++;
  }
  char* msg = (char*)malloc(packetSize+1);//Memory Allocation
  int len = udp.read(msg,packetSize+1);//length of the massage
  msg[len]=0;//NULL char for print() termination
  lcd.print(msg);
  Serial.print(msg);//Print Massage
  free(msg);//Free memory
}
while((packetSize = udp.available())>0);
udp.flush();
int success;
do{success = udp.beginPacket(udp.remoteIP(),udp.remotePort());}
while(!success);
udp.println("Hi from unit1");
udp.endPacket();
udp.stop();
udp.begin(5000);
```

```
  }

  displayCountFlag = true;

}
```