4-22-2011

REAL-TIME POWER MONITORING, HOME AUTOMATION AND SUSTAINABILITY

Wisam Nader

University of Nebraska-Lincoln, wnader@unomaha.edu

REAL-TIME POWER MONITORING, HOME AUTOMATION AND

SUSTAINABILITY

by

Wisam Nader

A THESIS

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Architectural Engineering

Under the Supervision of Professor Mahmoud Alahmad

Lincoln, Nebraska

May, 2011

REAL-TIME POWER MONITORING, HOME AUTOMATION AND

SUSTAINABILITY

Wisam Nader, M.S.

University of Nebraska, 2011

Adviser: Mahmoud Alahmad

As the supply and demand of electrical energy is challenged within the context of environmental awareness and national security, the need to evaluate and advance each side is growing in importance. The electrical distribution system plays a critical role in the built environment as it is an enabling product, an intangible necessity used to power our systems. In order to manage our energy and prevent the consequences caused by energy wastage we need to develop tools for the built environment to become elastic with its consumption to respond to the national smart grid. To advance the research in this field, ZNETH's (Zero-Net Energy Test Home) Power Monitor (ZPM) is proposed and implemented in which end-use detailed energy consumption data is provided for each Node in the building. The data is shown on a display panel and stored as an Excel file to facilitate further processing. This data is displayed on the computer screen via a novel virtual environment building system. The virtual environment gives residents immediate access to the building power consumption information in a user-friendly format. This thesis details the ZPM system and the process that leads to its creation. The basic components of the proposed system were designed and tested using a prototype board.

ACKNOWLEDGEMENT

DEDICATION

I would like to dedicate this work to my wonderful family: My mother Maha Elnadaaf, my father Reda Nader, and my brothers Khaldon and Mahran Nader who are my inspiration, my happiness, and my source of energy. I would also like to dedicate this work to my best friend ever, Manar Amyri, as well as to my other friends: Yuliya Rashchupkina, Emily Rieur, Waael Elhalabe, Rawad Eljaber, Mohamed Amer Chaaban, Hosen Hasna, and Muhammad Zulfiqar. Finally, I give thanks to all the faculty and staff in the Architectural Engineering department who do a great job to make this program one of the best in the nation.

GRANT INFORMATION

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Understanding how energy is spent and knowing how to monitor and control it are key prerequisites for residential energy conservation. Research has shown that, in general, when presented with the appropriate information on energy usage, average homeowners will change their consumption behavior to decrease their monthly electrical consumption and electricity bill. The magnitude of the savings depends on the type of feedback or information offered, cost of power, interface type and format, and other social and economic factors. The use of real-time feedback presents an opportunity to decrease energy consumption by 10%-20% [1-2]. Several real-time monitors (RTM) created to provide real-time information to prompt consumers to curb their electricity use are available in the market. However, their effectiveness is limited due to their interface and mixed information, which range from partial and limited to overwhelming and impractical. It is also limited due to the user's knowledge of what action to take to curb the current wasted energy. Studies [1-2] have demonstrated that it is the interface that fundamentally determines the behavior. To create both energy and monetary impulse saving, we need to create a simple interface that empowers users to take control of their actions. Hence, current RTMs are not designed to take into account consumer behaviors in response to residential energy consumption and conservation.

1.1 RELATION TO POLLUTION PREVENTION

Energy saved is, quite literally, energy found. A reduction of just 5% in global energy use would save the equivalent of more than 10 million barrels of oil a day and provide enough energy to power Australia, Mexico, and the United Kingdom [3]. The residential sector, unlike the commercial and industrial sectors, is made up of multiple small energy users, such as houses, mobile homes, and apartments. Residential energy consumption accounts for 21% of the electricity used in the United States but research shows that almost 41% of that power is being wasted [4]. While we are making improvements in energy efficiencies, we are offsetting it by increased consumption. The U.S. DOE's (Department of Energy) Energy Information Administration forecasts the trend in increased energy consumption to be in plug-in type loads as shown in Figure 1 [5]. Furthermore, they have estimated through their Residential Energy Consumption Survey (RECS) research that miscellaneous plug loads – comprising DOE's "miscellaneous" and "electronics" end use categories – make up 15% of residential electricity consumption or about 1,600 kWh per year. Previous studies by Lawrence Berkeley National Laboratory (LBNL) estimated an average base load of approximately 77 to 87 watts for all loads in a household including hard-wired HVAC equipment and controls as well as white goods [6]. Subtracting out the amount of power attributed to so-called infrastructure, HVAC, and food/beverage end uses – largely white goods or hard-wired products – from LBNL's findings, a base load of 64 to 74 watts is left [7]. This is equivalent to 60W-75W light bulb running continuously in each house hold. With a 127 Million homes in the USA in 2007 [8] that is approximately 7.7 GW of energy required to maintain loads in standby only. This energy, along with other loads remaining on

while not in use, such as lighting in rooms with no occupants, when reduced, will result

in substantial resource savings.
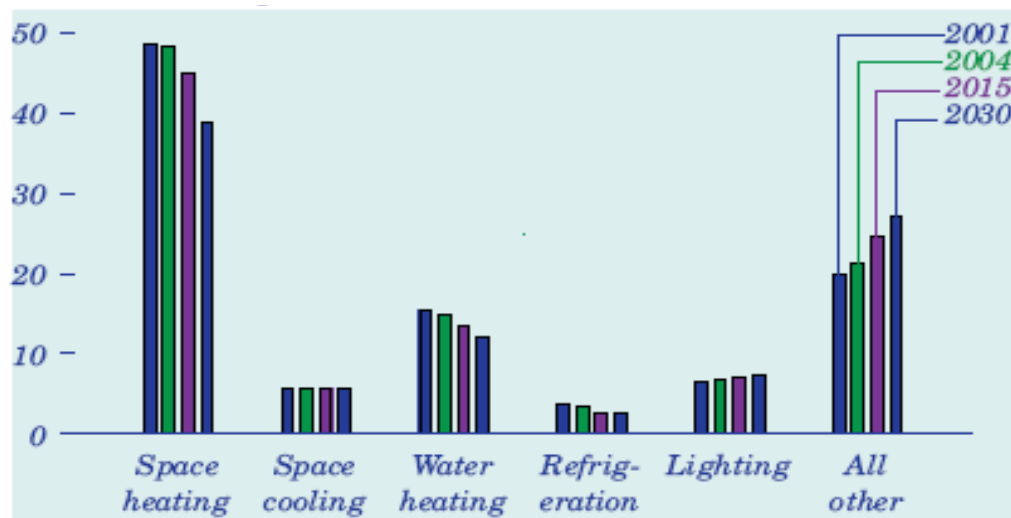
Figure 1: Delivered Residential Energy Consumption By End Use (Million Btu per Household) [5]

Active research is moving toward smart homes, home automation, and smart

meters with demand response integration to shut off loads by the utility when peak

demand is high. This is an intrusive method that collides with user comfort. However,

when the user is empowered to make that decision, the results are tolerated and accepted.

1.2 CHALLENGE DEFINITION

Rising energy demands are necessitating construction of new power plants across our country. To keep utility costs down and to minimize our contribution to carbon emissions, we must change our energy consumption behaviors while seeking efficient delivery and electricity.

According to the National Association of Home Builders [11], the average home size in the United States was 2,459 square feet in 2008, up from 1,400 square feet in 1970. This size increase results in increased energy consumption. This energy consumption can be classified as shown in Figure 2 [12]. As can be seen from Fig. 1, lighting, heating, and cooling take up 58 percent of the annual energy bill for a typical household. The question posed by the Figure is an interesting question that current RTM are trying to address. However, current RTM devices available in the market monitor and display overall energy consumption for the entire home only and do not have the capability to adaptively determine wasted energy consumption on behalf of the resident. The current devices rely on the user to know where the energy is being wasted and hope that the user will take actions after seeing the current usage. Hence we need to focus on creative and promising methods to encourage energy efficient lifestyles.

Where Does My Money Go?
Annual Energy Bill for a typical Single Family Home is approximately $2,200.

11%
4%
12%
13%
14%
29%
17%

- Heating
- Cooling
- Water Heating
- Appliances
  (includes refrigerator, dishwasher, clothes washer and dryer)
- Lighting
- Electronics
  (includes computer and monitor and TV and DVD player)
- Other*
  (includes external power adapters, telephony, set-top boxes, ceiling fans, vent fans and home audio)
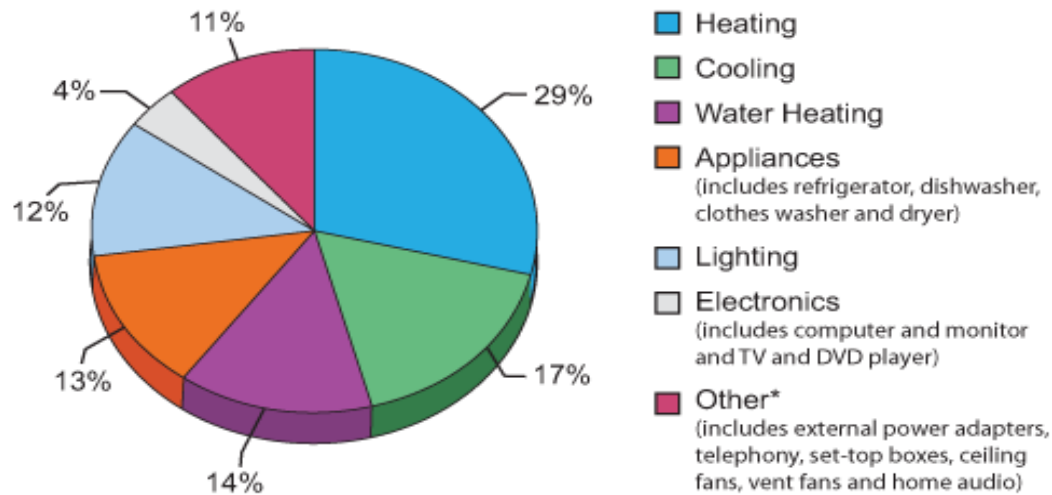
Figure 2: Energy Consumption For A Typical Home

1.1.2 Effectiveness of Real-time Monitoring Feedback in Reducing Residential Electricity Use

When it comes to human consumption behavior, information is power. The standard feedback mechanism for residential electricity consumption has been the monthly bill. While this provides an aggregated use summary in kilowatt-hours (kWh), it does not provide information timely enough to empower consumers to make informed decisions about how and when to reduce their consumption. Research (discussed below) shows that when feedback is coupled with monetary incentives, users changed their energy behavior. Further, the energy consumption feedback has advanced from paper format in the 1970s to graphical and digital RTM in recent years. This is critical in energy conservation since the needs for energy reduction are numerous.

The use of feedback to reduce energy consumption was initially studied following the Arab oil embargo of 1973 by psychologists who were interested in human behavior. Barbara Farhar reviewed several studies incorporating feedback in the form of notes left on the door or kitchen window, meter reading, and self-monitoring [13]. These studies were among the first aimed at the residential sector and sought to help consumers better understand the cost of using electricity in order to decrease their personal contribution to the power grid. These studies were performed in the 1970s and 1980s and realized energy savings of between 7% and 30% [14-19].

Real-time, in-home feedback presents a new technology to aid residential customers in minimizing their energy usage. The first study using real-time direct digital displays was by JK Dobson and JDA Griffin in 1992 [20]. A more aesthetically pleasing and intuitive device led to increased energy savings in later studies [21-22]. The current RTM technologies on the market provide a wide range of options to users, including allowing users to input utility rate structures and receive feedback in the form of numerical and graphical data including kWh consumption and cost of electricity. They present an opportunity to decrease residential energy consumption by 10%-20% [1, 2, 25-29].

Figure 3 shows some of the most common devices in the market today [30-36]. The Aztech In-Home Display offers both cumulative and current kWh consumption and cost data. It is programmed to read the digital signal generated by the new Itron utility meter [37]. This device offers feedback with a 2-4 minute delay and combines statistical, graphical, and visual data on a residence's energy usage. The Power Cost Monitor (PCM) offers instantaneous feedback with a delay of 2-15 seconds. The display offers

both cumulative and current kWh and cost information as well as visual cues on how much power is being used. The fast response allows users to pinpoint the impact of individual device consumption such as microwaves and clothes dryers. Installation requires placing a collar with an optical read on the electrical meter that transmits data to an indoor display unit. The outdoor collar and batteries required for operation may limit wide application of the device.

The EML2020H requires current transformers (CTs) to be placed around the service entrance conductors to the electrical panel. The CTs are hardwired to the display device, causing it to be located directly at the panel. The installation guide recommends a trained electrician to install the device causing the EML2020H to be less user friendly than other monitors. Like the EML2020H, The Energy Detective TED must be installed by a trained professional. This device sends its information through the homes electrical wiring requiring both CTs and a 20A single-pole circuit breaker. The data is offered in the form of current (kW) and cumulative (kWh) usage and alarms can be set when the home is operating past a certain threshold. The response time is one minute or less allowing the user to readily see changes in consumption.

Figure 3: RTM Displays From Left to Right: PCM, Aztech, Wattson, Ambient Orb, Cent-A-Meter [9-10] *(photo courtesy of Josh Eiden)*

The Energy Joule reads an ambient signal, much like a radio station, triggering users to the local utilities demand. This technology requires a quarterly fee of $30 per device/contract to be able to receive the signal. When the utility is peaking, the device will flash red indicating that decreasing usage will reduce the chance of brown-outs and black-outs. Further, the display gives an indication of the current cost of electricity under time-of-use pricing and the user's current energy usage. A device similar to the Energy Joule, the Ambient (energy) Orb, provides only a color change to indicate the utilities demand and could be used to see if a visual cue alone would influence a residence's usage. An information display pilot with Southern California Edison extensively researched the usefulness of the Energy Orb in decreasing residential and commercial electricity demand. The study found that consistent information, feedback and time between the action and the consequence are very important for cause-effect link between

behavior and energy consumption. The Cent-A-Meter uses CTs at the electrical panel connected to a digital transmitter that sends a wireless signal to a battery powered display unit that can be placed anywhere in the home. The information provided includes current kW usage, cost per hour while operating at the current threshold, and a correlation to greenhouse gas emissions generated by the use of electricity. The Wattson is a European device that is designed with one CT to read one phase of the electrical panel. It can be used in the United States with a plug converter. The 'doomsday' function projects how much electricity will cost for an entire year based on the current kilowatts being used.

1.2 RELATIONSHIP OF CHALLENGE TO SUSTAINABILITY

1.2.1 With Respect to People

People will be more satisfied with themselves when they are helping save the planet by saving energy. This will create an environmentally conscious society and make people more willing to reduce energy usage. Reducing energy consumption offers a ready-to-use solution for reducing the effects of climate change and the potential of natural disasters and hazards. This reduction in turn will result in downsizing and less expensive alternative renewable energy sources, such as PV, to supplement or replace the utility grid. This will promote sustainable design, create more job opportunities, and make this planet a safer place to live.

1.2.2 With Respect to Prosperity

The move toward Net-Zero Energy Homes (NZEH), will require new technologies and a paradigm shift. Integrating multiple technologies to provide a simple solution to energy saving will result in optimizing energy consumption without disturbing the convenience of the home resident, but instead saving them money that they can use to invest in renewable energy sources such as photovoltaic panels, or wind turbines. This will save even more money, and will save the environment. Reducing energy consumption will make the grid more reliable and able to operate during maximum demand with a good safety margin, so that a brown out of electricity will be almost improbable and electricity will be always available. This will increase the demand for renewable energy sources and that will lead to a mass production, translating to a price reduction for the consumer and at the same time opening the door for big scale industries and factories to compete for the highest quality and lowest prices.

1.2.3 With Respect to the Planet

The EIA (US Energy Information Administration) projects that between 2006 and 2030, carbon dioxide emissions will increase by 16% while the energy usage will go up by 19% [39]. This increase suggests that many opportunities are available to reduce greenhouse gas emissions. These opportunities include modernizing existing power plants and regulating the production of new plants and energy consumption and savings. **Error! Reference source not found.** provides a detailed view of the contribution of coal lectricity generation to carbon emissions. A reduction in electricity use according to this figure will contribute around 40% in reducing total carbon dioxide emissions.
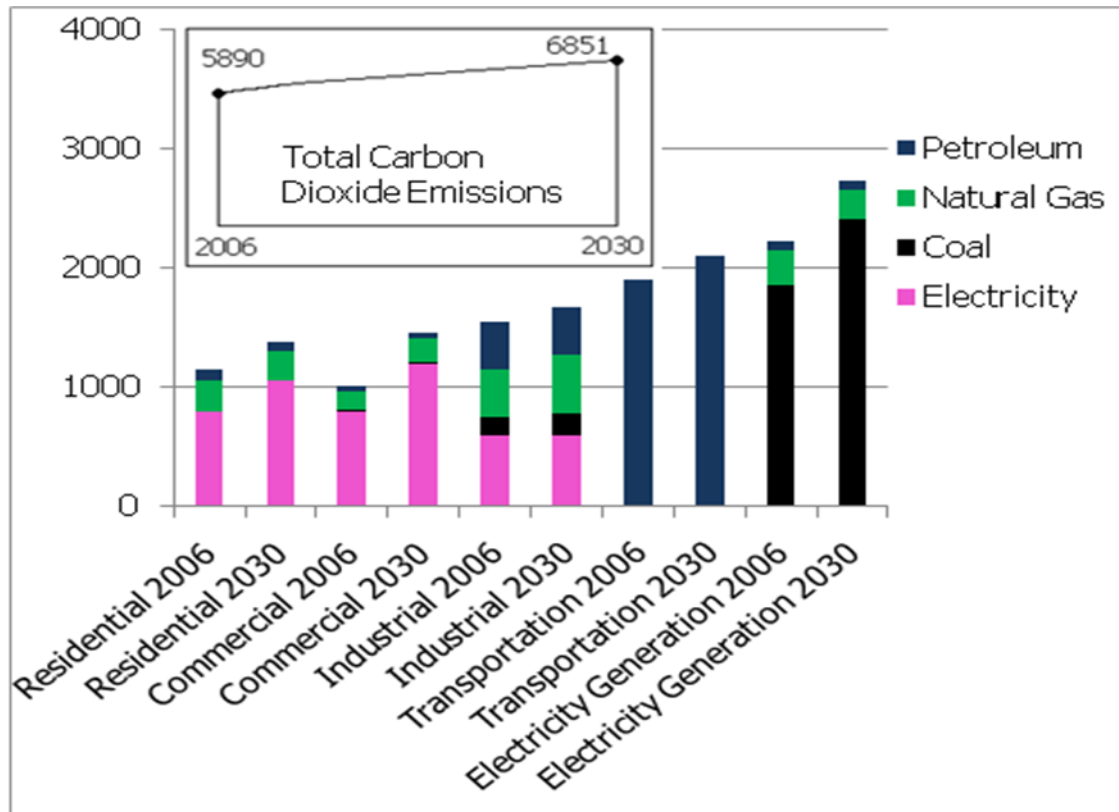
Figure 4: Carbon Dioxide Emissions By Sector and Fuel, 2006 and 2030 (Million Metric Tons) [39]

So energy conservation cannot be achieved unless feedback is provided to the home residents. Current real-time power monitoring devices in the market do not provide detailed information of when, where and how energy is being consumed or what actions to be made in order to shut off the wasted energy. Detailed load-level bottom-up real-time power monitoring systems are needed in order to have a deep understanding of the households' energy consumption so that smart energy management systems can be designed and implemented to achieve energy conservation while maintaining users' convenience. This thesis gives a solution to this lack of such deep information by proposing the ZPM system.

CHAPTER 2

BACKGROUND

Chapter 2 discusses related work, definitions and terms, it also reviews the literature with regards to real time power monitoring. A closer look at the components of the built environment can provide insight into integration between the occupants and those engineered systems of the building that are fulfilling the occupants' needs. Demand response and demand side management are a proven method to save energy for commercial and utility applications. However, this control of energy is a global control within the building, i.e. controlling the mechanical systems by adjusting the overall temperature for the building, and installing occupancy sensors to automatically shut-off the lighting system when no one is present. For this occupancy approach, the shut-off must occur within 30 minutes after the detection of vacancy, defeating the potential short-term savings when someone leaves the room [41].

However, many of these systems become inactive or improperly calibrated that a lack of interest in operation persists. Furthermore, plug-in loads that remain on even when not in use waste energy by design. At the residential level, active research is moving toward smart homes, home automation, and smart meters with demand response integration to shut-off loads by the utility when peak demand is high. This is an intrusive method that could collide with user comfort. However, when the user is empowered to make that decision, the savings are sustained, with an adaptive controller learning user's

energy behavior and determining locations and magnitudes of wasted energy and providing this information to the user to take impulse action to save energy and money at the same time, eliminating the guess work of what else needs to be done to save energy.

To address these issues and provide for a greater savings, a local and global control should be envisioned, modeled, simulated, and tested to make a profound change in how we use energy and how we change our behavior to save energy. The following specific topics are relevant to real-time power monitoring and energy saving, namely, plug-in loads and advanced metering initiatives (AMI), the smart grid and residential energy management, Zero-Net Energy Test Home (ZNETH home), and Building Information Modeling (BIM). With a holistic approach to energy, utilizing these concepts, sustained energy saving and sustainable development become a reality.

Plug-in loads and Advanced Initiative Metering (AIM): The electrical distribution system plays a profound role in the built environment as it is an enabling product, an intangible necessity used to power our systems [42]. According to United States Green Building Council, the built environment in the United States accounts for 72% of electricity consumption, 39% of energy use, 38% of all carbon dioxide (CO_2) emissions, 40% of raw materials use, 30% of waste output (136 million tons annually), and 14% of potable water consumption [43]. To influence the 72% electricity consumption, energy efficiency programs in the U.S. emphasize the use of energy saving technologies and design practices.

As of today, most homes use 20 to 40% more energy than they actually need. Advanced metering initiatives (AMI) are addressing multiple segments in this arena, one of which is that reduction will occur as a result of changing old habits as it is essential to

influence human behavior since when it comes to energy it is the habitual behavior that influences our energy consumption [42]. Research into energy reduction in the residential sector [52-80] has shown that feedback in the form of direct, real-time energy consumption information has resulted in an energy saving in the range of 0-15%. However, current real-time monitoring (RTM) devices available in the market monitor and display overall energy consumption for the entire home only and do not have the capability to adaptively determine wasted energy consumption on behalf of the resident. The current devices rely on the user to know where the energy is being wasted and hope that the user will take actions after seeing the current usage. Hence we need to focus on creative and promising methods to encourage energy efficient lifestyles. However, a 2010 study by ACEE [52] of 57 energy conservation projects from 1970-2010 found that feedback gadgets alone are unlikely to maximize energy savings. A similar study conducted by research team in the University of Nebraska-Lincoln found that combining real-time information with novel control methodologies and state-of-the-art technology to provide and enable information to empower users to save energy instantaneously is the motivation in adaptive impulse behavior saving approach in this proposed research.

The Smart Grid and Residential Energy Management: More sophisticated residential energy management is expected to be a key component of the national smart grid power management and control initiative. Recent commentators [62, 76] describe a future in which physical or virtual in-home monitors (so-called "energy dashboards") provide real-time electricity consumption data to residents who will use this data to actively manage their home power consumption. For example, GridPoint and Tendril

already offer this and both Google (Google Powermeter) [81] and Microsoft (Microsoft Hohm) [82] have announced initiatives to display residential power consumption on personal computers, and many other companies offer a variety of other physical and virtual devices that display aggregate residential electrical power use (at wide ranging prices, from under $100 up to $10,000, others available only through electrical utility pilot projects). These devices are developed and marketed to consumers and electrical utilities with the twin goals of reducing electrical energy consumption (in the case of consumers) and reducing or shifting residential electrical demand (in the case of electrical utilities).

Most of these devices and systems apparently assume that tracking home electricity use will become a national hobby, though the novelty will likely wear off quickly (as informally reported to the authors from industry players). None of the devices or systems targeted to residential consumer use offer automated power management: all require homeowners to manually switch electrical loads, which will limit the effectiveness of these devices and programs, since past research clearly shows that energy savings programs that rely on users to switch off electrical loads do not produce persistent savings [69-71, 83-84]. None of the devices can be expected to produce significant demand savings, since peak demand occurs in the afternoon, during normal working hours, when most residences are vacant, and hence no one is present to respond to switch off unnecessary loads. Persistent electrical energy and demand savings require an automated load switching component that is missing from current solutions. These devices are a good first step towards the development of measurement systems to monitor building energy use (e.g., systems to provide detailed, ongoing information on how

energy is being used within a building), and systems that link embedded intelligence to building control systems.

Zero Net Energy Test House (ZNETH): To foster research and innovation and save energy, the Net-Zero Energy buildings, and specifically Net-Zero Energy Homes (NZEH), are currently being researched and designed by research centers to meet the Department of Energy, Building America, Research Toward Zero Energy Homes [85]. One such house is the Zero Net Energy Test House (ZNETH) currently being built and developed at our campus, UNL's Durham School in Omaha, Nebraska [86]. The ZNETH home is a collaborative research project initiated by the University of Nebraska Peter Kiewit Institute, USGBC Flatwater Chapter, and the Green Omaha Coalition. The ZNETH research project will investigate effective approaches to achieve whole-house environmental and energy goals. This project will also investigate design and construction strategies leading to zero net energy homes as the basis for analyzing future residential and commercial performance. The house is designed as a living laboratory for sustainable research and education taking place in the Architectural Engineering program, to integrate environmental/energy innovations and onsite/renewable power solutions that can be used to achieve a zero net energy building and to receive a USGBC Leader in Energy and Environmental Design (LEED) Platinum certification [87]. The house is about 2,800 square feet. It contains four bedrooms and four bathrooms on two floors and a basement. Three graduate students will live in the house when it is completed. As a living laboratory, the house will be the ideal setting for the modeling simulation, verification, and validation of energy management system and home automation.

Building Information Modeling (BIM): BIM is a digital representation of physical and functional characteristics of a facility creating a shared knowledge resource for information about it forming a reliable basis for decisions during its life cycle, from earliest conception to demolition [88]. BIM software involves creating a virtual model that mimics a real building that if created correctly, will contain all the information necessary to answer any query. It is often a "3-D Model", with building-like parts of a model that mimic the behavior of the real world [89]. BIM is a tool that is expected to be used and provide the opportunity to build a facility virtually, predict and monitor energy performance to reduce energy usage, examine material sustainability used in the facility over its lifecycle; and improve environmental impact of the facility–internally and externally and over the lifecycle including disposal (building Smart alliance).To the best of our knowledge, a complete model for a comprehensive electrical system and its components does not exist in BIM. Further, BIM model works off-line [90] and does not yet show online information of, for example, how much energy is being consumed at a specific moment, in a certain load, or in a specific room. With this work we are advancing the knowledge and tools to revolutionize BIM as online tool for real-time energy monitoring, location, identification, diagnostic, and efficiency enhancement to the built environment.

The majority of research on energy monitoring to date focuses on the pre-operative design and work flow as opposed to post-design monitoring. In recent years, several key energy monitoring systems have been proposed. The authors of [91] introduce "QUEEN", a real-time power monitoring tool for designing sustainable

buildings that perform efficiently through optimization of the HVAC system. A large database is amassed in order to enable a dynamic simulation. Google SketchUp is also utilized to further define the 3-D building model. Another study proposes the Cluster Energy Management System (CEMS), a real-time power monitoring system able to optimize the energy usage of a zone by maintaining a balance between the supply and demand sides.

The system is unique in that it links several home energy management systems (HEMS) together. CEMS was created using the Keio University Network Oriented Intelligent and Versatile Energy Saving System (KNIVES), a demand side management system that controls electronic devices based on co-operative and distributed control algorithms and real-time measurement variables [92]. In [93], the authors integrate the KNIVES system with a smart circuit box to predict residential power needs.

In another energy monitoring system to achieve automated demand respond in microgirds, presented in [94], a plug-and-play microgrid is proposed for the electrification of off-grid regions. Dynamic Time Warping (DTW) and Hidden Markov Models (HMM) are analyzed in terms of how effectively they can determine which loads to reduce in situations in which demand exceeds the generation capacity. The authors conclude that the HMM is more effective since the usage behavior of the device affects its ability to recognize loads. Much previous research has also been devoted to improving ways to inform residents of their energy usage. The authors of [95], for example, created the General Packet Radio Service (GPRS), a wireless sensor network that controls and monitors the status of home appliances and alerts occupants to potential danger through instant messaging. In another study, a Zigbee wireless network was built

that both tracked occupants in real-time and provided on/off control of household devices. Other studies have focused on automating devices with SMS (Short Message Service) and GSM (Global System for Mobile) technologies [96-97] and controlling outlets with Bluetooth modules [98].

CHAPTER 3

DESIGN AND METHODOLOGY

The field of information technology is developing so rapidly that it is not uncommon for new inventions, ideas, and upgrades to be introduced at a nearly daily pace. Architectural Engineering is just one discipline rife with new advancements in the area of information technology. A prime example is the development of the Building Information Modeling (BIM) Model. The BIM Model works off-line and offers information of a building's components and elements. Notably, the current BIM Model does not yet have the capacity to provide online information detailing, for example, the amount of energy consumed at a specific moment, at an exact location, or for a particular load. The ZNETH's Power Monitor (ZPM system) is therefore proposed as a solution to the BIM Model's lack of real-time data. The ZPM is expected to be a powerful tool complete with detailed energy-consumption databases and load profiles to give engineers the data they need for a range of endeavors, such as developing intelligent energy management systems, creating learning environments for non-intrusive load monitoring systems, and analyzing the impact of informing users of their location based energy consumption rates.

The ZPM system is a complete hardware/physical and software/virtual system intended to highlight energy consumption of home residents in such a way that is very easy to understand and take appropriate actions (by manually plugging on/off loads or

dimming them). ZPM provides the user with a simple computer interface where they see an actual off-line picture(s)/layout(s) of their building along with their power consumption locations highlighted by means of red circles. The thickness of the circle depends on the power being consumed and the maximum power consumed in the whole building. In the hardware/physical sub-system each load is provided with a power sensor to measure the power parameters. The software, in addition to highlighting the power consumption locations/loads by power-dependent variant-thickness red circles, also shows power consumption percentage, instant power consumption, power consumption over a period of time. One glance at the computer screen gives a complete picture of where energy is being used/wasted, which load is consuming too much compared to others, a history of power consumption, and many other features/capabilities being integrated into the ZPM system.

The proposed ZPM system involves physical/hardware and virtual/software systems linked by way of a data communication link. The physical/hardware sub-system will be referred to as the Real-Time Power Monitoring System and will be abbreviated as RTPM. Fig. 5 is an illustration of the components of the ZPM system. Each load is provided with a sensor for measuring the power parameters and communicating these measurements to a centralized unit in the building. The virtual system is composed of two subsystems: the virtual building and the data base. The virtual building is a program written used a visual programming language using a picture of the building or its rooms/zones as a background for displaying power consumption information on the load location. The program communicates and exchanges data with the RTPM system via a serial port, a USB port, or any other available computer port. The data base can be a

Microsoft Access file, Microsoft Excel file, text file or any designed data base

architecture. If the physical/hardware system and virtual/software system are in the same

location (i.e. the RTPM system and the computer containing the virtual/software program

are located in the same building), then a serial port cable is used as the data

communication link. Online access can be achieved by implementing the data link over

the Internet as a wireless link with two modems at each end, by way of traditional phone

cables and phone modems, or by utilizing any other data communication technologies.

The RTPM system will be described in terms of its application to a residential home in

order to better illustrate its specific design features and capabilities.
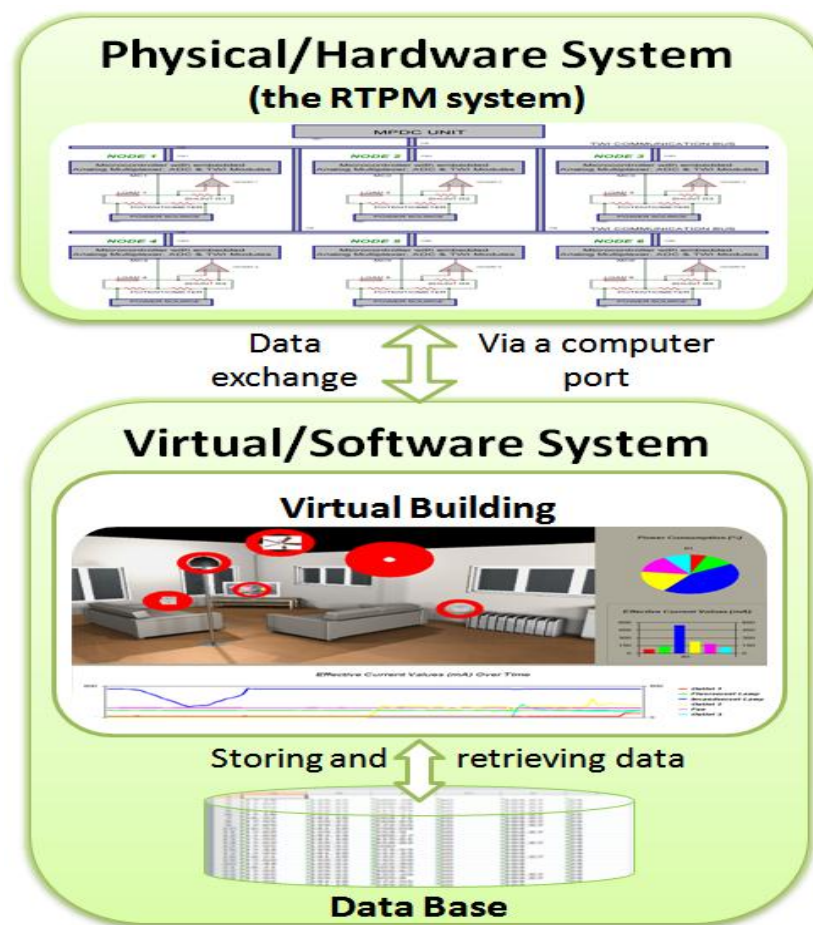
Figure 5: The ZPTM System

3.1 RPTM SYSTEM DESCRIPTION

In order to achieve the load-level monitoring of power consumption a Measurement, Processing and Communication Unit (MPCU) is installed at each load location. The load is any type of fixed loads such as a lighting device, an air conditioner unit, a refrigerator, fan; or it can be an outlet where power-consuming devices are plugged in and out. The load along with its associate Measurement and Communication Unit all together comprise a Node. For each building there is one MPDC (Main Processing, Display and Communication) Unit which communicates with the Nodes in a two-way manner to exchange data in real time. The MPDC Unit communicates also to a computer where the virtual environment exists. the proposed RTPM system then consists of the MPDC Unit and the Nodes. Fig. 6 illustrates the RTPM system structure and components.

Figure 6: RPTM System Architecture

Each Node is given a unique address which is used as its identity to distinguish it from the other Nodes. The MPDC Unit plays the role of the Master in terms of data communication, while the Nodes play as the slaves. Data communication is always initiated and terminated by the MPDC Unit.

Measurements of the main power consumption parameters, such as the effective current and voltage values, are taken place at the Measurement and Communication Units of the Nodes. Nodes are provided with current and voltage transducers which generate si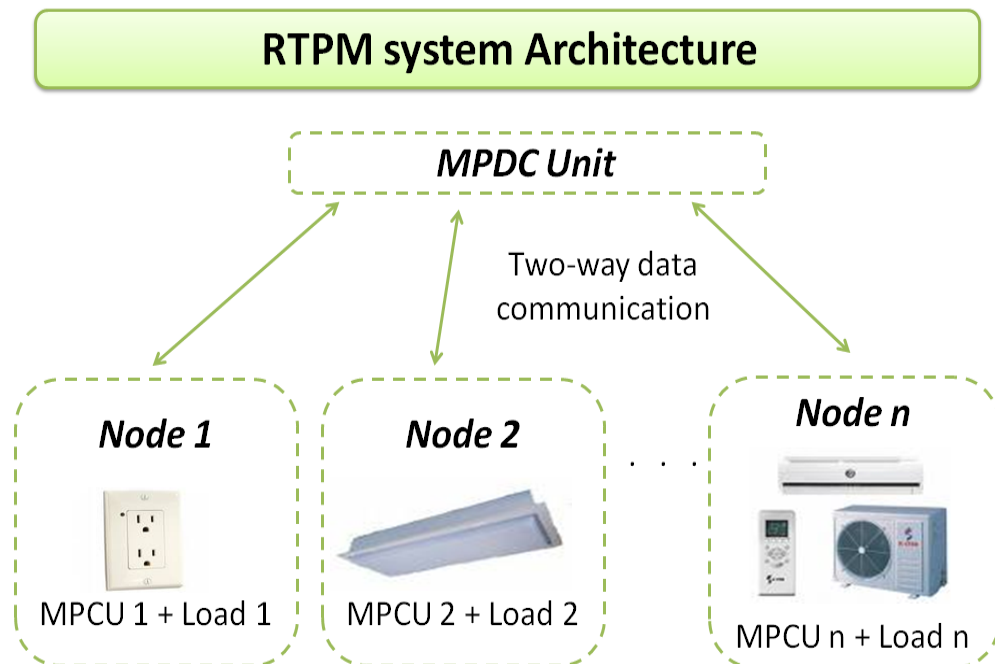gnals proportional to the original waveforms to be the inputs of Analog to Digital Converters (ADC Modules). The output of the ADC Modules is the digital values which are processed afterwards to calculate the main power consumptions parameters. More power consumption characteristics are extracted by saving the samples of the actual voltage and current waveforms at each Node, and then sending these samples to the MPDC Unit which, in its turn, sends this huge amount of data to the computer for more complex processes such as extracting the harmonics, the power factor, and using these waveforms for load recognition and load forecasting.

Display Unit can be a character LCD displaying, for example, the current power consumption of the whole building and/or some/all the loads, the current bill, the percentage of today's power consumption of the building compared to the building's average power consumption and so on and so forth. The Display Unit can also be a graphical LCD showing the information just mentioned above in more interesting and meaningful way than numbers using graphs, columns, percentage pies etc. It can also be a

special lights convention resembling, for instance, the traffic lights where red is used to warn users of extreme power consumption and to indicate that an urgent power reduction action is highly recommended, whereas green expresses that the building power consumption is still within its minimal possible range.

The data communication between the MPDC Unit and the Nodes can be wired or wireless. For wired option, and for research/experimental purposes, many choices are possible such as Serial Communication Interface (SCI), as utilizing Universal Synchronous Asynchronous Receiver Transmitter (USART) Modules as the communication units for the MPDC Unit and the Nodes. The Communication Units can also be implemented as Two Wire serial Interface (TWI) Modules, also known as I2C Modules. More reliable wired message-based protocols can be utilized such as Controller-Area Network (CAN) which was basically designated for automotive applications, nevertheless nowadays it is widely used as the means of data communication between/within medical and industrial devices. The data communication can also be implemented wirelessly utilizing the state-of-the art and nowadays standards for wireless mesh networks for building automation which is Zigbee protocol. The trade-off between the cost and the reliability of data communication is not the focus of this paper though. The Data Processing unit of the MPDC Unit is a processor or a microcontroller programmed to do the tasks involving managing data communication, displaying data, and communicating the data to the computer.

To demonstrate the RTPM system and its design, a prototype RTPM system is built and tested. In this prototype there are 6 Nodes and one MPDC Unit. Atmega16 microcontrollers are used to make use of its embedded modules to work as the

components of the MPDC Unit and the Nodes since these modules are already embedded in the chips. In this prototype the MPDC Unit is composed of a character LCD as the Display Unit, TWI Module embedded in the microcontroller as the Data Communication Unit, and the microcontroller itself as the Processing Unit. Current and voltage transducers, operational amplifiers, Analog to Digital Convertor (ADC) and TWI Modules embedded in the microcontroller and the microcontroller itself comprise the MPCU Units of the Nodes for the RTPM prototype system. Six different loads are installed on the prototype board which was built by Josh Eiden and provided by the lighting lab of the Architectural Engineering department of the University of Nebraska-Lincoln. These loads are: 1- Outlet 1. 2- fluorescent lamp. 3- Incandescent lamp. 4- Outlet 2. 5- Fan. 6- Outlet 3. Three of these loads- the fluorescent lamp, the incandescent lamp and the fan- are fixed loads. Whereas different types of power-consuming devices can be plugged in and out of the outlets. A board with these six loads is built and shown in figure 7.
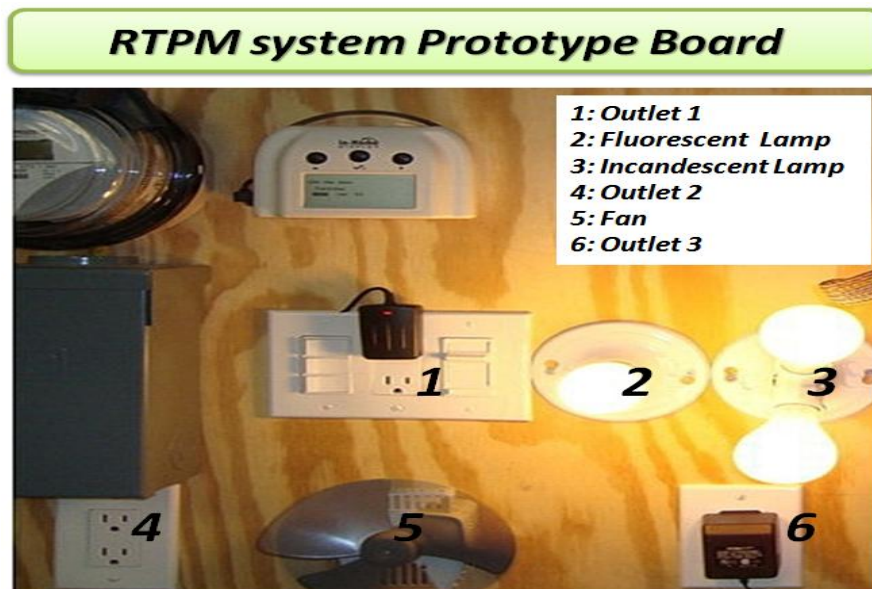
Figure 7: RPTM System Prototype Board with a Unique Load ID

The Measurement sub-Unit in the MPCU Unit of each Node is composed of voltage transducer, which is a potentiometer, current transducer, which is a shunt resistor. The voltage signal generated from the current transducer is the input of the operational amplifier (opamp) which adjusts the signal to be suited to be an input to the Analog to Digital Convertor (ADC Module) embedded in the microcontrollers. The digital values, which are the output of the ADC Modules are processed then by the software installed in the microcontroller to calculate the current and voltage effective values. Fig. 8 illustrates the RTPM system prototype and its main components. The circuit diagram of the RTPM system is also illustrated in Fig. 9.

The MPDC Unit scans all the Nodes beginning from the first Node to the last Node. It requests information one Node at a time, waits for a response, receives the data if the Node responds, shows the data on the display unit, which is a character LCD, sends this data to the computer and then moves forward to the next Node. After all the Nodes are scanned and the data is gathered, the whole process starts all over again. Numbers are used as IDs to distinguish the Nodes from each other. For the RTPM prototype these Node IDs start from number 1 and ends with number 6. The USART Module embedded in the MPDC Unit Processing Unit (i.e. the microcontroller is the media used for sending data to the computer. This flowchart is illustrated in Fig. 10.
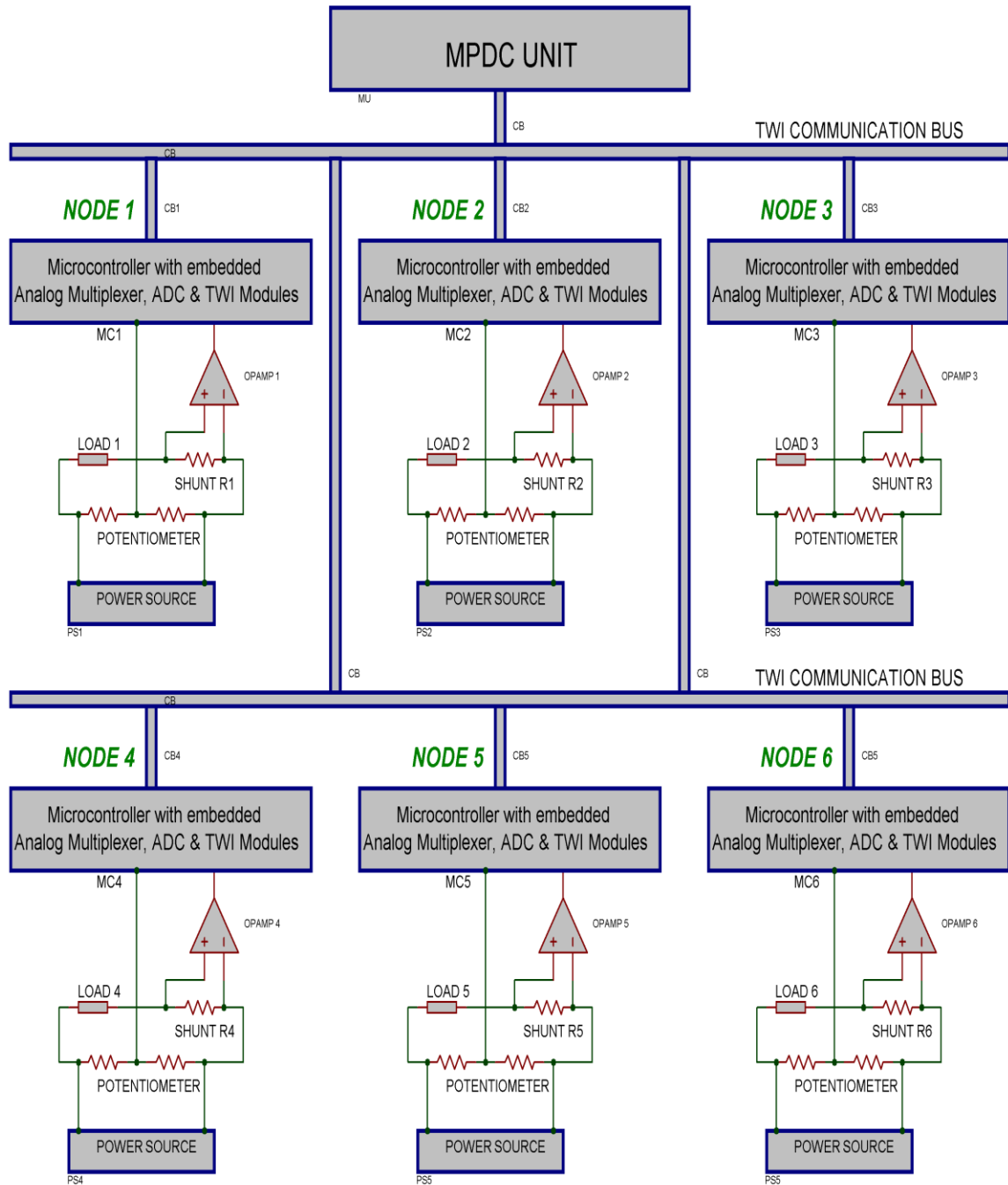
Figure 8: Block Diagram of RPTM System Prototype
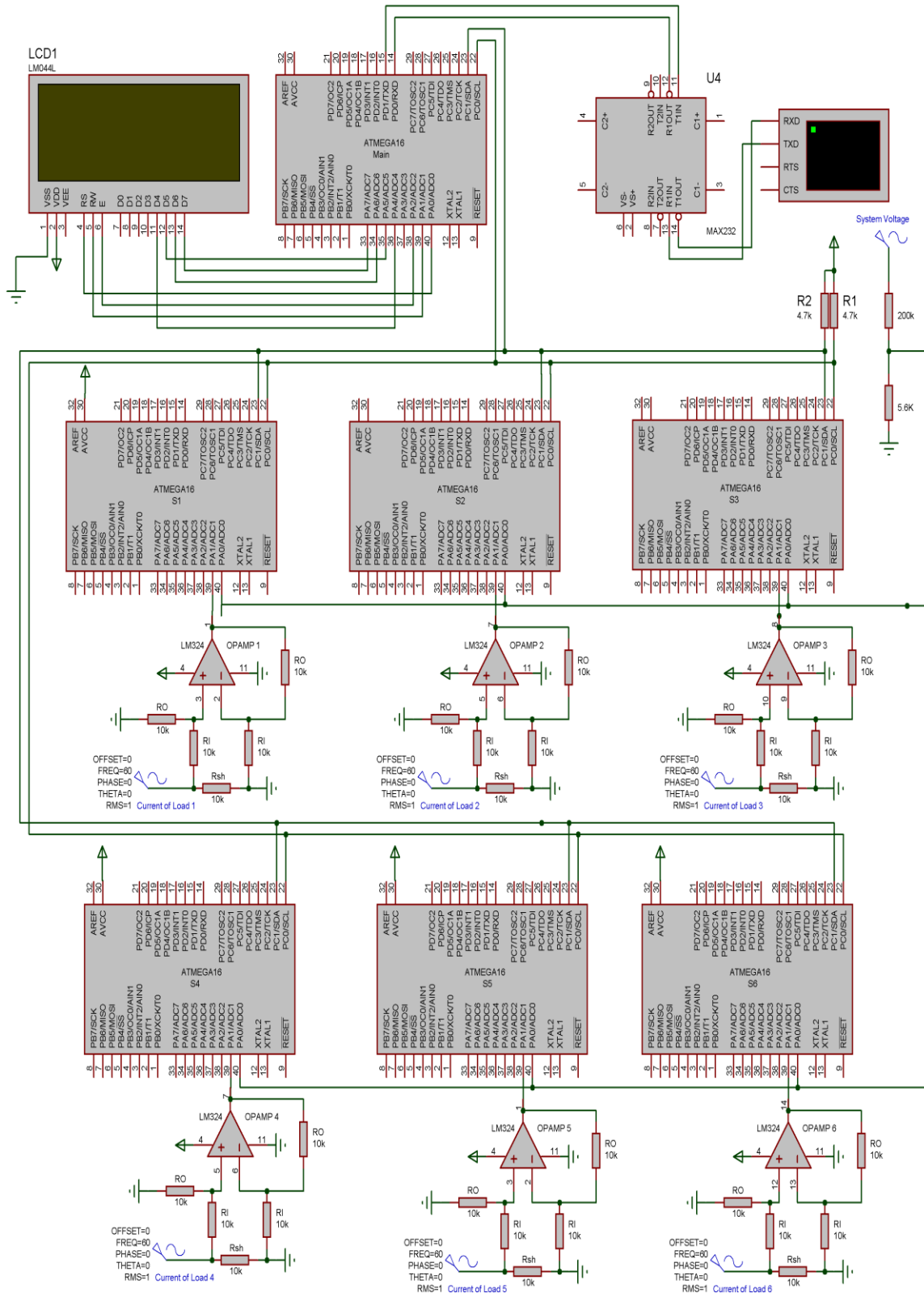
Figure 9: Circuit Diagram of RPTM System Prototype
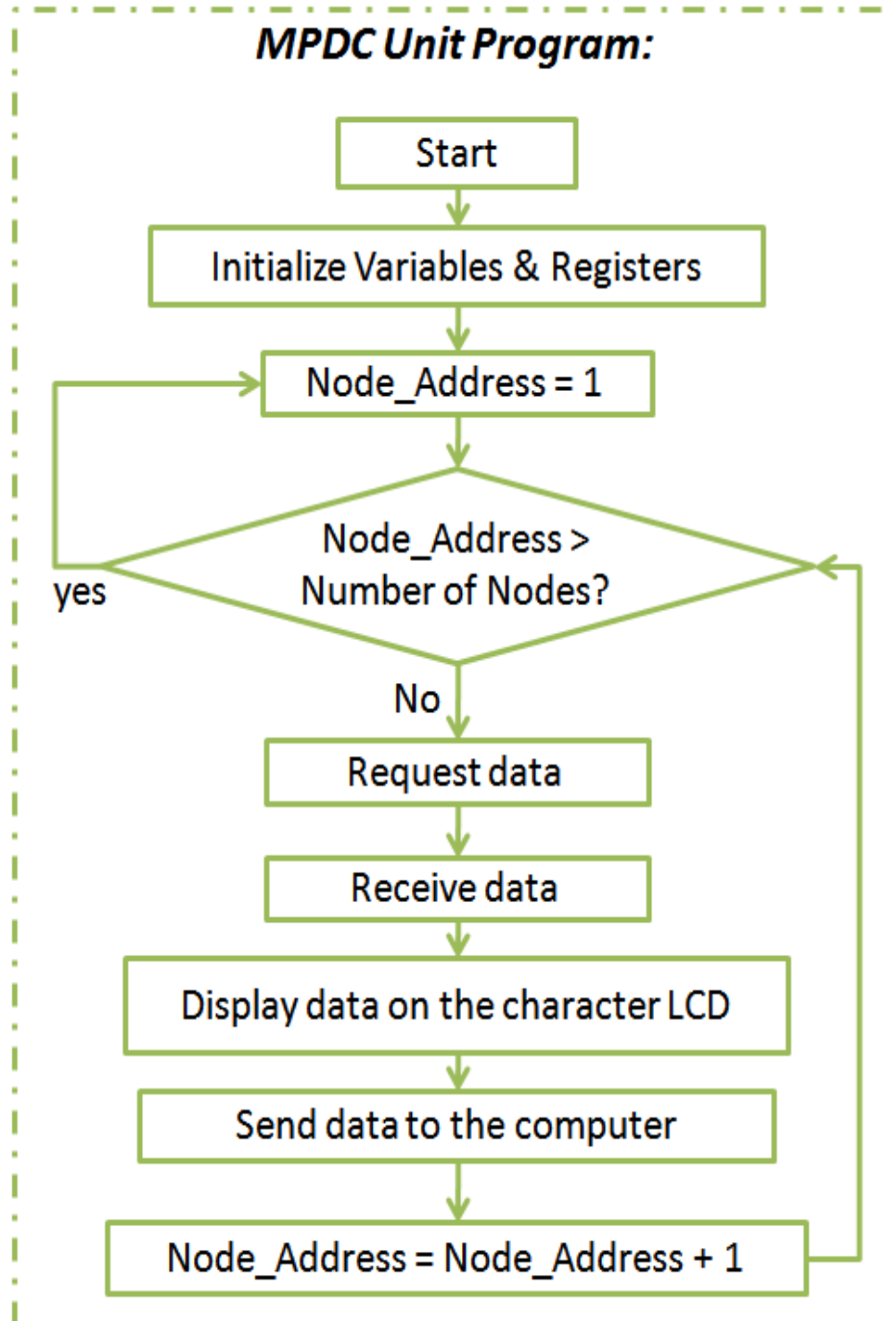
Figure 10: MPDC Unit Program Flowchart

The Node software architecture is composed of two types of routines, the main routine which keeps running and executing over time, the second type of routine is the interrupt based routines. These interrupt-based routines execute only when an interrupt occurs. For the Node software there are two types of interrupt: the ADC analog to digital conversion complete interrupt, abbreviated as the ADC Interrupt, and the TWI Receive Interrupt. The ADC Interrupt occurs when the conversion of one sample of an analog value into digital value is complete. The TWI Receive Interrupt occurs when the data is requested from the Node by the MPDC Unit. After the Interrupt Service Routine is executed the Main Routine continues executing from the point where it branched out to execute the Interrupt Service Routine. Fig. 11 describes the Node software architecture.
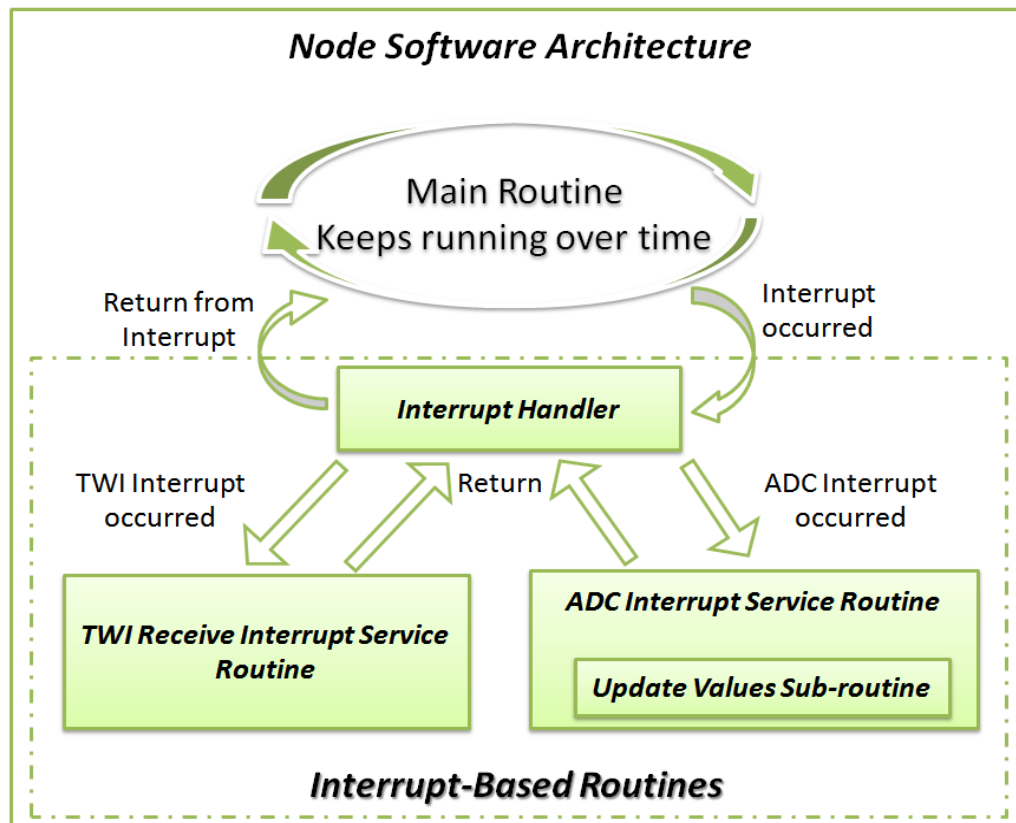
Figure 11: Node Software Architecture

When the MPDC Unit requests data from the Node, the Main Routine of the Node stops executing and the program jumps to execute the TWI Receive Interrupt Service Routine. When data is available in the TWI receive buffer, the data is read and then parsed to check if the MPDC Unit is requesting data from the Node. If the Node has been addressed by the MPDC Unit the Node Main Routine is informed by raising a data request flag as an indicator of this event. The Node Main Routine checks the data request flag continuously to take the appropriate action if the Node is addressed. Fig. 12 shows the TWI Interrupt Service Routine.
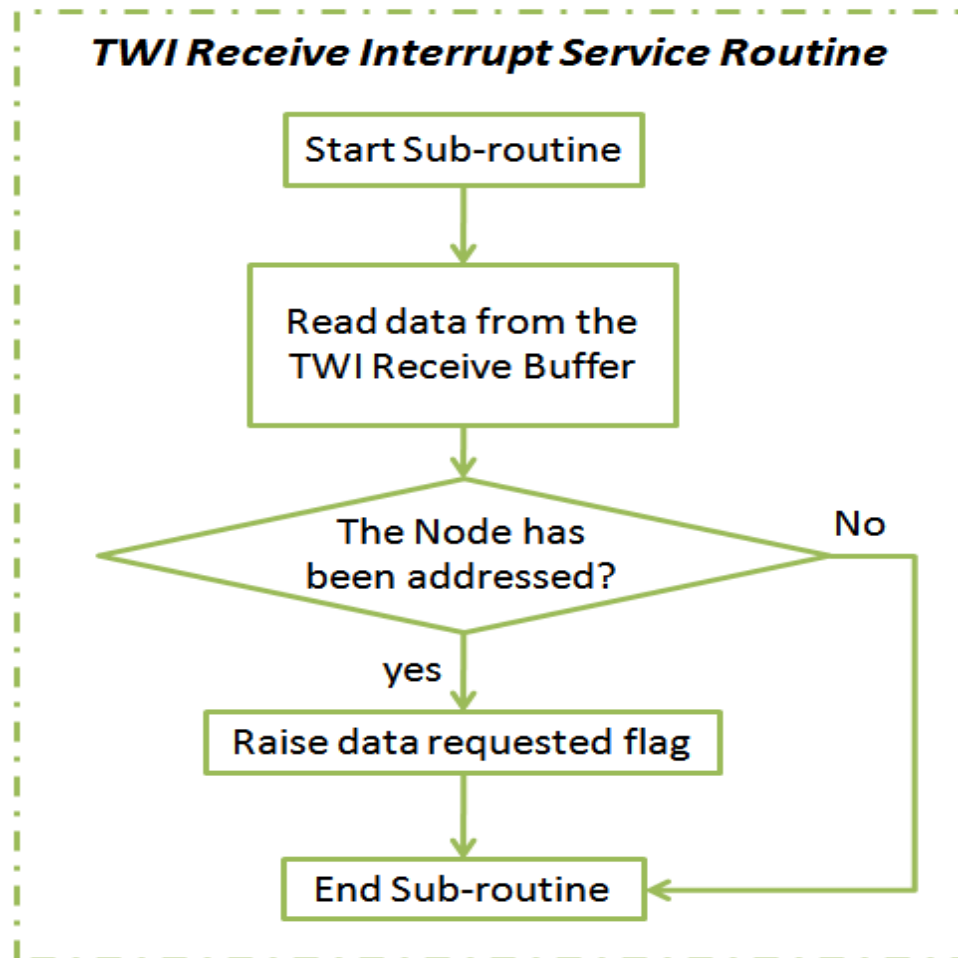
Figure 12: TWI Receive Interrupt Service Routine

The Node Main Routine manages all the tasks done by the Node. As illustrated in Fig. 13 the Node Main Routine triggers the sampling process of the voltage and the current waveforms. Once the sampling is complete, the effective values of the current and the voltage are calculated based on variables being updated during the sampling process (which are Vsq & Isq) and based on (1) and (2):

$$I = k * \sqrt{\frac{I_{sq}*dt}{T}} \qquad (1)$$

$$V = k * \sqrt{\frac{V_{sq}*dt}{T}} \qquad (2)$$

Where:

I: The effective (rms) value of the current

V: The effective (rms) value of the voltage

I_{sq}: The sum of the current squared sampled values over a period T (T = 1/F)

V_{sq}: The sum of the current squared sampled values over a period T (T = 1/F)

F: The voltage frequency (F = 60 Hz)

dt: The time (in micro seconds) between two samples of the same channel (i.e. the voltage channel or the current channel)

k: the ADC conversion ratio. For the last assumptions in (3):

$$k = \frac{V_{ref}}{V_{max}} \qquad (3)$$

V_{ref}: The reference voltage for the ADC conversion.

$V_{ref} = 5$ V is used

V_{max}: The maximum digital value for the sampled value. For 10-bit ADC module $V_{max} =$ 1023. Then k = 5V/1023

    Vsq and Isq are calculated by the Update Values Sub-Routine flowcharted in Fig. 14 and based on (4) and (5):

$$I_{sq} = \sum_{i=1}^{N} S_v^2 \qquad (4)$$

$$V_{sq} = \sum_{i=1}^{N} S_v^2 \qquad (5)$$

Where:

S_v: the sampled value (for either the current or the voltage)

N: the number of samples over the period T

S_v: the sampled value (for either the current or the voltage)

$$N = \frac{1000,000}{dt*F} \qquad (6)$$

    Every time the ADC conversion from analog to digital is complete the ADC Interrupt Service Routine is called. It starts by checking if the sampling over the entire waveform is complete; if not, the Update Values Sub-Routine illustrated in Fig. 14 is called to update the values of Isq and Vsq according to (7) and (8):

$$I_{sq} = I_{sq} + S_v^2 \qquad (7)$$

$$V_{sq} = V_{sq} + S_v^2 \qquad (8)$$

Once the Number of Samples equals the number required to scan the entire waveform of the voltage or the current, the Node Main Routine is informed of that by raising the Sampling Done flag as an indicator that the sampling is done. The Node Main Routine checks the Sampling Done flag continuously to take the appropriate action described in the flowchart in Fig. 13.
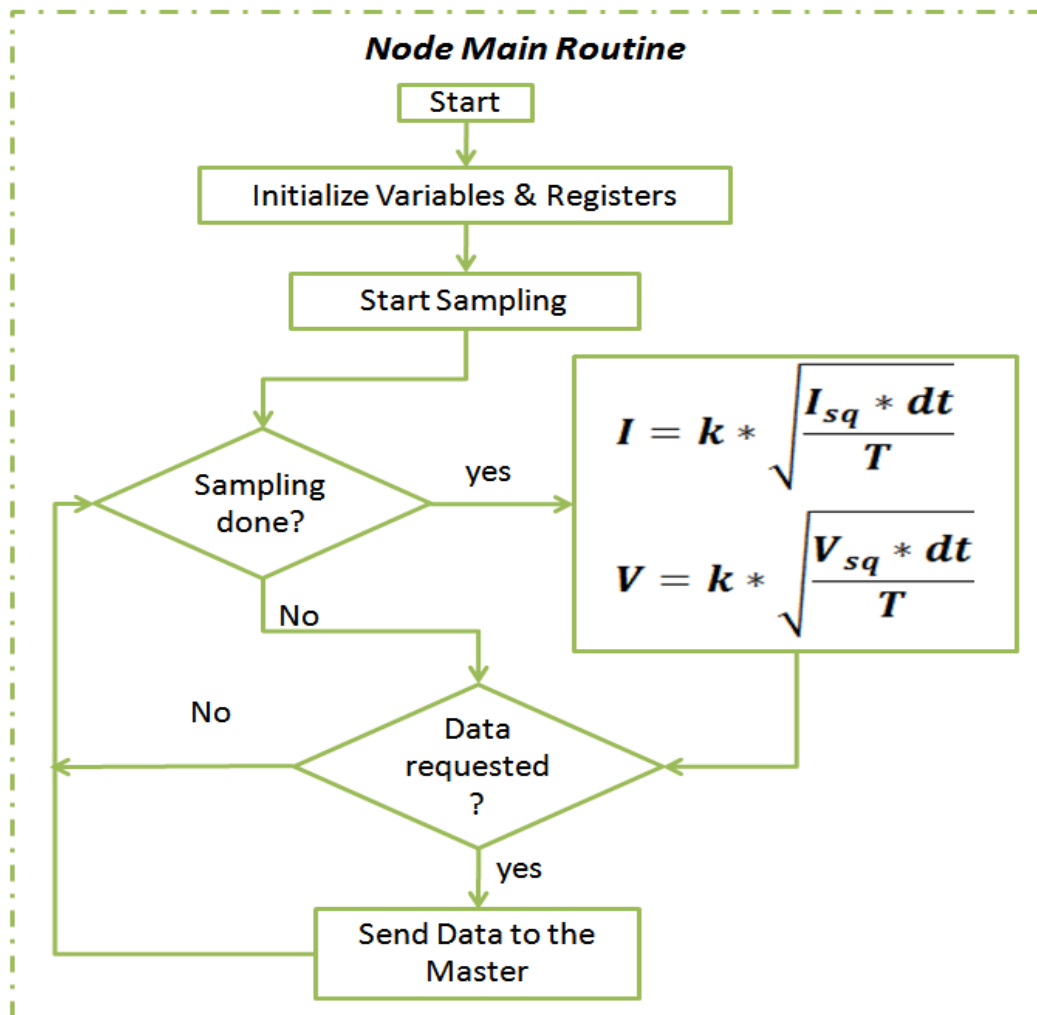
Figure 13: Node Main Routine
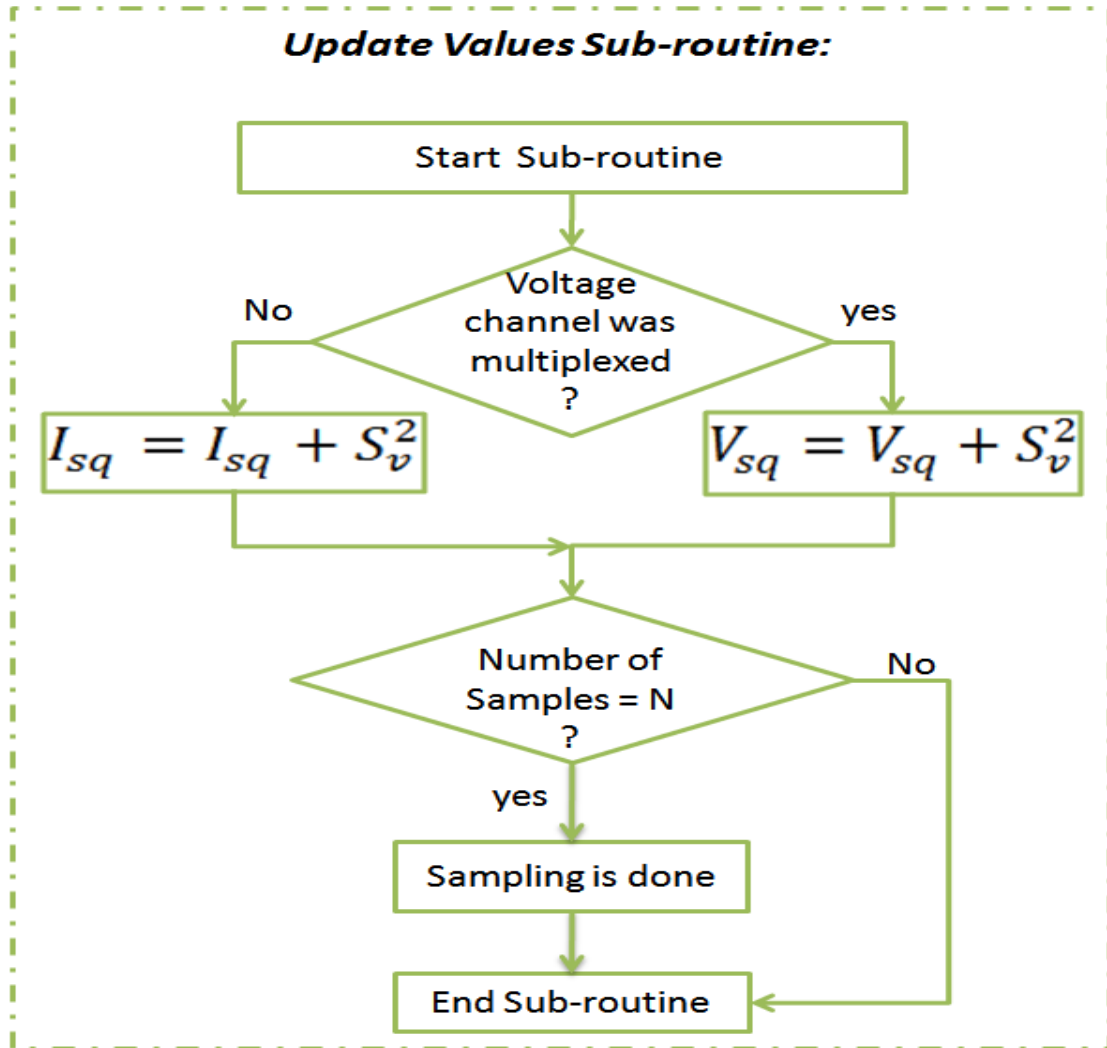
Figure 14: Update Values Sub-Routine

After updating one of these two values, and according to which channel was multiplexed, the other channel is multiplexed for the next ADC conversion to take place. The voltage and the current waveforms are sampled on turns, one at a time using only one embedded ADC module and embedded Analog Multiplexer to choose one the voltage and the current channels every time the ADC Interrupt Service Routine is recalled. This is depicted in Fig. 15.
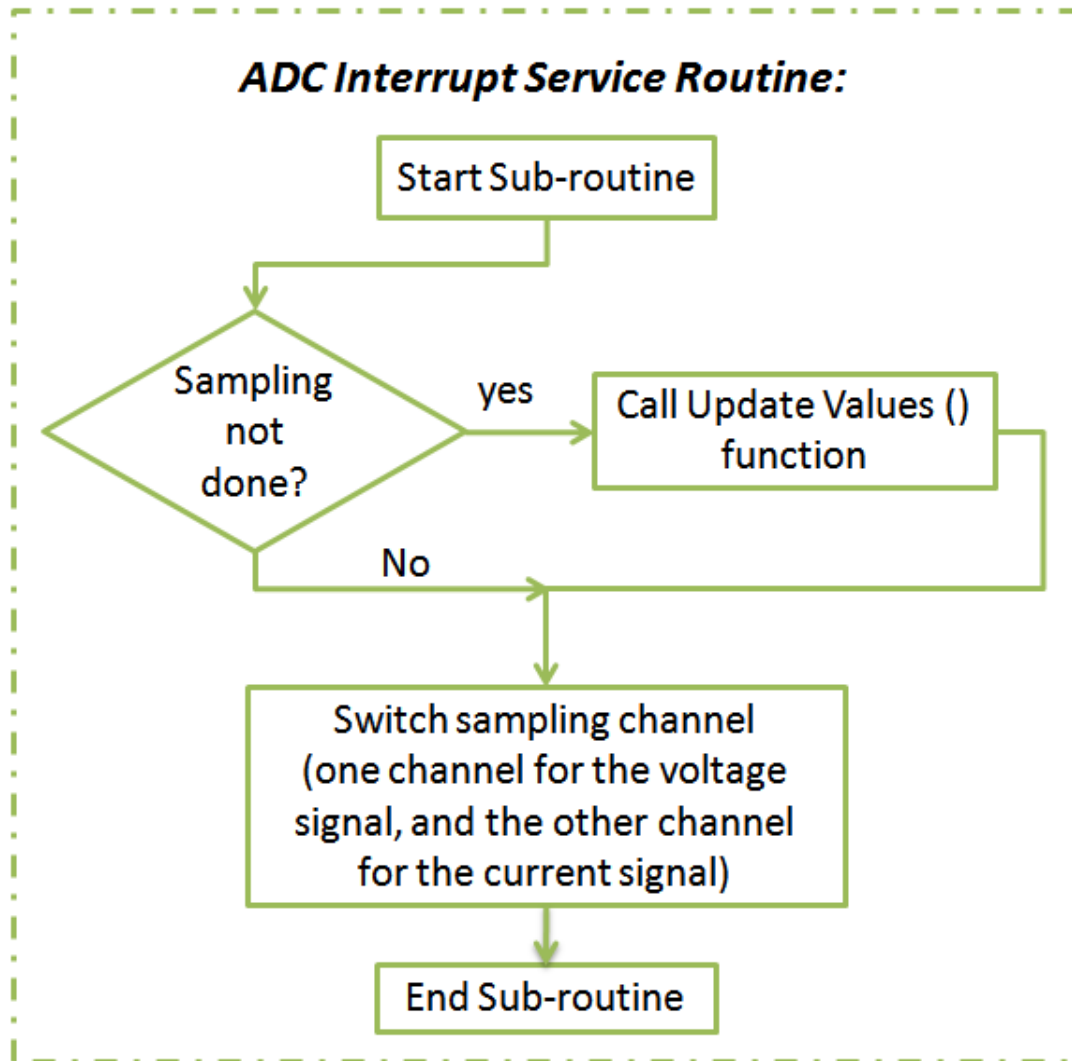
Figure 15: ADC Interrupt Service Routine

3.2 THE VIRTUAL SOFTWARE SYSTEM

The virtual system was built using Visual Basic. It provides the user with a simple computer interface that gives an actual off-line picture/layout of their building along with their power consumption locations. The effective values of the currents over a period of time and instant power consumption of each load are displayed as both a percentage of the total power consumed and as a column representing the instant effective current value. A picture of one of the rooms in ZNETH Home is used as a background to show the real-time power consumption information. Each load on this room is associated with one on the RTPM system prototype. When power is consumed at a load location a red circle with a thickness proportional to the power consumption of that load will show up indicating which load is consuming power at the moment. This way to present information is very user friendly and easy to understand since it gives a big picture of where energy is being consumed/wasted. Fig. 16 shows the virtual/software system displaying real-time power information with the curves of the effective values of the six loads currents on the same graph. In Fig. 16 the Effective Current Values Over Time curve shows how the loads in this room where turned on in the following sequence: The Incandescent lamp, the fluorescent lamp, the fan, outlet 1's plugged in load, outlet 3 plugged in load and finally outlet 2's plugged-in load. On the other hand, Fig 16 shows how the room's loads where turned off in the following sequence: the incandescent lamp was dimmed and then turned off, the fluorescent lamp, the fan, outlet 3's load was turned off, outlet 2's load was turned off, and finally outlet 1's load was left turned on.
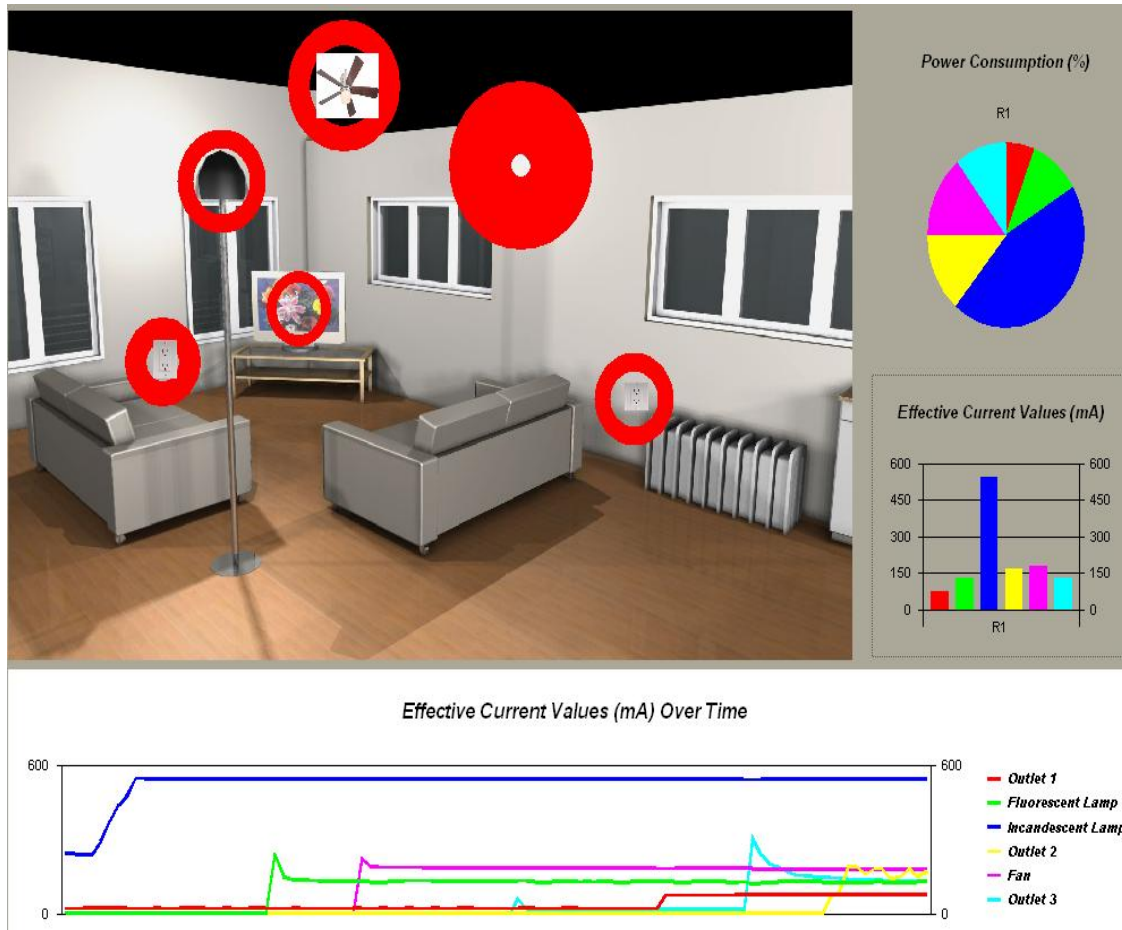
Figure 16: The Virtual System Displaying Real-Time Power Consumption Information and a Short History of All Loads Displayed in the Graph *(background photo courtesy of Dr. Yong Cho)*

Clicking on one the graphs allows users to access additional information about the loads, such as the load type (i.e. outlet, flourecent lamp, incandescent lamp, fan ..), load ID (i.e. a unique number/code identifying each load), and the power consumption parameters (the effective value of the voltage and current). Figs. 15, 16, and 17 are screenshots of the system displaying this information. The virtual board is conncected to a database built using the Microsoft Excel program for storing and retrieving data.
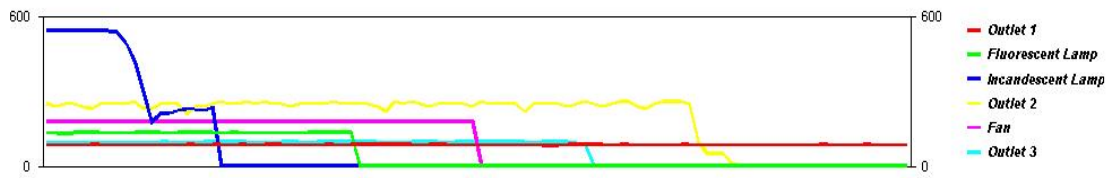
Figure 17: Virtual System Display After Turning Off Five Loads
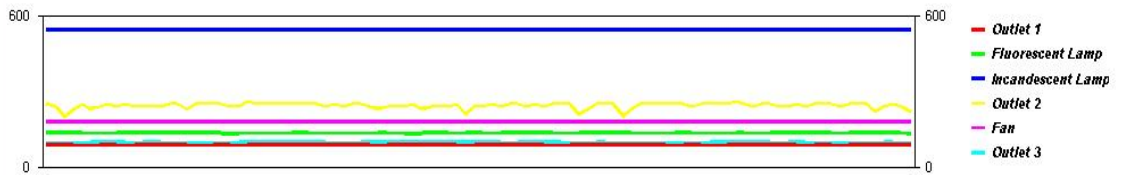
Figure 18: Virtual System Displaying Real-Time Power Consumption Values

3.3 THE COMPLETE ZPM SYSTEM

The physical board and virtual environment are connected to each other via a computer serial port. Real time power parameters are displayed simultaneously in both the physical and virtual environments. Fig. 19 shows the ZPM system in operation. Instant power information is displayed on the character LCD, breadboards, and computer screen.
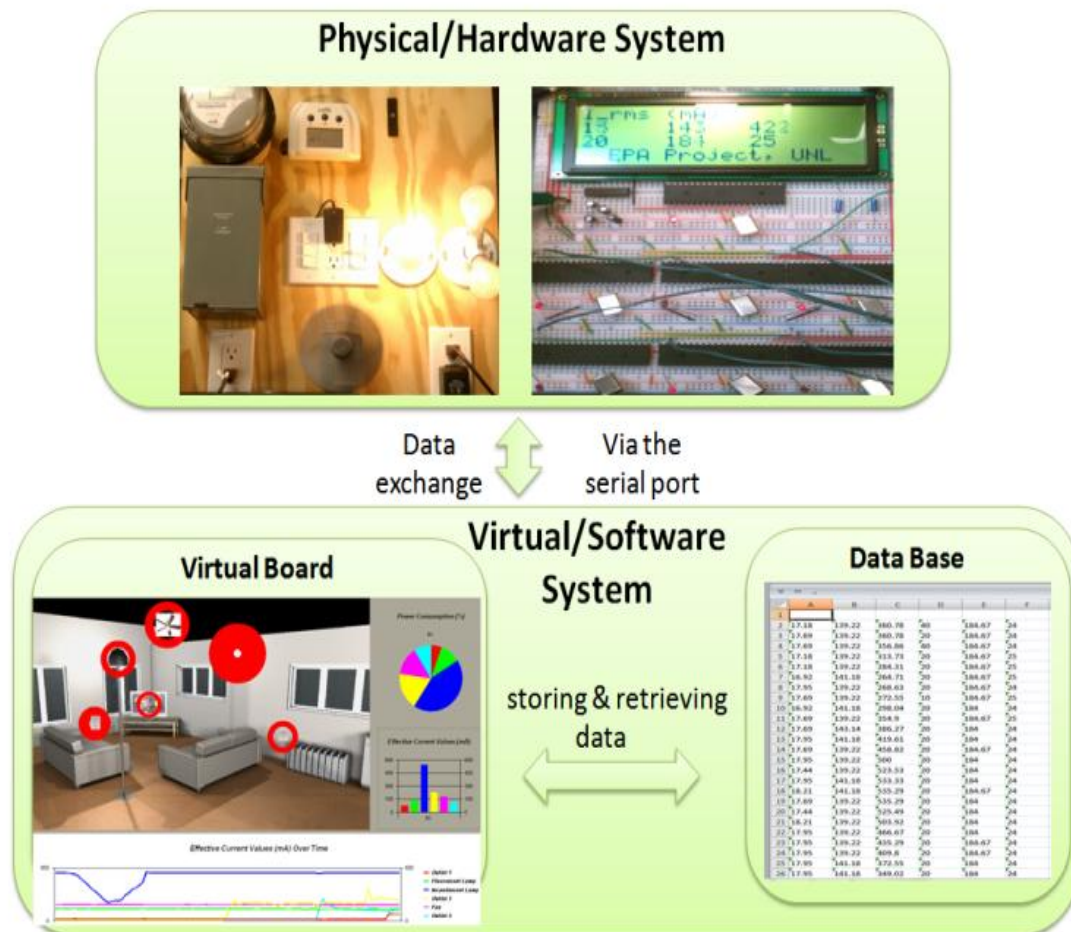
Figure 19: Overview of the Physical and Virtual Systems Connected to Each Other and Giving the Information Instantly and in Real-Time

Fig. 20 is a screenshot of the virtual environment showing excess energy consumption at two locations. Two loads are plugged into the two outlets displayed in Fig 19. Although these two loads are turned off, they still consume power as a phantom load. In the same way a red stop sign warns drivers to "stop", the red indicators function as warning indicators cautioning users to stop their energy waste.

Another virtual model using the Revit Program is currently being investigated as another potential virtual tool to enable engineers to monitor systems already in operation. A variety of programs can be integrated to work in the virtual environment, such as the Revit program. This integration may be especially beneficial to engineers working together on multidisciplinary projects. However, an independent virtual environment can still deliver the basic needs in terms of real-time power monitoring and be easier to understand to home residents. Following this developement the system will be tested in the Zero-Net Energy Test Home (ZNETH Home) [10], an energy sustanible house/lab owned by the University of Nebraska-Lincoln (UNL) and built to support "green" research within the Architectural Engineering Department. The outcome of this study will determine the energy saving capacity of the proposed ZPM system and provide a more accurate assessment of its potential for industry sponsorship and real-world implementation.
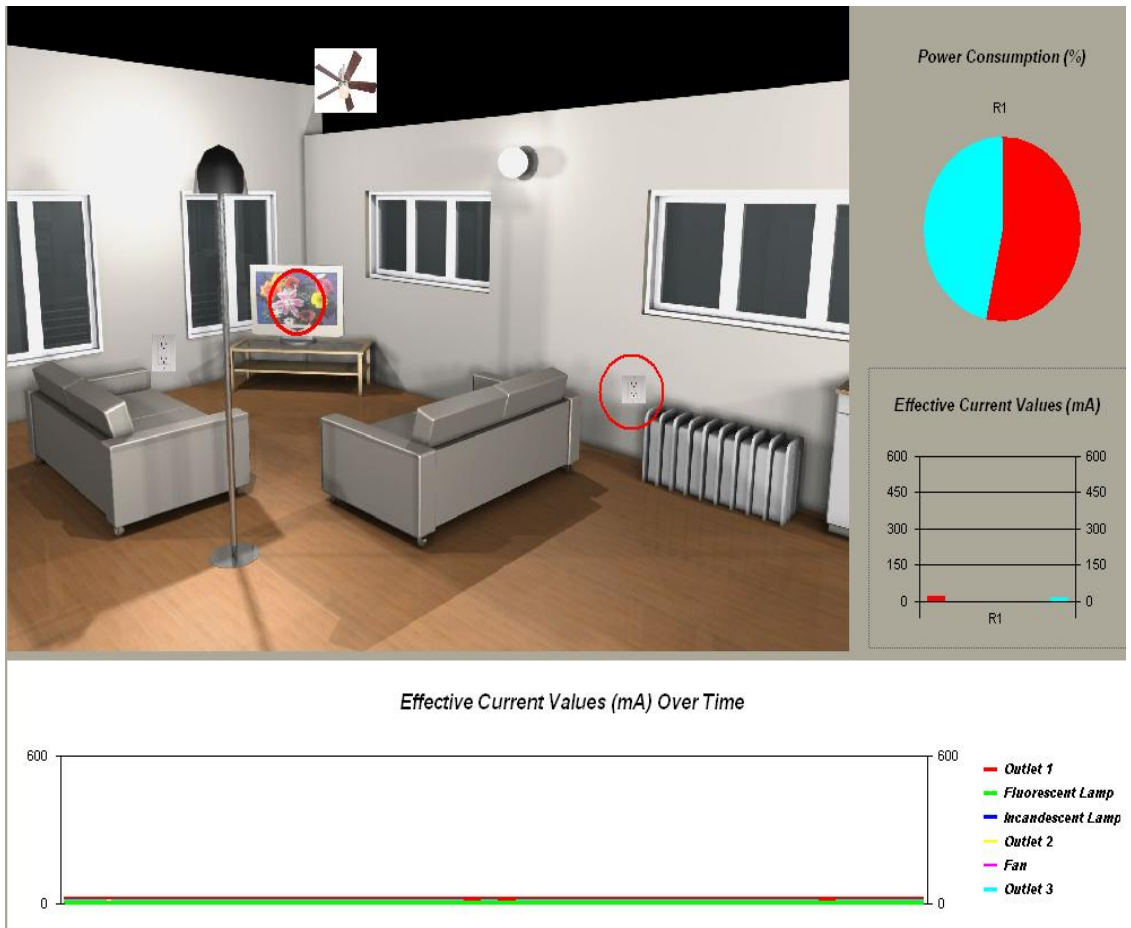
Figure 20: Wasted Energy Denoted in Two Locations. Two Phantom Loads are Plugged into the Two Displayed Outlets

CHAPTER 4

CONCLUSIONS

Real-Time power monitoring is essential for energy conservation and sustainability. As engineers are increasingly turning to virtual environments such as the BIM Model to build and test their creations, they need a better way to virtually monitor the energy input and output of their designs. Detailed load-level power monitoring systems are highly needed, therefore A bottom-up end-use load level power monitoring system has been designed and demonstrated in this thesis as a tool/solution for this lack in real time power monitoring systems. The ZPM system is a promising tool for online real-time energy consumption monitoring in that it is accessible to any user. A virtual environment simulates an actual room, allowing users to easily locate those devices that are causing energy waste in the real environment. A simple click of the mouse informs users exactly how and when energy is consumed. This simplicity gives the system both commercial appeal and use in multiple applications within both academia and the industry. It is hoped that this approach to visualizing information will encourage users to be more conscious of their energy consumption needs. Future studies will be devoted to improving this system with new features and updates according to the demands of the consumer.

Many improvement can be done to the ZPM system to make it more complete and powerful tool. For instance, the data communication between the MPDC unit and the Nodes can be implemented wirelessly using Zigbee protocol (the standard protocol for

mesh networks in home automation) instead of the TWI interface utilized in the prototype to demonstrate the system. This will make the ZPM system easier to install in both new and existing homes and will reduce or even eliminate the need to rewire the electrical system. The ZPM system is to be installed in ZNETH home (Zero-Net Energy Test Home) on our campus. Once it is installed, research is to be conducted to evaluate ZPM's effectiveness on home residents and on energy conservation. More accurate estimation can be achieved by testing the ZPM system on a larger scale, such as testing it in one hundred houses or even more in the local community in Omaha. Adding control in addition to monitoring would make the ZPM system more complete and powerful solution for energy conservation. A feasibility study for such systems, including installing and maintenance costs, besides minimizing the energy used to power the ZPM system compared to the energy saved achieved would also foster the research in this area.

As the supply and demand of electrical energy is challenged within the context of environmental awareness and national security, the need to evaluate and advance each side is growing in importance. The electrical distribution system plays a critical role in the built environment as it is an enabling product, an intangible necessity used to power our systems.

The proposed direct feedback will result in savings more than the 10-20% average savings of the existing systems, since it is provides both aggregate and detailed information of energy consumption in a typical home. The outcome of this research will promote greater interest in the design and development of novel devices and parts to increase the use of power monitoring and home automation. This approach is expected to pave the way for innovation in the built environment and provide for independence from

the grid resulting in reduction of energy utilization at critical times during the day when energy is most needed by larger scale users. It will open opportunities to advance research in other areas. In summary, the simple marriage of the electrical distribution system, novel monitoring and automation, and virtual energy management tools is expected to provide many advantages over exiting approaches including energy efficiency, simplified system components, and the advancement of renewable resource power sources.

Energy consumption and conservation are among society's greatest challenges, and the built environment has a concentrated impact on our natural environment, economy, and health. Significantly reducing energy consumption in the built environment will have profound economical and ecological benefits. But in order to realize those benefits, we need an energy system that is truly be able to interface with the smart grid and be able to react to negotiations and respond to changes in energy requirements.

APPENDIX A

Master Software

#define Own_Slave_Address 100

#define   Num_slaves 6

Chip type   : ATmega16

Program type   : Application

Clock frequency   : 16.000000 MHz

Memory model   : Small

External RAM size   : 0

Data Stack size   : 256

```
****************************************************/


#include <mega16.h>

#include <delay.h>

#include <stdlib.h>

#include <string.h>


// Alphanumeric LCD Module functions

#asm

  .equ __lcd_port=0x1B ;PORTA

#endasm

#include <lcd.h>

char str[8] = "Hi";

unsigned char slave_Nom = 0, Node_Address =0;
```

```
#define STX 2   //ASCII code for Start of Text

#define ETX 3   //ASCII code for End of Text

#define LF 10   //ASCII code for Line Feed (New Line)
```

// ---

//   USART Functions

// ---

```
#define Num_strings 3   //3 strings: slave number, voltage and current strings

#define String_Length 8

#define Extra_data    16

//#define    starting_number 100  //add this number to slave numbers just in communication

#define    V_Coe 36

#define    Calib_Coe0 39

#define    Calib_Coe1 5.1

#define    Calib_Coe2 5.1

#define    Calib_Coe3 1

#define    Calib_Coe4 15

#define    Calib_Coe5 10

#define    display_time 700   //ms, 2500 is good for real circuits

#define TX_BUFFER_SIZE (Num_strings*String_Length+Extra_data)

#define RXB8 1

#define TXB8 0

#define UPE 2

#define OVR 3

#define FE 4

#define UDRE 5

#define RXC 7

#define FRAMING_ERROR (1<<FE)
```

```
#define PARITY_ERROR (1<<UPE)

#define DATA_OVERRUN (1<<OVR)

#define DATA_REGISTER_EMPTY (1<<UDRE)

#define RX_COMPLETE (1<<RXC)

// USART -----------------------------------------------------

// USART Receiver buffer

#define RX_BUFFER_SIZE 60

char rx_buffer[RX_BUFFER_SIZE];

unsigned char USART_RX_NL_counter = 0;

#if RX_BUFFER_SIZE<256

unsigned char rx_wr_index,rx_rd_index,rx_counter;

#else

unsigned int rx_wr_index,rx_rd_index,rx_counter;

#endif


// This flag is set on USART Receiver buffer overflow

bit rx_buffer_overflow;

// USART Receiver interrupt service routine

interrupt [USART_RXC] void usart_rx_isr(void)

{

char status,data;

status=UCSRA;

data=UDR;

if ((status & (FRAMING_ERROR | PARITY_ERROR | DATA_OVERRUN))==0)

  {

     rx_buffer[rx_wr_index]=data;

     if(data == 10){
```

```
      ++USART_RX_NL_counter;

      if (USART_RX_NL_counter == 255) USART_RX_NL_counter=0;

    };

    if (++rx_wr_index == RX_BUFFER_SIZE) rx_wr_index=0;

    if (++rx_counter == RX_BUFFER_SIZE)

    {

      rx_counter=0;

      rx_buffer_overflow=1;

    };

  };

}


#ifndef _DEBUG_TERMINAL_IO_

// Get a character from the USART Receiver buffer

#define _ALTERNATE_GETCHAR_

#pragma used+

char getchar(void)

{

char data;

while (rx_counter==0);

data=rx_buffer[rx_rd_index];

if (++rx_rd_index == RX_BUFFER_SIZE) rx_rd_index=0;

#asm("cli")

--rx_counter;

#asm("sei")

return data;

}
```

```
#pragma used-
#endif


// USART Transmitter buffer
char tx_buffer[TX_BUFFER_SIZE];


#if TX_BUFFER_SIZE<256
unsigned char tx_wr_index,tx_rd_index,tx_counter;
#else
unsigned int tx_wr_index,tx_rd_index,tx_counter;
#endif


// USART Transmitter interrupt service routine
bit send_to_slave=1, data_sent=0;
interrupt [USART_TXC] void usart_tx_isr(void)
{
   if (tx_counter){  //i.e. if all buffered data has not been transmitted
     --tx_counter;
     //if (send_to_slave) delay_us(100);   // delay is very important to avoid data overrun
at the slave side
     UDR=tx_buffer[tx_rd_index];
     if (++tx_rd_index == TX_BUFFER_SIZE) tx_rd_index=0;
   }
   else
     data_sent = 1;
}


#ifndef _DEBUG_TERMINAL_IO_
```

```
// Write a character to the USART Transmitter buffer

#define _ALTERNATE_PUTCHAR_

#pragma used+

void putchar(char c)

{

    data_sent = 0;

    while (tx_counter == TX_BUFFER_SIZE);

    #asm("cli")

    if (tx_counter || ((UCSRA & DATA_REGISTER_EMPTY)==0)){

        tx_buffer[tx_wr_index]=c;

        if (++tx_wr_index == TX_BUFFER_SIZE) tx_wr_index=0;

        ++tx_counter;

    }

    else

        UDR=c;

    #asm("sei")

}

#pragma used-

#endif
```

//---

```
#include <stdio.h>    //include after the special versions of getchar() and putchar() are
defined
```

//---

```
//////////////////////////////

struct DATA{

    unsigned int chip_Nom;

    float I, V;

} data[Num_slaves+1];   //data[0] cells are not used
```

```
// ----------------------------------------------------------------------------------------

void reset_buffers_n_flags ()

{

    //flush the TX buffer

    tx_wr_index = 0;

    tx_rd_index = 0;

    tx_counter = 0;

    //reset flags

    send_to_slave = 1;

    data_sent = 0;

}

//----------------------------------------------------------------------------------

void send_data_to_computer()

{

    reset_buffers_n_flags();

    send_to_slave = 0;   //data will be sent to the computer not to a slave


    //Start the data package

    putsf("STX");


    //Send the data package

    //Node Number

    itoa(Node_Address, str);

    puts(str);

    //voltage

    ftoa(data[Node_Address].V, 2, str);
```

puts(str);

//current

ftoa(data[Node_Address].I, 2, str);

puts(str);

//End the data package

putsf("ETX");

//Wait until the trasmission is complete

while(!data_sent);

//delay_ms(1);

}

// --

// 2 Wire bus interrupt service routine & Functions

// --

#define TWINT 7 //TWI Interrupt Flag Bit

#define TWEA 6 //TWI Enable Acknowledge Bit

#define TWSTA 5 //TWI START Condition Bit

#define TWSTO 4 //TWI STOP Condition Bit

#define TWEN 2 //TWI Enable Bit

#define TWIE 0 //TWI Interrupt Enable Bit

#define SR_SLA_W_ACK 0x60   //Own SLA+W has been received, ACK has been returned

#define SR_DATA_ACK 0x80   //Previously addressed with own SLA+W; data has been received; ACK has been returned

#define STOP 0xA0 //A STOP condition or repeated START condition has been received while still addressed as slave

```
#define R 1     //Read bit (high level at SDA)

#define W 0     //Write bit (low level at SDA)

#define START 0x08       //A START condition has been transmitted

#define MT_SLA_ACK  0x18   //SLA+W has been transmitted, ACK has been received

#define MT_DATA_ACK 0x28   //Data byte has been transmitted, ACK has been
received

#define twi_rx_buf_size  10

bit MT_mode=0, SR_mode=1, twi_data_available=0, twi_rx_buf_Full = 0, twi_rx_error
= 0, string_received = 0;

char twi_rx_buffer[twi_rx_buf_size], twi_rx_string[twi_rx_buf_size], twi_rx_byte;

char Voltage[twi_rx_buf_size], Current[twi_rx_buf_size];

unsigned char twi_rx_wr_index=0, twi_rx_counter=0;

#define twi_tx_buf_size  10

bit string_sent=0, twi_tx_error=0, sending_in_process = 0;

char twi_tx_buffer[twi_tx_buf_size], twi_tx_byte;

unsigned char twi_tx_rd_index=0, twi_tx_str_length=0;

unsigned char SLA_R, SLA_W;   //destination Slave Address, ...

void twi_rx_reset(void)

{

  SR_mode = 1;

  MT_mode = 0;

  twi_rx_error = 0;

  twi_data_available = 0;

  twi_rx_wr_index = 0;

  twi_rx_counter=0;

  twi_rx_buf_Full = 0;

  string_received = 0;

}
```

```
interrupt [TWI] void twi_isr(void)

{

  if(SR_mode)

  {

    if(!twi_data_available && !twi_rx_buf_Full)

    {

        if ((TWSR & 0xF8) == SR_SLA_W_ACK){    //slave address received, ACK returned

            twi_rx_reset();

            PORTB.0 = 1;

        }


        else if ((TWSR & 0xF8) == SR_DATA_ACK){   //data received, ACK returned

            twi_rx_byte = TWDR;


            if     (twi_rx_byte == STX)   twi_rx_reset();   //reset twi RX buffer and flags

            else if (twi_rx_byte == ETX)   string_received = 1;            //Don't store the ETX character

            else{                              //store data in the rx buffer

              twi_rx_buffer[twi_rx_wr_index] = twi_rx_byte;

              if(++twi_rx_wr_index == twi_rx_buf_size) twi_rx_wr_index = 0;

              if(++twi_rx_counter == twi_rx_buf_size)  twi_rx_buf_Full = 1;

            };

        }


        else if ((TWSR & 0xF8) == STOP){

            PORTB.0 = 0;
```

```
            if (twi_rx_byte == ETX){   //Data is available (data received successfully)

              twi_data_available = 1;

            }

            else

            {

              twi_rx_error = 1;

              twi_rx_reset();

            };

          };

        };


    //reset twi control values (reset the flag, enable twi and enable acknoledgement)

    TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);

    }


    else if(MT_mode)

    {

      if(!string_sent)

      {

          if ((TWSR & 0xF8) == START){      //START has been sent; ACK has been
received

              PORTB.0 = 1;

              //Load SLA_W into TWDR Register.

              TWDR = SLA_W;

              //reset twi control values (reset the flag, enable twi and enable acknoledgement)

              TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);

          }
```

```
      else if ((TWSR & 0xF8) == MT_SLA_ACK)      //Slave address has been sent;
ACK has been received
      {
         twi_tx_byte = STX;

         TWDR = twi_tx_byte;

         //reset twi control values (reset the flag, enable twi and enable acknoledgement)

         TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);

      }
      else if ((TWSR & 0xF8) == MT_DATA_ACK)      //Data has been sent; ACK has
been received
      {
         if(twi_tx_byte == ETX)

         {

            string_sent = 1;

            //Enter Slave Receiver Mode

            SR_mode = 1;

            MT_mode = 0;


            //send STOP signal

            TWCR = (1<<TWINT)|(1<<TWEN)|(1<<TWSTO)|(1<<TWIE)|(1<<TWEA);

            PORTB.0 = 0;

            sending_in_process = 0;

         }

         else

         {

            twi_tx_byte = twi_tx_buffer[twi_tx_rd_index];

            TWDR = twi_tx_byte;

            ++twi_tx_rd_index;
```

```
            //reset twi control values (reset the flag, enable twi and enable
acknoledgement)

            TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);

        };

      };

    };

  };

}


void twi_get_string(char* V_I)

{

  unsigned char i;


  if(twi_data_available){

    for(i=0; i<twi_rx_counter; i++)

    {

       twi_rx_string[i] = twi_rx_buffer[i];

       V_I[i] = twi_rx_buffer[i];

    };

    twi_rx_string[twi_rx_counter] = NULL;

    V_I[twi_rx_counter] = NULL;


    //twi_rx_reset();

    twi_data_available = 0;

  };

}


void twi_tx_reset(void)
```

```c
{
    //Enter Master Transmitter Mode
    SR_mode = 0;
    MT_mode = 1;
    string_sent = 0;
    twi_tx_error = 0;
    sending_in_process = 0;
    twi_tx_rd_index = 0;
}


void twi_put_string(char* twi_tx_string, unsigned char slave_add)
{
    unsigned char i;

    //Reset twi TX buffer and enter Master Transmitter Mode
    twi_tx_reset();

    //prepare slave address + write byte
    //SLA_R = 2*slave_add + R;
    SLA_W = 2*slave_add + W;

    //store the string in the twi tx buffer
    twi_tx_str_length = strlen(twi_tx_string);
    for(i=0; i<twi_tx_str_length; i++)
        twi_tx_buffer[i]= twi_tx_string[i];
    twi_tx_buffer[twi_tx_str_length] = ETX;
    twi_tx_str_length += 1;
```

```
  //send START signal
  TWCR = (1<<TWINT)|(1<<TWSTA)|(1<<TWEN)|(1<<TWIE);
  sending_in_process = 1;


  //wait until the sting is sent
  while(sending_in_process);


  //go back to the Slave Receiver Mode
  twi_rx_reset();
}


void show_and_calibrate_data()
{
  unsigned char row=0, colum=0;
  float Calib_Coe;   //Calibration Coeffecient


  slave_Nom = Node_Address - 1;


  if     (slave_Nom == 0)   Calib_Coe=10;
  else if (slave_Nom == 1)   Calib_Coe=10;
  else if (slave_Nom == 2)   Calib_Coe=10;
  else if (slave_Nom == 3)   Calib_Coe=10;
  else if (slave_Nom == 4)   Calib_Coe=10;
  else if (slave_Nom == 5)   Calib_Coe=10;


  if (slave_Nom < 3 )
```

```
{
    row = 0;
    colum = slave_Nom*6;
}
else
{
    row = 2;
    colum = (slave_Nom-3)*6;
};


data[Node_Address].V = V_Coe*atof(Voltage);
data[Node_Address].I = (float) 1000*atof(Current)/Calib_Coe;


//display the new data

/*
//voltage
ftoa(data[Node_Address].V, 0, str);
lcd_gotoxy(colum,row); //clear the last value first
lcd_putsf("     ");
lcd_gotoxy(colum,row);
lcd_puts(str);
//lcd_puts(Voltage);
//lcd_putsf("V ");
*/


//
```

```
   //current

   ftoa(data[Node_Address].I, 0, str);

   lcd_gotoxy(colum,row);

   lcd_putsf("     ");

   lcd_gotoxy(colum,row);

   lcd_puts(str);

   //lcd_puts(Current);

   //lcd_putsf("mA ");

   //

}


void twi_get_data()

{

   //request the voltage value

   itoa(1,str);   //1 is the code to ask for voltage value

   twi_put_string(str, Node_Address);

   while(!twi_data_available);

   twi_get_string(Voltage);


   //request the current value

   itoa(2,str);   //2 is the code to ask for the current value

   twi_put_string(str, Node_Address);

   while(!twi_data_available);

   twi_get_string(Current);

}
// ----------------------- Control Part -----------------------------------

void Read_RX_buffer_and_Respond (void)
```

```
{
    char *str,*rx_str;
    signed char str_cmp_rslt = -1;
    int node = 0;


    str=gets(rx_str, 3);
    str_cmp_rslt = strcmpf(rx_str,"Go");
    if (str_cmp_rslt == 0)  PORTB  ^= (1 << 7);   //if compare match, toggle pin #7 status
    else
    {
        node = atoi(rx_str);
        if (node >= 1 && node <=6 ) PORTB ^= (1 << node);
    };


    //lcd_clear();
    //lcd_puts(rx_str);
}


// Declare your global variables here

void main(void)
{
// Declare your local variables here
unsigned char Reset_Source =0;
// Reset Source checking
if (MCUCSR & 1)
    {
```

```
  // Power-on Reset

  MCUCSR&=0xE0;

  // Place your code here

  Reset_Source=0;

  }

else if (MCUCSR & 2)

  {

  // External Reset

  MCUCSR&=0xE0;

  // Place your code here

  Reset_Source=1;

  }

else if (MCUCSR & 4)

  {

  // Brown-Out Reset

  MCUCSR&=0xE0;

  // Place your code here

  Reset_Source=2;

  }

else if (MCUCSR & 8)

  {

  // Watchdog Reset

  MCUCSR&=0xE0;

  // Place your code here

  Reset_Source=3;

  }

else if (MCUCSR & 0x10)
```

```
{
// JTAG Reset
MCUCSR&=0xE0;
// Place your code here
Reset_Source=4;
};
```

```
// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTA=0x00;
DDRA=0x00;
```

```
// Port B initialization
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out
Func0=Out
// State7=T State6=T State5=T State4=T State3=0 State2=0 State1=0 State0=0
PORTB=0x00;
DDRB=0xFF;
```

```
// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTC=0x00;
DDRC=0x00;
```

```
// Port D initialization
```

```
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;


// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;


// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
```

```
TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;


// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;


// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;

MCUCSR=0x00;
```

```
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;


// USART initialization
// Communication Parameters: 8 Data, 1 Stop, Even Parity
// USART Receiver: Off
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 38400
UCSRA=0x00;
UCSRB=0x48;
UCSRC=0xA6;
UBRRH=0x00;
UBRRL=0x19;


// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;


//
// 2 Wire Bus initialization
// Generate Acknowledge Pulse: On
// 2 Wire Bus Slave Address:
// General Call Recognition: On
// Bit Rate: 400.000 kHz
```

```
TWSR=0x00;

TWBR=0x0C;

TWAR = Own_Slave_Address*2 + 1;

TWCR=0x45;

//

/*

// 2 Wire Bus initialization

// Generate Acknowledge Pulse: On

// 2 Wire Bus Slave Address: Own_Slave_Address

// General Call Recognition: On

// Bit Rate: 40.000 kHz

TWSR=0x00;

TWBR=0xC0;

TWAR = Own_Slave_Address*2 + 1;

TWCR=0x45;

*/


// LCD module initialization

lcd_init(20);

//lcd_gotoxy(0, 0);



//

if(Reset_Source == 0){

    lcd_putsf("Power-on Reset"); delay_ms(600); lcd_clear();

}

else if(Reset_Source == 1){
```

```
        lcd_putsf("External Reset"); delay_ms(300); lcd_clear();

}
else if(Reset_Source == 2){

        lcd_putsf("Brown-Out Reset"); delay_ms(300); lcd_clear();

}
else if(Reset_Source == 3){

        lcd_putsf("Watchdog Reset"); delay_ms(300); lcd_clear();

}
else if(Reset_Source == 4){

        lcd_putsf("JTAG Reset"); delay_ms(300); lcd_clear();

};
//


// Global enable interrupts
#asm("sei")
lcd_gotoxy(0, 1);lcd_putsf("I_rms (mA)");
//delay_ms(100);
lcd_gotoxy(2, 3);lcd_putsf("EPA Project, UNL");
while (1)
 {


   for(Node_Address=1; Node_Address <= Num_slaves; Node_Address++) {
       //get the voltage and current values from a Node
       twi_get_data();
       //display the data and convert it to the actual values
       show_and_calibrate_data();
       //send the data to the computer
```

```
        send_data_to_computer();

        //optional delay

        //delay_ms(10);

    };


    // in the while loop
    if(USART_RX_NL_counter > 0){

      Read_RX_buffer_and_Respond();

      --USART_RX_NL_counter;

    };


 };


}
```

//////////////////////////////////// End of the Master Program ////////////////////////////////////

APPENDIX B

The Slave Program:

```
#define Own_Slave_Address 1

#define Master_Address 100

/****************************************************

Chip type           : ATmega16

Program type        : Application

Clock frequency     : 16.000000 MHz

Memory model        : Small

External RAM size   : 0

Data Stack size     : 256

****************************************************/

#include <mega16.h>

#include <delay.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

// User Setting (modifiable according to the application)

//------------------------- User Setting -------------------------------------

#define ADC_VREF_TYPE 0x40     //Vref is AVCC pin voltage

#define FIRST_INPUT 0

#define LAST_INPUT  1

#define T0_clk   2    //2MHz  adjust according to your T0 speed

#define OCR0_value 99

#define k ((float)5/1023)     // Trnasfer ratio = Vref/maximum ADCW value

#define F 60    //F=60 Hz
```

```c
#define T ((float)1000000/F)   //[uS]

#define RX_BUFFER_SIZE 8

#define TX_BUFFER_SIZE 50

//-----------------------------------------------------------------------------

// Settings left according to the User Setting (Please do not modify)

//---------------------------------------------------------------------------------

#define Num_channels (LAST_INPUT-FIRST_INPUT+1)   //Number of channels

#define period ((OCR0_value +1)/T0_clk)  //[uS] time required by timer0 to count from 0
to OCR0 Register value

#define dt period*Num_channels  //[uS]for example: dt=200us= 2*(OCR0+1)/T0_clk=
2*(199+1)/2MHz

#define samples_per_period (1000000/(dt*F) + 1)

//---------------------------------------------------------------------------------

/* Alphanumeric LCD Module functions

#asm

  .equ __lcd_port=0x18 ;PORTB

#endasm

#include <lcd.h>

char str[8] = "Hello";

*/

// 2 Wire bus interrupt service routine & Functions ----------------------------------------------
------------------

#define TWINT 7 //TWI Interrupt Flag Bit

#define TWEA  6 //TWI Enable Acknowledge Bit

#define TWSTA 5 //TWI START Condition Bit

#define TWSTO 4 //TWI STOP Condition Bit

#define TWEN  2 //TWI Enable Bit

#define TWIE  0 //TWI Interrupt Enable Bit
```

```
#define STX 2   //ASCII code for Start of Text

#define ETX 3   //ASCII code for End of Text

#define LF 10   //ASCII code for Line Feed (New Line)

#define SR_SLA_W_ACK  0x60   //Own SLA+W has been received, ACK has been
returned

#define SR_DATA_ACK 0x80     //Previously addressed with own SLA+W; data has
been received; ACK has been returned

#define STOP 0xA0  //A STOP condition or repeated START condition has been
received while still addressed as slave

#define R 1     //Read bit (high level at SDA)

#define W 0     //Write bit (low level at SDA)

#define START 0x08       //A START condition has been transmitted

#define MT_SLA_ACK  0x18   //SLA+W has been transmitted, ACK has been received

#define MT_DATA_ACK 0x28   //Data byte has been transmitted, ACK has been
received

#define twi_rx_buf_size  10

bit MT_mode=0, SR_mode=1, twi_data_available=0, twi_rx_buf_Full = 0, twi_rx_error
= 0, string_received = 0;

char twi_rx_buffer[twi_rx_buf_size], twi_rx_string[twi_rx_buf_size], twi_rx_byte;

unsigned char twi_rx_wr_index=0, twi_rx_counter=0;

#define twi_tx_buf_size  10

bit string_sent=0, twi_tx_error=0, sending_in_process = 0, data_sent = 0;

char twi_tx_buffer[twi_tx_buf_size], twi_tx_byte;

unsigned char twi_tx_rd_index=0, twi_tx_str_length=0;

unsigned char SLA_W;   //destination Slave Address, ...

//unsigned char SLA_R;


void twi_rx_reset(void)

{
```

```
  SR_mode = 1;

  MT_mode = 0;

  twi_rx_error = 0;

  twi_data_available = 0;

  twi_rx_wr_index = 0;

  twi_rx_counter=0;

  twi_rx_buf_Full = 0;

  string_received = 0;

}


interrupt [TWI] void twi_isr(void)

{

  if(SR_mode)

  {

    if(!twi_data_available && !twi_rx_buf_Full)

    {

      if ((TWSR & 0xF8) == SR_SLA_W_ACK){    //slave address received, ACK
returned

        twi_rx_reset();

        PORTB.0 = 1;

      }


      else if ((TWSR & 0xF8) == SR_DATA_ACK){   //data received, ACK returned

        twi_rx_byte = TWDR;


        if     (twi_rx_byte == STX)   twi_rx_reset();   //reset twi RX buffer and flags

        else if (twi_rx_byte == ETX)   string_received = 1;              //Don't store the
ETX character
```

```
    else{                                    //store data in the rx buffer
      twi_rx_buffer[twi_rx_wr_index] = twi_rx_byte;
      if(++twi_rx_wr_index == twi_rx_buf_size) twi_rx_wr_index = 0;
      if(++twi_rx_counter == twi_rx_buf_size)  twi_rx_buf_Full = 1;
    };
  }


  else if ((TWSR & 0xF8) == STOP){
    PORTB.0 = 0;


    if (twi_rx_byte == ETX){   //Data is available (data received successfully)
      twi_data_available = 1;
    }
    else
    {
      twi_rx_error = 1;
      twi_rx_reset();
    };
  };
};


//reset twi control values (reset the flag, enable twi and enable acknoledgement)
TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);
}


else if(MT_mode)
{
```

```
    if(!string_sent)

  {

      if ((TWSR & 0xF8) == START){      //START has been sent; ACK has been
received

        PORTB.0 = 1;

        //Load SLA_W into TWDR Register.

        TWDR = SLA_W;

        //reset twi control values (reset the flag, enable twi and enable acknoledgement)

        TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);

      }

      else if ((TWSR & 0xF8) == MT_SLA_ACK)      //Slave address has been sent;
ACK has been received

    {

      twi_tx_byte = STX;

      TWDR = twi_tx_byte;

      //reset twi control values (reset the flag, enable twi and enable acknoledgement)

      TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);

    }

      else if ((TWSR & 0xF8) == MT_DATA_ACK)      //Data has been sent; ACK has
been received

    {

      if(twi_tx_byte == ETX)

      {

        string_sent = 1;

        //Enter Slave Receiver Mode

        SR_mode = 1;

        MT_mode = 0;
```

```
                    //send STOP signal

                    TWCR = (1<<TWINT)|(1<<TWEN)|(1<<TWSTO)|(1<<TWIE)|(1<<TWEA);

                    PORTB.0 = 0;

                    sending_in_process = 0;

                }

                else

                {

                    twi_tx_byte = twi_tx_buffer[twi_tx_rd_index];

                    TWDR = twi_tx_byte;

                    ++twi_tx_rd_index;

                    //reset twi control values (reset the flag, enable twi and enable
acknoledgement)

                    TWCR = (1<<TWINT) | (1<<TWEA) | (1<<TWEN) | (1<<TWIE);

                };

            };

        };

}


void twi_tx_reset(void)

{

    //Enter Master Transmitter Mode

    SR_mode = 0;

    MT_mode = 1;

    string_sent = 0;

    twi_tx_error = 0;

    sending_in_process = 0;

    twi_tx_rd_index = 0;
```

```c
    data_sent = 0;

}


void twi_put_string(char* twi_tx_string, unsigned char slave_add)

{

    unsigned char i;


    //Reset twi TX buffer and enter Master Transmitter Mode

    twi_tx_reset();


    //prepare slave address + write byte

    //SLA_R = 2*slave_add + R;

    SLA_W = 2*slave_add + W;


    //store the string in the twi tx buffer

    twi_tx_str_length = strlen(twi_tx_string);

    for(i=0; i<twi_tx_str_length; i++)

        twi_tx_buffer[i]= twi_tx_string[i];

    twi_tx_buffer[twi_tx_str_length] = ETX;

    twi_tx_str_length += 1;


    //send START signal

    TWCR = (1<<TWINT)|(1<<TWSTA)|(1<<TWEN)|(1<<TWIE);

    sending_in_process = 1;


    //wait until the sting is sent

    while(sending_in_process);
```

```
    //go back to the Slave Receiver Mode

    twi_rx_reset();

}


// ----------------------------------------------------------------------

//              ADC & Measurements Functions

// ----------------------------------------------------------------------


char str[8];

unsigned char channel_Nom=0;

unsigned int sampled_value=0, V_max=0;


struct Channel{

    unsigned int sample_Nom;

    unsigned long int segma_sqr_val; //segma_sqr_val: accumulated sum of the squared
values

    float effective_value;

} ch[Num_channels];


bit sampling_done =0;


void update_values ()

{

  if(!sampling_done)

  {

    if(channel_Nom==0)  // takes about 2 uS

    {
```

```c
            ++ch[0].sample_Nom;

            if(sampled_value > V_max)

               V_max = sampled_value;

        }

      else // takes about 10 uS

      {

         ch[1].segma_sqr_val += (unsigned long int) sampled_value * sampled_value;

         if(ch[0].sample_Nom >= samples_per_period )   sampling_done = 1;

      };

   };

}


void adc_reset(void)

{

  V_max = 0;

  ch[0].sample_Nom = 0;

  ch[1].segma_sqr_val = 0;

  sampling_done = 0;

}
// -------------------------------------------------------------------------------------

// ADC interrupt service routine with auto input scanning

interrupt [ADC_INT] void adc_isr(void)  // is called every 50 uS approximately (about 45
uS practically)

{

  // Read the AD conversion result

   sampled_value=ADCW;

   //update values

   if(!sampling_done)
```

```
        {PORTB.5 = 1; update_values(); PORTB.5 = 0;};

    // Select next ADC input

    if ( ++channel_Nom > (LAST_INPUT-FIRST_INPUT) )    channel_Nom=0;

    //ADMUX = channel_Nom | (ADC_VREF_TYPE & 0xff);

    ADMUX=(FIRST_INPUT | (ADC_VREF_TYPE & 0xff))+channel_Nom;

    //reset the flag of the auto-triger source

    //the autotriger source here is timer/counter0 output compare match flag(rising
flag/positive edge)

    TIFR |= 0x02;

}


void calculate_values (void)  //takes about 500 uS

{

    if(sampling_done)

    {

        //stop ADC module

        TCCR0=0x08;

        //calculate RMS values

        ch[0].effective_value=k*V_max*0.7071;

        PORTD = V_max;

        //2 is added because the negative half is grounded

        ch[1].effective_value=k*sqrt((float)2*ch[1].segma_sqr_val*dt/T);

        //reset_calculations & resume ADC Module

        adc_reset();

        TCCR0=0x0A;

        TCNT0=0x00;

    };

}
```

```c
// --------------------------------------------------------------------------------

void respond (void)

{

  unsigned char i;

  int data_requested=0;

  // 1 means Voltage

  // 2 means Current


   if(twi_data_available)

   {

     PORTB.1 = 1;


     //read data

     for(i=0; i<twi_rx_counter; i++)

       twi_rx_string[i] = twi_rx_buffer[i];

     twi_rx_string[twi_rx_counter] = NULL;

     data_requested = atoi(twi_rx_string);


     if (data_requested == 1)

     {

       //send voltage value

       ftoa(ch[0].effective_value, 2, str);

       twi_put_string(str, Master_Address);

     }

     else if (data_requested == 2)

     {

       //send current value
```

```
        ftoa(ch[1].effective_value, 2, str);

        twi_put_string(str, Master_Address);

    };


    twi_data_available = 0;

    PORTB.1 = 0;

  };

}


// Declare your global variables here

void main(void)

{

// Declare your local variables here


// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTA=0x00;

DDRA=0x00;


// Port B initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out
Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTB=0x00;

DDRB=0xFF;
```

// Port C initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTC=0x00;

DDRC=0x00;

// Port D initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTD=0x00;

DDRD=0xFF;

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: 2000.000 kHz

// Mode: CTC top=OCR0

// OC0 output: Disconnected

TCCR0=0x0A;

TCNT0=0x00;

OCR0=OCR0_value;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer 1 Stopped

// Mode: Normal top=FFFFh

// OC1A output: Discon.

// OC1B output: Discon.

```
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;


// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;
```

```c
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;


// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;


// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;


//
// 2 Wire Bus initialization
// Generate Acknowledge Pulse: On
// 2 Wire Bus Slave Address:
// General Call Recognition: On
// Bit Rate: 400.000 kHz
TWSR=0x00;
TWBR=0x0C;
TWAR = Own_Slave_Address*2 + 1;
TWCR=0x45;
//
```

```
/*
// 2 Wire Bus initialization
// Generate Acknowledge Pulse: On
// 2 Wire Bus Slave Address: Own_Slave_Address
// General Call Recognition: On
// Bit Rate: 40.000 kHz
TWSR=0x00;
TWBR=0xC0;
TWAR = Own_Slave_Address*2 + 1;
TWCR=0x45;
*/
// ADC initialization
// ADC Clock frequency: 1000.000 kHz
// ADC Voltage Reference: AVCC pin
// ADC Auto Trigger Source: Timer0 Compare Match
ADMUX=FIRST_INPUT | (ADC_VREF_TYPE & 0xff);
ADCSRA=0xAC;
SFIOR&=0x1F;
SFIOR|=0x60;

// LCD module initialization
//lcd_init(16);

// Global enable interrupts
#asm("sei")
//lcd_putsf("OKey");
```

```
//stop ADC module

//TCCR0=0x00;

while (1)

  {

      if(sampling_done){

        PORTB.3 = 1;

        calculate_values();   //it takes about 500uS

        PORTB.3 = 0;

        sampling_done = 0;

      };


      if(twi_data_available){

        respond();

        twi_data_available = 0;

      };

  };


}

// End of the Program
```

APPENDIX C

The Virtual Environment Program (written using Visual Basic 6.0):

Option Explicit

Private Const Record_size = 100

Private Const Number_of_Nodes = 6

Dim DataArray(1 To Record_size, 1 To Number_of_Nodes) As String

Dim PieArray(0, 1 To Number_of_Nodes) As String

Dim file_name As String

Dim Data_Received As String

Dim Data_After_STX As String

Dim First_STX As Integer

Dim NodeNumber_LF As Integer

Dim Voltage_LF As Integer

Dim Current_LF As Integer

Dim First_ETX As Integer

Dim last_number As Integer

Dim Record_request As Boolean

Dim Node_number As String

Dim Voltage As String

```
Dim Current As String

Dim Current_Sample As String


Dim oExcel As Object

Dim oBook As Object

Dim oSheet As Object


Private Type B_node

 Num_of_rx_values As Long

 Index As Long

 Voltage(1 To Record_size) As String

 Current(1 To Record_size) As String

End Type


Dim board_node(1 To 6) As B_node


Private Sub Update_Graph()

  'Create an array with Number_of_Nodes columns and Record_size rows

  Dim r As Integer, c As Integer

  For c = 1 To Number_of_Nodes

    For r = 1 To Record_size

      DataArray(r, c) = board_node(c).Current(r)

    Next r

  Next c


  'Update PieArray data

  For c = 1 To Number_of_Nodes
```

PieArray(0, c) = board_node(c).Current(Record_size)

Next c

'Associate data with each graph

MSChart1.ChartData = DataArray

MSChart2.ChartData = PieArray

MSChart3.ChartData = PieArray

'Set legends

MSChart1.Column = 1

MSChart1.ColumnLabel = "Outlet 1"

MSChart1.Column = 2

MSChart1.ColumnLabel = "Fluorescent Lamp"

MSChart1.Column = 3

MSChart1.ColumnLabel = "Incandescent Lamp"

MSChart1.Column = 4

MSChart1.ColumnLabel = "Outlet 2"

MSChart1.Column = 5

MSChart1.ColumnLabel = "Fan"

MSChart1.Column = 6

MSChart1.ColumnLabel = "Outlet 3"

'Set legends

MSChart2.Column = 1

MSChart2.ColumnLabel = "Outlet 1"

MSChart2.Column = 2

MSChart2.ColumnLabel = "Fluorescent Lamp"

```vb
    MSChart2.Column = 3

    MSChart2.ColumnLabel = "Incandescent Lamp"

    MSChart2.Column = 4

    MSChart2.ColumnLabel = "Outlet 2"

    MSChart2.Column = 5

    MSChart2.ColumnLabel = "Fan"

    MSChart2.Column = 6

    MSChart2.ColumnLabel = "Outlet 3"


End Sub


Private Sub Record_Data()


    'Transfer the array to the worksheet starting at cell A2

    'oSheet.Range("A2").Resize(Record_size, Number_of_Nodes).Value = DataArray


End Sub


Private Sub Initialize_Values()
    Dim i As Integer
    Dim j As Long
    For i = 1 To Number_of_Nodes
        PieArray(0, i) = 0
        board_node(i).Index = 0
        board_node(i).Num_of_rx_values = 0
        For j = 1 To Record_size
            board_node(i).Voltage(j) = 0
```

```
        board_node(i).Current(j) = 0
    Next j
  Next i


  MSChart1.ChartData = DataArray

  MSChart2.ChartData = PieArray


End Sub


'Private Function Buffer_Values()

Private Sub Buffer_Values()


  Dim i As Long


  If board_node(Node_number).Num_of_rx_values < Record_size Then


    board_node(Node_number).Index = board_node(Node_number).Index + 1

    i = board_node(Node_number).Index

    board_node(Node_number).Num_of_rx_values =
board_node(Node_number).Num_of_rx_values + 1

    board_node(Node_number).Current(i) = Current

    board_node(Node_number).Voltage(i) = Voltage


  Else

    For i = 1 To (Record_size - 1)

     board_node(Node_number).Voltage(i) = board_node(Node_number).Voltage(i + 1)

     board_node(Node_number).Current(i) = board_node(Node_number).Current(i + 1)

    Next
```

```
        board_node(Node_number).Voltage(Record_size) = Val(Voltage)

        board_node(Node_number).Current(Record_size) = Val(Current)


   End If


End Sub


Private Sub Air1_Change()


End Sub


Private Sub Command1_Click()
MSComm1.Output = "Go" + Chr$(10)
End Sub


Private Sub Command2_Click()


End Sub


Private Sub Command2_DragDrop(Source As Control, X As Single, Y As Single)


End Sub


Private Sub Command2_MouseDown(Button As Integer, Shift As Integer, X As Single,
Y As Single)


End Sub
```

```
Private Sub Form_Load()

    ' Close the serial port if it is open

    If MSComm1.PortOpen = True Then

    MSComm1.PortOpen = False

    End If

    ' Open the serial port

    MSComm1.CommPort = 1

    MSComm1.Settings = "38400,E,8,1"

    MSComm1.RThreshold = 200

    MSComm1.InputLen = 200

    MSComm1.PortOpen = True

    ' Initialize Values

    Call Initialize_Values


    'Start a new workbook in Excel

    'Set oExcel = CreateObject("Excel.Application")

    'Set oBook = oExcel.Workbooks.Add

    'Set oSheet = oBook.Worksheets(1)


End Sub


Private Sub Form_Unload(Cancel As Integer)

    MSComm1.PortOpen = False

    'Save the Workbook and Quit Excel

    'Kill "C:\RTPM_Record.xlsx"

    'oBook.SaveAs "C:\RTPM_Record.xlsx"
```

```vbnet
    'oExcel.Quit

End Sub



Private Sub MSChart2_Click()
  Dim i As Integer


  For i = 1 To Number_of_Nodes
    If Node_TextBox(i).Visible = True Then
      Node_TextBox(i).Visible = False
    Else
      Node_TextBox(i).Visible = True
    End If
  Next i

End Sub

Private Sub MSChart1_Click()
  Dim i As Integer


  For i = 1 To Number_of_Nodes
    If Node_TextBox(i).Visible = True Then
      Node_TextBox(i).Visible = False
    Else
```

```
          Node_TextBox(i).Visible = True

      End If

   Next i


End Sub


Private Sub MSChart3_Click()

   Dim i As Integer



   For i = 1 To Number_of_Nodes

      If Node_TextBox(i).Visible = True Then

        Node_TextBox(i).Visible = False

      Else

        Node_TextBox(i).Visible = True

      End If

   Next i


End Sub


Private Sub MSComm1_OnComm()


On Error GoTo Exit_this_sub


   'Data_Received = MSComm1.Input

   Data_Received = Data_Received + MSComm1.Input
```

```
RX_Buffer_TextBox.Text = Data_Received + vbCrLf


'Check if the packet contains all the values

First_STX = InStr(1, Data_Received, "STX")

If First_STX > 0 Then

    NodeNumber_LF = InStr(First_STX + 4, Data_Received, Chr$(10))

   If NodeNumber_LF > 0 Then

    Voltage_LF = InStr(NodeNumber_LF + 1, Data_Received, Chr$(10))

    If Voltage_LF > 0 Then

        Current_LF = InStr(Voltage_LF + 1, Data_Received, Chr$(10))

        If Current_LF > 0 Then

           First_ETX = InStr(Current_LF + 1, Data_Received, "ETX")

        End If

      End If

     End If

  End If


  Do While (First_STX > 0) And (First_ETX > 0) And (NodeNumber_LF > 0) And
(Voltage_LF > 0) And (Current_LF > 0)


    Data_Received = Mid$(Data_Received, First_STX + 4)

    NodeNumber_LF = InStr(1, Data_Received, Chr$(10))

    Node_number = Left$(Data_Received, NodeNumber_LF - 1)


    Data_Received = Mid$(Data_Received, NodeNumber_LF + 1)

    Voltage_LF = InStr(1, Data_Received, Chr$(10))

    Voltage = Left$(Data_Received, Voltage_LF - 1)
```

```vbnet
    Data_Received = Mid$(Data_Received, Voltage_LF + 1)

    Current_LF = InStr(1, Data_Received, Chr$(10))

    Current = Left$(Data_Received, Current_LF - 1)


    Data_Received = Mid$(Data_Received, Current_LF + 1)

    First_ETX = InStr(1, Data_Received, "ETX")

    If Len(Data_Received) > 4 Then

       Data_Received = Mid$(Data_Received, First_ETX + 4)

    End If


    ' show current data on the text boxes

    MSChart1.Column = Node_number

    Node_TextBox(Node_number).Text = _

       Node_number & vbCrLf & MSChart1.ColumnLabel & vbCrLf & Voltage & " V" &
vbCrLf & Current & " mA"


    'change the widht of the associated circle

    If (Current < 2) Then

       Shape(Node_number).Visible = False

    Else

      If Shape(Node_number).Visible = False Then

         Shape(Node_number).Visible = True

      End If

      Shape(Node_number).BorderWidth = (Current / 10) + 1

    End If


    ' buffer the new values (always at the end)

    Call Buffer_Values
```

```vb
   'Check if the packet contains all the values
   First_STX = InStr(1, Data_Received, "STX")
  If First_STX > 0 Then

     NodeNumber_LF = InStr(First_STX + 4, Data_Received, Chr$(10))
    If NodeNumber_LF > 0 Then

       Voltage_LF = InStr(NodeNumber_LF + 1, Data_Received, Chr$(10))
      If Voltage_LF > 0 Then

         Current_LF = InStr(Voltage_LF + 1, Data_Received, Chr$(10))
        If Current_LF > 0 Then

           First_ETX = InStr(Current_LF + 1, Data_Received, "ETX")
        End If

      End If

     End If

   End If


  Loop


  If (Record_request) Then

    Call Record_Data

    Record_request = False
  End If


Exit_this_sub:


End Sub
```

```vb
Private Sub Node_TextBox_Click(Index As Integer)


    If Node_TextBox(Index).Visible = True Then
        Node_TextBox(Index).Visible = False
    Else
        Node_TextBox(Index).Visible = True
    End If


End Sub




Private Sub Picture1_Click()


End Sub


Private Sub Record_Button_Click()
 Record_request = True


End Sub


Private Sub switch_Click(Index As Integer)
MSComm1.Output = Str(Index) + Chr$(10)
'MSComm1.Output = "1" + Chr$(10)
End Sub


Private Sub Timer1_Timer()
```

```
' * * * * * * * * * * * * * * * * * * * * * * * * * *


' Second Step: Update the graph/curves

' * * * * * * * * * * * * * * * * * * * * * * * * * *

  Call Update_Graph


End Sub
```

REFERENCES

[1] Parker, Danny, Hoak,David, Meier, Alan, & Brown, Richard. (2008). "How Much Energy Are We Using? Potential of Residential Energy Demand Feedback Devices." Florida Solar Energy Center/University of Central Florida.

[2] Frank, Susan. (2008). "Hydro One Networks Inc. Time-of-Use Pricing Project Results."

[3] Chevron's energy efficiency and conservation, http://www.chevron.com/globalissues/energyefficiencyconservation/

[4] Williams, Eric D., & Matthews, H. Scott. (2007). Scoping the Potential of Monitoring and Control Technologies to Reduce Energy use in Homes. IEEE.

[5] Department Of Energy, Energy Information Admission, Annual Energy Outlook 2006 with Projections to 2030, DOE/EIA-0383(2006), February 2006, http://tonto.eia.doe.gov/ftproot/forecasting/0383(2006).pdf

[6] Lawrence Berkeley National Laboratory (2004). PIER Final Report: Developing and Testing Low Power Mode Measurements Methods. Sacramento, CA, California Energy Commission.

[7] Suzanne Foster Porter, Laura Moorefield, Peter May-Ostendorp, "Final Field Research Report", Ecos Consulting, for California Energy Commission, October 31, 2006

[8] FedStats on housing, http://www.fedstats.gov/qf/states/31000.html

[9] Patrick G. Wheeler, Avery D. Schwer, Mahmoud Alahmad, Joshua M. Eiden, Andrea M. Wikerson, and Heidi A. Kuchta, "Conservation and behavior change through real-time energy monitoring", GOC_2009 – The 8th Ball State University Greening of the Campus Conference, Indianapolis, IN, 20-24 September 2009.

[10] M. Alahmad, Andrea Wilkerson, Josh Eiden (2008) "Collaboration with Industry to Promote Energy Conservation and Education", accepted to the Annual American Society for Engineering Education (ASEE), Austin, Texas, June 14-17, 2009.

[11] National Association of Home Builders, http://www.nahb.org/

[12] Energy Star, http://www.energystar.gov/index.cfm?c=products.pr_pie

[13] Farhar, Barbara C. (1989). "Effects of Feedback on Residential Electricity Consumption: A Literature Review." Solar Energy Research Institute. U.S. Department of Energy.

[14] Seligman, C. & Darley, J.M. (1977). "Feedback as a Means of Decreasing Residential Energy Consumption." Journal of Applied Psychology 62 (4): 363-368.

[15]Winett, Richard A. and Michael T. Nietzel (1975). "Behavioral Ecology: Contingency Management of Consumer Energy Use." American J. of Community Psychology, 3.2:123-133.

[16] Winett, Richard A., John H. Kagel, Raymond C. Battalio, and Robin C. Winkler (1978). "Effects of Monetary Rebates, Feedback, and Information on Residential Electricity Conservation." Journal of Applied Psychology 63:73-80.

[17] Bittle, Ronald G., Robert Balesano, and Greg Thaler (1979). "The Effects of Daily Cost Feedback on Residential Electricity Consumption." Behavior Modification 3:187-201.

[18] Winett, Richard A., Michael S. Neale, and Cannon Grier (1982). "Effects of Self-Monitoring and Feedback on Electricity Consumption." Journal of Applied Behavior Analysis 12:173-184.

[19] Van Houwelingen, Jeannet H. and W. Fred Van Raaij (1989). "The Effect of Goal-Setting and Daily Electronic Feedback on In-Home Energy Use." The Journal of Consumer Research 16, 1:98-105.

[20] Dobson, JK and JDA Griffin (1992). "Conservation Effect of Immediate Electricity Cost Feedback on Residential Consumption Behaviour." Proceedings, American Council for an Energy-Efficient Economy, 10.33-10.35.

[21] Tanabe, K (2000). "Energy Conservation Results of the Survey Project on the Status of Energy Saving in the Residential Sector." IEA Presentation on standby electricity. The Energy Conservation Center, Japan.

[22] Mansouri, I and Newborough, M (1999). "Dynamics of Energy Use in the UK Households: End-Use Monitoring of Electric Cookers. Proceedings, European Council for an Energy-Efficient Economy." Panel III, 08.

[23] Gyorok, G.; Mako, M.; Lakner, J.; , "Combinatorics at electronic circuit realization in FPAA," *Intelligent Systems and Informatics, 2008. SISY 2008. 6th International Symposium on* , vol., no., pp.1-4, 26-27 Sept. 2008.

[24] Thomas, A.G.; Berg, M.C.; , "Medium PRF set selection: an approach through combinatorics," *Radar, Sonar and Navigation, IEE Proceedings -* , vol.141, no.6, pp.307-311, Dec 1994.

[25] CEATI (2008). http://www.bluelineinnovations.com/documents/pr-ceati.pdf (Accessed: August 25, 2008).

[26] TXU Energy Power MonitorTM (2008). http://www.txupartners.com/powermonitor/. (Accessed: August 25, 2008).

[27] Botkins, David (2008). "Dominion Virginia Power Launches First of Nine Energy Conservation Pilots." Dominion Virginia Power http://www.dom.com (Accessed: August 25, 2008).

[28] NSTAR (2007). Residential Energy Efficiency Department http://www.ase.org/ (Accessed: August 25, 2008).

[28] Jackson, William G. Jr. (2008) NV Energy. A Review of Equipment Home Energy Display Devices "Description, Application and Issues." Paragon Consulting Services Inc.

[29] Nexus Energy Software, Opinion Dynamics Corporation, and Primen (2005). "Information Display Pilot: California Statewide Pricing Pilot." Southern California Edison. White paper.

[30] Ambient Energy Orb www.consumerpowerline.com (Accessed: February 1, 2009).

[31] Aztech Associates Incorporated http://www.aztechinc.com/en/ (Accessed: February 1, 2009).

[32] Blueline Innovations http://www.bluelineinnovations.com (Accessed: February 1, 2009).

[33] Wattson (Accessed: February 1, 2009).

[34] The Energy Detective www.energydetective.com (Accessed: February 1, 2009).

[35] Cent-A-Meter www.cenergies.com (Accessed: February 1, 2009).

[36] EML 2020H www.powermeterstore.com (Accessed: February 1, 2009).

[37] Itron utility AMR meter, http://www.itron.com/pages/products_detail.asp?id=itr_000240.xml

[38] Martinez, Mark S. (2006). "Residential Demand Response Technologies: A Consumer's Guide." National Town Meeting and Symposium on Demand Response. Southern California Edison.

[39] Department Of Energy, Energy Information Admission, http://eia.doe.gov/

[40] Mark W. Earley, Jeffrey S. Sargent, Joseph V. Sheehan, E. William Buss, "National Electrical Code Handbook," NEC 2008, NFPA, pp 29.

[41] ANSI/ASHRAE/IESNA Standard 90.1-2007: Energy Standard for Buildings Except Low-rise Residential Buildings, American Society of Heating, Refrigeration and Air-conditioning Engineers, Atlanta, GA.

[42] Ed Carroll, Eric Hatton, Mark Brown, "Residential Energy Use Behavior change Pilot," CMFS project code B21383, April 20, 2009 Franklin Energy

[43] U.S. Green Building Council, "Green Building Facts." U.S. Green Building Council. www.usgbc.org/ShowFile.aspx?DocumentID=5961, Accessed 7-21-2010.

[44] Venkataraman, V.; Porikli, F.; , "RelCom: Relational combinatorics features for rapid object detection," Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on , vol., no., pp.23-30, 13-18 June 2010.

[45] Giroire, H.; Le Calvez, F.; Tisseau, G.; , "Benefits of Knowledge-Based Interactive Learning Environments: A Case in Combinatorics," Advanced Learning Technologies, 2006. Sixth International Conference on , vol., no., pp.285-289, 5-7 July 2006.

[46] National Association of Home Builders, http://www.nahb.org/, Accessed 7-21-2010

[47] Szpankowski, W.; , "Analytic algorithmics, combinatorics, and information theory," Information Theory Workshop, 2005 IEEE , vol., no., pp. 1 pp., 29 Aug.-1 Sept. 2005.

[48] Beichl, Isabel; Sullivan, Francis; , "Combinatorics in Computing," Computing in Science & Engineering , vol.10, no.2, pp.12-13, March-April 2008.

[49] Berinde, R.; Gilbert, A.C.; Indyk, P.; Karloff, H.; Strauss, M.J.; , "Combining geometry and combinatorics: A unified approach to sparse signal recovery,"Communication, Control, and Computing, 2008 46th Annual Allerton Conference on , vol., no., pp.798-805, 23-26 Sept. 2008.

[50] Qin Wang; Hempstead, M.; Yang, W.; , "A Realistic Power Consumption Model for Wireless Sensor Network Devices," Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society on , vol.1, no., pp.286-295, 28-28 Sept. 2006.

[51] Karen Ehrhardt-Martinez, Kat A. Donnelly, & John A. "Skip" Laitner, "Advanced metering Initiatives and Residential Feedback Programs: A Meta-Review of Household Electricity-Saving Opportunities," American Council for an Energy-Efficient Economy, report Number E105, June 2010.

[52] Abrahamse, W., Steg, L., Vlek, C. & Rothengatter, T. (2010). A review of intervention studies aimed at household energy conservation. Journal of Environmental Psychology, 25, 273-291.

[53] Allen, D. & Janda, K. (2006). The effects of household characteristics and energy use consciousness on the effectiveness of real-time energy use feedback: A pilot study. Proceedings of the ACEEE Summer Study on Energy Efficiency in Buildings. 7.1-7.12.

[54] Brandon, G. & Lewis, A. (1999). Reducing Household Energy Consumption: a Qualitative and Quantitative Field Study. Journal of Environmental Psychology, 19, 75-85.

[55] CEATI (2008). Results of Two-Year Study Demonstrates Residential Electricity Monitors Help Homeowners Conserve Electricity in a Big Way. http://www.bluelineinnovations.com/documents/pr-ceati.pdf. Accessed June 30th 2010.

[56] Darby, S. (2003). Making sense of energy advice. Proceedings of the ECEEE Summer Study, 1217-1226.

[57] Darby, S. (2006). The effectiveness of feedback on energy consumption: A review for DEFRA of the literature on metering, billing and direct displays. Environmental Change Institute, Oxford, U.K.

[58] Farhar, B.C. & Fitzpatrick, C. (1989). Effects of feedback on residential electricity consumption: A literature review. SERIITR·254·3386. UC Category: 95. DE89000872. Solar Energy Research Institute, Golden, CO.

[59] Faruqui, A., Sergici, S. & Sharif, A. (2010). The impact of informational feedback on energy consumption. Energy, 35, 1598-1608.

[60] Fischer, C. (2008). Feedback on household electricity consumption: A tool for saving energy. Energy Efficiency, 1, 79-104.

[61] Geller, H. & Nadel, S. (1994). Market transformation strategies to promote end-use efficiency. Annual Review of Energy and Environment, 19, 301-346.

[62] Grunwald, M. (2009). Wasting our watts: We don't need new drilling or new power plants, we need to get efficient. Time, 173 (1), 32-36.

[63] Hydro One. (2006). Summary – The impact of real-time feedback on residential electricity consumption: The Hydro-One pilot.

[64] Irish, L. & Christensen, K.J. (1998). A "Green TCP/IP" to reduce electricity consumed by computers. SoutheastCon '98 Proceedings IEEE, 302-305.

[65] Koomey, J.G., Cramer, M., Piette, M. & Eto, J.H. (1995). Efficiency Improvements in U.S. Office Equipment: Expected Policy Impacts and Uncertainties. LBNL-37383. Ernest Orlando Lawrence National Laboratory, Berkeley, CA.

[66] Matsukawa, I. (2004). The effects of information on residential demand for electricity. The Energy Journal, 25, 1-17.

[67] Meier, A. & Nordman, B. (2008). Low Power Mode Energy Use in California Homes. ACEEE Summer Study on Energy Efficiency in Buildings. American Council for an Energy-Efficient Economy: Washington, DC.

[68] Meier, A. & Huber, W. (1997). Results from the investigations on leaking electricity in the USA. LBNL-40909. Ernest Orlando Lawrence National Laboratory, Berkeley, CA.

[69] Newsham, G.R. & Tiller, D.K. (1995). The energy conservation potential of power management for fax machines. Energy and Buildings, 23, 121 - 130.

[70] Newsham, G.R. & Tiller, D.K. (1994). The energy consumption of desktop computers: Measurement and savings potential. IEEE Transactions on Industry Applications, 30, 1065 - 1072.

[71] Newsham, G.R. & Tiller, D.K. (1992). A case study of the energy consumption of desktop computers. Proceedings of the IEEE Industry Applications Society, Houston, TX, October.

[72] Newsham, G.R. & Veitch, J.A. (2000). Energy-efficient lighting options: Predicted savings and occupant impressions of lighting quality. Clima2000 Proceedings, Brussels, Belgium.

[73] Nordman, B., Meier, A. & Piette, M. (2000). PC and monitor night status: Power management enabling and manual turn-off. Proceedings of the 2000 ACEEE Summer Study on Energy Efficiency in Buildings, 7.89-7.99.

[74] Okamura, H., Dohi, T. & Osaki, S. (2000). Performance analysis of a transaction-based software system with shutdown. Proceedings of the 2nd International Workshop on Software Performance. 168-179, Ottawa, ON Canada.

[75] Parker, D.S., Hoak, D. & Cummings, J. (2008). Pilot evaluation of energy savings from residential energy demand feedback devices. FSEC-CR-1742-08. Florida Solar Energy Center, Cocoa, FL.

[76] Parks, B. (2009). Home energy dashboards. Make: Technology on Your Own Time. 18, 48-51.

[77] Roth, K. & Brodrick, J. (2008). Home energy displays. ASHRAE Journal, July, 136-138.

[78] Roberson, J.A., Webber, C.A., McWhinney, M.C., Brown, R.E., Pinckard, M.J. & Busch, J.F. (2004). After-hours Power Status of Office Equipment and Energy Use of Miscellaneous Plug-Load Equipment. LBNL-53729-Revised. Ernest Orlando Lawrence National Laboratory, Berkeley, CA.

[79] Roth, K.W. & McKenney, K. (2007). Energy consumption by consumer electronics in US residences. TIAX Reference D5525. TIAX LLC, Cambridge MA.

[80] Ueno, T., Sano, F., Saeki, O. & Tsuji. K. (2006). Effectiveness of an energy-consumption information system on energy savings in residential houses based on monitored data. Applied Energy, 83, 166-183.

[81] Google powermeter, http://www.google.com/powermeter/about/, Accessed 7-21-2010

[82] Conserve Energy, Save Money-Microsoft Hohm, http://www.microsoft-hohm.com/, Accessed 7-21-2010.

[83] Tiller, D.K. & Newsham, G.R. (1993). Automatic power management of desktop computers. Proceedings of the IEEE Industry Applications Society, Toronto, ON, October.

[84] Tiller, D.K. & Newsham, G.R. (1992). Switch off your desktop computer and save energy! Proceedings of the CIB 92 World Building Congress, 18-22 May 1992, Montreal, Canada.

[85] Department of Energy, Building America, Research Toward Zero Energy Homes, http://www1.eere.energy.gov/buildings/building_america/about.html, Accessed 7-21-2010.

[86] UNL - Architecture Research and Creative, http://architecture.unl.edu/programs/research/ZNETH.shtml, Accessed 7-21-2010.

[87] USGBC: LEED, http://www.usgbc.org/DisplayPage.aspx?CategoryID=19, Accessed 7-21-2010.

[88] National institute of building Science, Building Smart alliance, national BIM Standard, http://www.buildingsmartalliance.org/index.php/nbims/, Accessed 7-21-2010.

[89] H. Edward Goldberg, "The BIM (Building Information Model): What is it and where are we going?," AIA, AUGIWorld, Nov/Dec 2005, pp. 8-9.

[90] Autodesk Revit White paper, "Building Information Modelling for sustainable design". 2005

[91] Brychta, Markus; Dubisch, Florian; Stift, Florian; Palensky, Peter; , "QUEEN — A novel design flow and decision support tool for sustainable buildings," IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society , vol., no., pp.2448-2453, 7-10 Nov. 2010.

[92] Kamiyoshi, Yuto; Nakabe, Tomohisa; Mine, Gouki; Nishi, Hiroaki; , "Construction of energy measuring system in a University for Cluster Energy Management System," IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society , vol., no., pp.2423-2429, 7-10 Nov. 2010.

[93] Suhara, Yukio; Nakabe, Tomohisa; Mine, Gouki; Nishi, Hiroaki; , "Distributed demand side management system for home energy management," IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society , vol., no., pp.2430-2435, 7-10 Nov. 2010.

[94] Zaidi, Adeel Abbas; Kupzog, Friederich; Zia, Tehseen; Palensky, Peter; , "Load recognition for automated demand response in microgrids," IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society , vol., no., pp.2442-2447, 7-10 Nov. 2010.

[95] J. she Jin; J. Jin; Y.Wang; K. Zhao; J. Hu; , "Development of Remote-Controlled Home Automation System with Wireless Sensor Network," Embedded Computing, 2008. SEC '08. Fifth IEEE International Symposium on , vol., no., pp.169-173, 6-8 Oct. 2008.

[96] J. Y. Pan, W. T. Lee, and N. F. Huang, "Providing multicast short message services over self- routing mobile cellular backbone network," IEEE Trans. Veh. Technol., vol. 52, no. 1, pp. 240-253, January 2003.

[97] Dr. M. S. H. Khiyal, "SMS Based Wireless Home Appliance Control System (HACS) for Automating Appliances and Security," Issues in Informing Science and Information Technology, vol 6, 2008.

[98] C. H. Lien, C. H. Lin, Y. W. Bai, M.F. Liu and Ming-Bo Lin, "Remotely Controllable Outlet System for Home Power Management," Consumer Electronics, pp. 1-6, 2006.