Rajesh Singh, Siddharth Sobti, Naman Bhandula, Shashank Agarwal, Laxman Poonia/ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 4, July-August 2012, pp.051-055

Wireless Micro Power Meter System up to 100 W Load Simulation and Design using 2.4 GHz Transceiver Module

Rajesh Singh[1], Siddharth Sobti[2], Naman Bhandula[3], Shashank Agarwal[4], Laxman Poonia[5]

[1](Head, Robotics Institute R&D,University of Petroleum and Energy Studies,India)
[2](Student, Power System Engineering,University of Petroleum and Energy Studies,India)
[3](Student, Power System Engineering,University of Petroleum and Energy Studies,India)
[4](Student, Power System Engineering,University of Petroleum and Energy Studies,India)
[5](Student, Power System Engineering,University of Petroleum and Energy Studies,India)

Abstract

These days no building is without power supply. Every home and industry is consuming a large amount of power and keeping the log of this energy is a great headache for a person keeping record. This project proposes a model for logging of this data of units consumed by the user remotely. These days the energy meters used are digital and hence a digital system is presented to remotely log the data and recording it. This system applies ZigBee communication protocol and uses the IC-CC2500, 2.4GHz(ISM) as RF Transceiver.

1. Introduction

The system is based on ATmega16 microcontroller from Atmel. The microcontroller is powered by powerful AVR core. The controller is capable of processing 16MIPS with 16MHz crystal. The system comprises of the power measurement system. The measured units are read by the microcontroller which is capable of transmitting it to a remote station with the help of a suitable protocol like ZigBee, Ethernet or WLAN. In this system we have used ZigBee for remote monitoring of power consumed. ZigBee has the characteristics of low power consumption, low cost, flexible structure and accurate measurement, and it can achieve the long-distance Energy meter monitoring of the bus bar junction in real time. The wireless Energy meter sensor node senses and transmits the variations in the local Energy meter to the central computing unit placed within the range. The central computing unit receives the data and displays the data continuously on MATLAB data logger on PC through USB to serial cable. ZigBee are inexpensive, Low-power communication devices that can be deployed throughout a physical space, providing dense sensing close to physical phenomena, processing and communicating this information and coordinating actions with other nodes. The wireless sensor networks based on ZigBee has the characteristics of insulation, strong electromagnetic immunity, low power and high accuracy, solving the problem of easy breaking, easy aging, saving the trouble of wiring, ensuring the reliability and safety of the operation. The Energy meter monitoring system building with this wireless sensor networks can achieve real time data collection, analysis and monitoring of the operating Energy meter within the region of network , while determining fault point in time to perform early warning and alarm.

The presented model is sensitive to micro-unit consumption of power, the readings are noted in a very precise manner, hence the accuracy of readings is very high. The readings start off as micro units then to milli units and finally to unit power consumption.

2. Block Diagram

A micro controller (ATmega 16) receives Energy meter information from Energy meter sensor. This Energy meter information is transmitted to the coordinator and the transceiver. Now, the nodes Transceiver send a signal to the coordinator Transceiver to share the Energy meter information in the wireless network. We consider a network of one slave nodes using Zigbee module and one master node (sink).Slave node will send the acquired data to the master node. The master node is interfaced with PC with the help of USB to serial cable. The result can be logged of with the help of MAT lab (GUI) on PC. The program of each node is written on embedded C through coder and debugger AVR studio 4 and compiling through PONY-PROG software.

Fig. 2.1 and Fig. 2.2 shown below shows the block diagram for the proposed model.

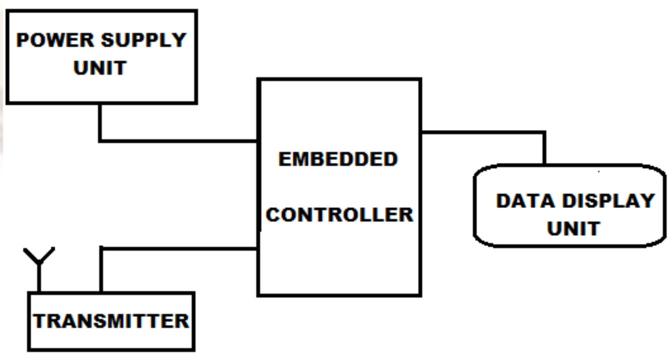

Fig.1: Block diagram for power logging

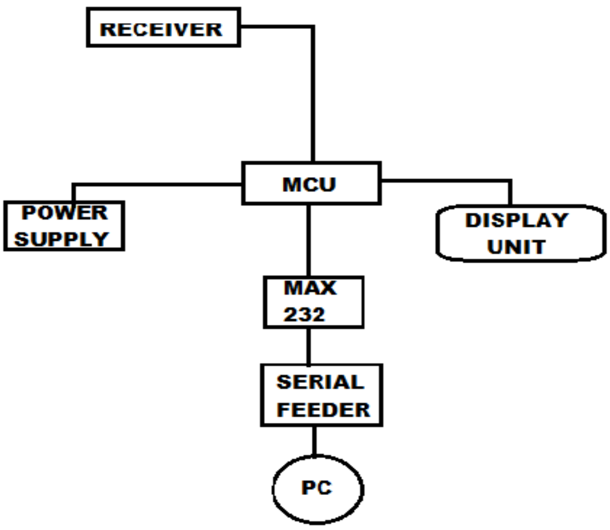

Fig.2: Block diagram for central data logging system

3. Hardware Development

The power measuring unit measures the power from main line. This data is fed to microcontroller. Microcontroller fetches the data and transmits it over the network. The following is the list for the components used in the proposed model:

- ATmega16/32
- MAX232
- Power supply units
- Data logging systems
- ZigBee modules
- LCD Display

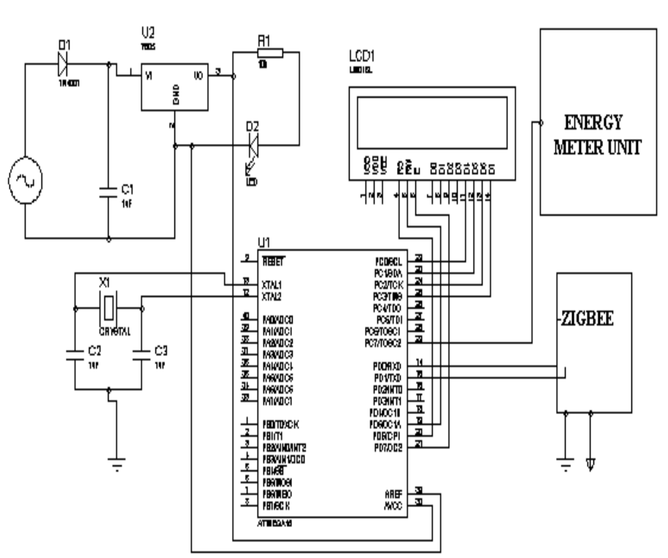

Fig.3: Power suppy interfacing with ATMEGA16 with LCD

3.1. Atmega16

It is a microcontroller from Atmel which is powered by the AVR core. It is an 8-bit,low powered microcontroller with 16 kilobytes in-system self programmable flash. This core is capable of running 16MIPS with a 16MHz crystal. It has an advanced RISC architecture with 32 X 8 general purpose working registers. The microcontroller features programmable serial USART and master/slave SPI serial interface. It has 32 programmable I/O lines and 40 pin PDIP. It is capable of executing one instruction per cycle.

3.2. MAX232

This is level converter IC from MAXIM which is used to make logic compatibility between TTL and RS232 logic. The IC converts the 5V logic into a 8V negative logic. This converter is located between the almega16 microcontroller and the zigbee module, the microcontroller uses TTL logic whereas the zigbee module uses RS logic. The main purpose of this converter is to convert the TTL logic to RS logic.

3.3. POWER SUPPLY UNIT

This unit is basically designed to power up the node 1 and node 2. This provides 5 V, 500mA output to drive the nodes. Here, the AC voltage at 220V is stepped down to 20V using a 220/20V step down transformer. This AC voltage at 20V is fed to rectifier that converts it to DC voltage and is then filtered using 40 Farad shunt capacitor. The filtered DC voltage is then regulated using a 7805 regulator, and is then supplied to the the microcontroller at 5V, 500 mA.

3.4. LCD

This is most widely used display device for embedded systems. The LCD unit receives character codes (8 bits per character) from a microprocessor or microcomputer, latches the codes to its display data RAM (80-byte DD RAM for storing 80 characters), transforms each character code into a 5 ´ 7 dot-matrix character pattern, and displays the characters on its LCD screen.

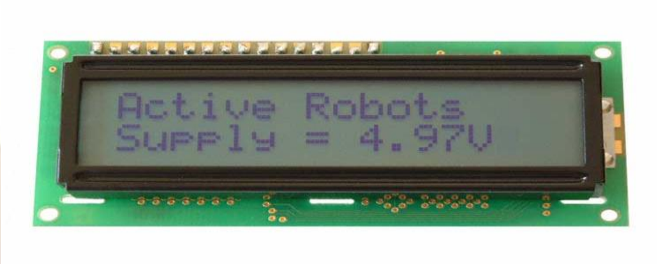

Fig.4 LCD 16x2

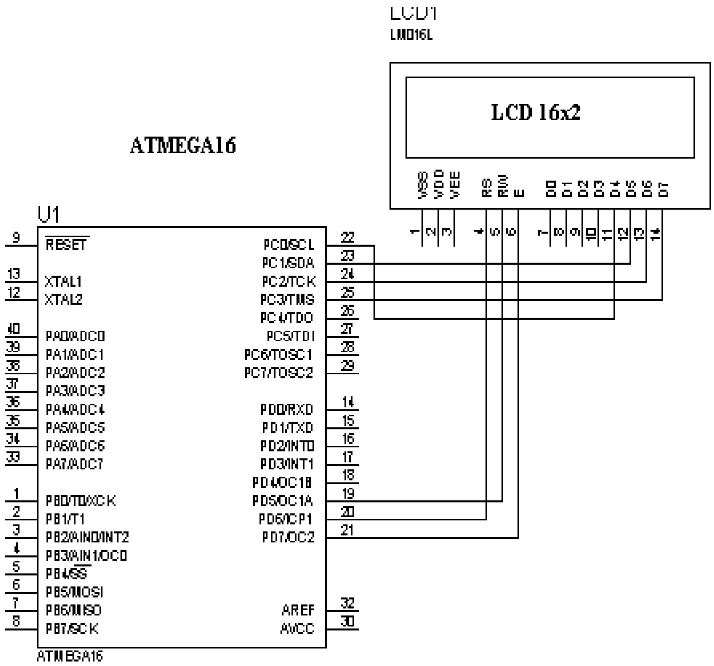

Fig.5 LCD interfacing with ATMEGA16

3.5. ZigBee Modules

This is the radio frequency transceiver module, which can facilitate the OEM designers to design their remote control applications in the quickest way. The circuit is designed with SMD components and the module size is small enough to be able to be fitted in almost any application. These modules are based on IC CC2500 by chipcom. It works at frequency of 2.4GHz, bandwidth of 250 kbps at LAN 802.15.4 protocol. The device works in half duplex mode. Zigbee modules are low power consuming devices, i.e. they can even work on battery power for long durations. These modules are one time expenditure devices, they do not require constant maintenance. These devices have no moving parts and are rugged in construction hence can tolerate rough usage.

Fig.6 Zigbee view

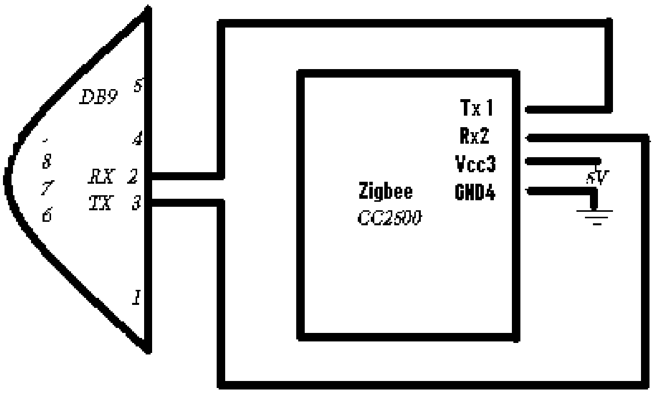

Fig.7 Zigbee interfacing with DB9

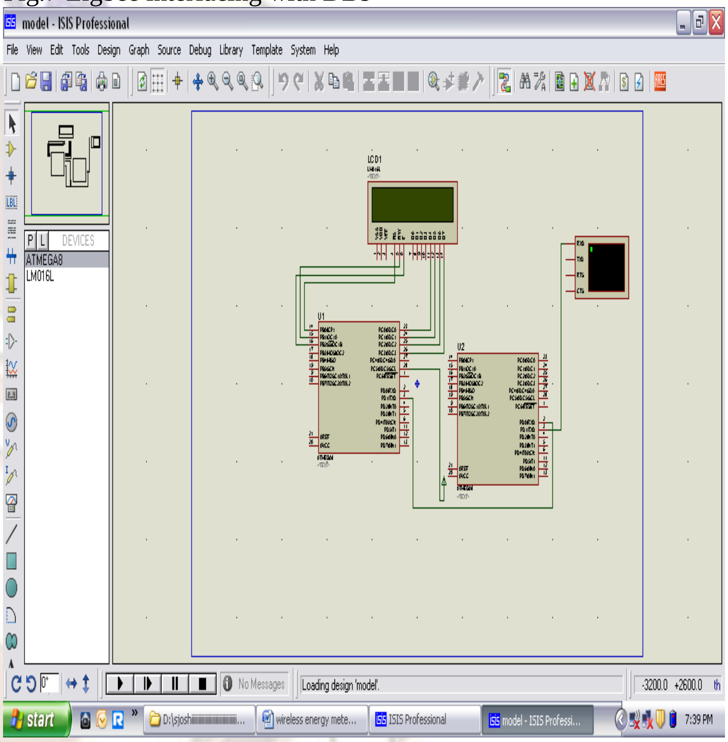

Fig.8 Simulation model of proposed model using Protesus trial version

4. Software Development

The firmware for the model is developed using C programming language. The binary code is generated with the help of WinAVR compiler based on GCC port by GNU. The IDE used is AVR Studio.

The data logging system may use any of software that is capable of logging data from a serial port. It may be a user created software or a standard software like HyperTerminal. Microcontroller has been programmed to test the hardware as well to achieve the goal of WSN application,which involved the following steps

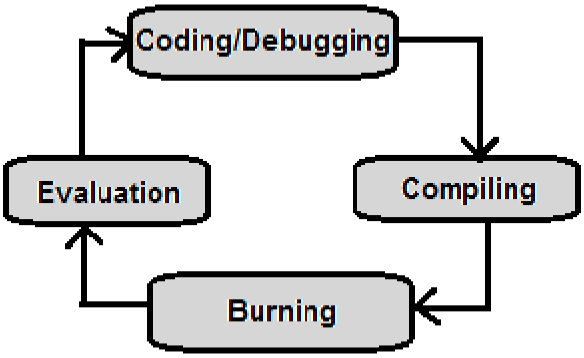

Fig.9 Steps for software development

4.1. Coding / Debugging

Coding / debugging in a high-level language (such as C, or Java) or assembler. A compiler for a high level language helps to reduce production time. To program the microcontrollers the WinAVR [2] was used. Although inline assembly was possible, the programming was done strictly in the C language. The source code has been commented to facilitate any occasional future improvement and maintenance. WinAVR is a suite of executable, open source software development tools for the Atmel AVR series of RISC microprocessors hosted on the Windows platform. It includes the GNU GCC compiler for C and C++. WinAVR contains all the tools for developing on the AVR. This includes AVR-gcc (compiler), AVR-gdb (debugger) etc. Test Source Code has been written in C Language to test the microcontroller.

4.2. Compiling

The compilation of the C program converts it into machine language file (.hex). This is the only language the microcontroller will understand, because it contains the original program code converted into a hexadecimal format. During this step there were some warnings about eventual errors in the program.

4.3. Burning

Machine language (hex) file of compile program burned into the microcontroller's program memory is achieved with a dedicated programmer, which attaches to a PC's peripheral. PC's serial port has been used for the purpose. In the present work the Ponyprog programmer has been used to burn the machine language file into the microcontroller's program memory. Ponyprog is serial device programmer software with a user-friendly GUI framework available for Windows95/98/ME/NT/2000/XP and Intel Linux. Its purpose is reading and writing every serial device. It supports I²C Bus, Micro wire, SPI eeprom, and the Atmel AVR and Microchip PIC microcontroller. The microcontrollers were programmed in approximately two seconds with a high speed-programming mode. The program memory, which is of Flash type, has, just like the EEPROM, a limited lifespan. On the AVR microcontroller family it may be reprogrammed up to a thousand times without any risk of data corruption.

Atmega16Programmer (ISP) which is used to burn the program into AVR microcontrollers is shown in Fig. 10.

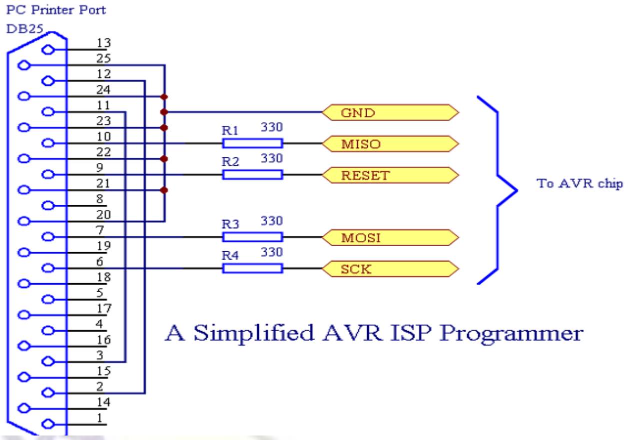

Fig.10 AVR ISP programmer

4.4. Evaluation

If the system performs all the required tasks and behaves as expected the software development phase is over. If not, the whole procedure will have to be repeated again. One of the difficulties of programming microcontrollers is the limited amount of resources the programmer has to deal with. In PCs resources such as RAM and processing speed are basically limitless when compared to microcontrollers. In contrast to a PC, the code on microcontrollers should be as low on resources as possible.

We have used our fully buffered, interrupt driven USART library for usart related job. The library comes in two files.

- USART.c
- USART.h
- Main_energymT.c
- Main_energymR.c

5. Set up of the system

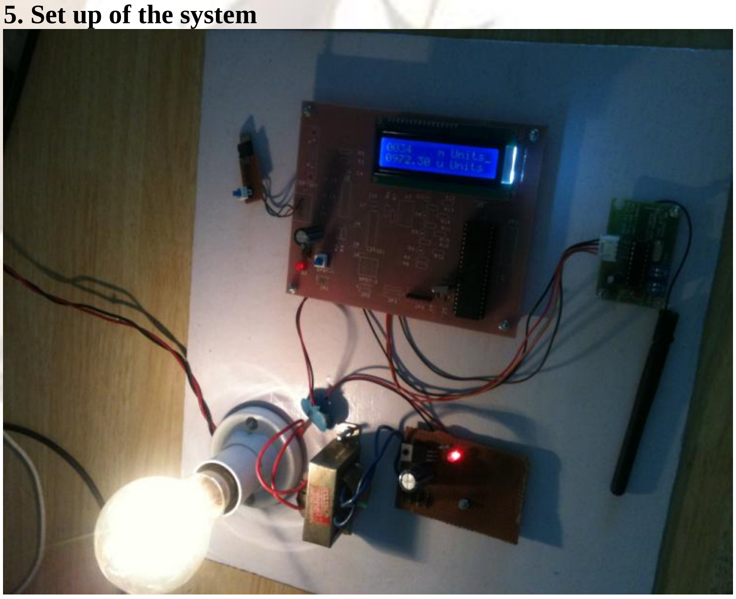

Fig.11 Master Node

The master node shown in Fig.11 consists of the transmiting end of the customized meter reading hardware which contains a 100w bulb, a power supply unit, the atmega16 controller, and the zigbee module as the main components.

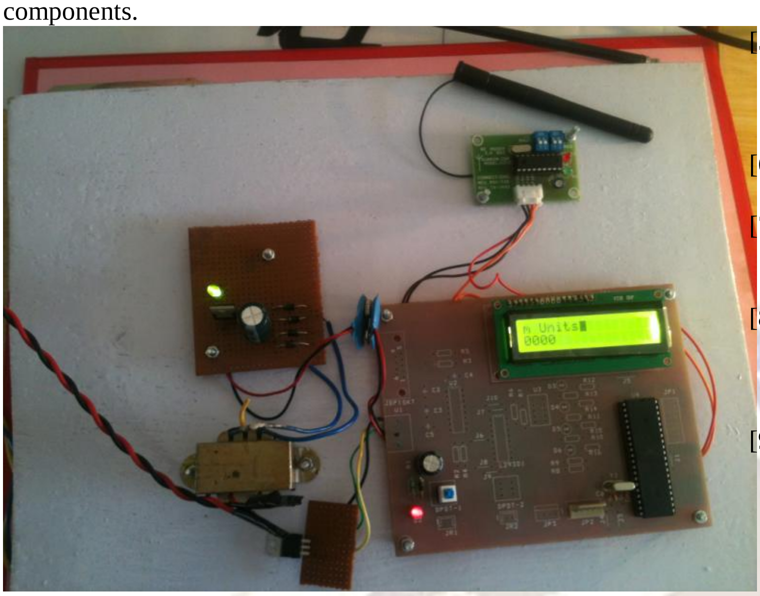

Fig.12 Slave Node

The slave node shown in Fig.12 consists of receiving end of the customized meter reading hardware which contains the power supply unit, the atmega16 controller, the LCD display unit, and the zigbee module as its main components.

6. Future Scopes

The information logged by the proposed system can be transmitted over the internet. It may be used there to log the data into the used account and online dispatching of the electricity bills. Unification of all metering devices to one central coordinating node can be achieved.

In the future these metering devices can be used in many other fields such as gas metering, water usage in domestic and industrial use, companies can benefit from these close range metering devices and save costs and minimize errors in metering.

7. Conclusion

This system, comparing with traditional manual inspection and large scale wiring, the accuracy of data acquisition and the real-time of transmission is improved significantly, and its construction is flexible, measurement is precise, operation is simple and energy is saved for entire monitoring system. The major savings are in the form of expenses to monitor every single metering device accompanied by savings in manual labour and negligible error in readings procured by personnel. It also eliminates the chances of manual error in meter reading.

References

[1] F.L.Lewis, *Wireless Sensor Networks-Chapter 4*, Smart environments:Technologies, Protocols, and Applications Journal.

[2] Mike Horton and John Suh, A Vision for Wireless Sensor Networks, *IEEE transactions on Industrial Electronics 0-7803-8846-1/05, 2005.*

[3] Atmel Documentation: http://www.atmel.com

[4] Extreme Electronics: http://www.extremeelectronics.co.in

[5] A.Flammini, D.Marioli, E.Sisinni, A.Taroni, A real-time wireless sensor Network for temperature monitoring, *IEEE Transactions on Industrial Electronics 1-4244-0755-9/07 2007*

[6] Das, V.V., *Wireless communication system for energy meter reading* , ACEEE, Trivandrum, India

[7] Bharath, P.Ananth, N.Vijetha, S. Prakash, K.V.J., *Wireless automated digital energy meter*, JSS Acad. of Tech. Educ., Bangalore

[8] Xiaofan Jiang, Dutta P. ,Culler D. , Stoica I, *Micro Power Meter for Energy Monitoring of Wireless Sensor Networks at Scale*, Univ. of California, Berkeley

[9] Kistler R., Bieri M., Wettstein R., Klapproth A. , CEESAR, *Tunneling Smart Energy protocols over ZigBee*, iHomeLab, Lucerne Univ. of Appl. Sci. & Arts, Horw, Switzerland