Middle-East Journal of Scientific Research 20 (12): 2434-2438, 2014
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2014.20.12.354

Wi-Fi Energy Meter Implementation Using Embedded Linux in ARM 9

S. Philomina and B. Karthik

Departmemt of ECE,
Bharath University, Chennai, India

Abstract: This paper introduces the design and implementation of a Wi-Fi wireless meter reading system. The design of management software and wireless terminal including hardware and embedded software are introduced in detail. ARM9 and Linux are adopted for the implementation of wireless terminal. The communication between wireless terminal and management software is realized by the program based on Winsock. Access2003 is used for the storage of information. Smart Message Language (SML Protocol) is used to design the data structure aiming to make data communication maximally simple and suitable for implementation in low-power embedded systems and solve the communication problem between different equipment. A Wi-Fi network experiment system including a intelligent meter, a wireless terminal, an access point and the management software is constructed to test the performance of the wireless meter reading system and the result shows that the meter data can be correctly received and updated every two seconds which indicates that the system has good communication performance and reaches the purpose of meter reading, efficiently solving the low safety and efficiency problems of traditional meter reading

Key words: Wireless Terminal · Management Software · Smart Message Language · Data Structure

INTRODUCTION

Wireless Meter Reading is a process that the meter data is read and processed automatically via special equipment using wireless communication and computer network technology. Compared with the traditional meter reading, it not only effectively saves human resources but also save the wiring cost and helps the management department find problems in time and take appropriate measures to deal with [1][2]. WI-FI, the wireless communication technology used in this system is also called 802.11 standards and has the advantages of high transmission rate, convenient networking, bestrow scope, strong anti-interference capability and low price. Wi-Fi network construction cost is lower than that of traditional LAN, requiring the installation of a certain number of wireless Access Point to satisfy the signal coverage of designated area [3]. The main part of the design of the system is the management software and WI-FI wireless terminal including the hardware and embedded software. One of the most important problems of wireless meter reading is that products of manufactures are lack of interoperability, so it is necessary to define a unified data structure and protocol for communication. Smart Message Language (SML), a German national standard, is created against the background of drawing up a specification that lays down a communication protocol for applications in the environment of data procurement and equipment parameterization. SML is used in this system for the design of data structure in the communication between Wi-Fi wireless terminal and management software [4].

The outline of the paper is as follows. The overall structure is described in section 2. The design of the wireless terminal is described in details in section 3.In section 4 and section 5, the design of data structure based on SML and management software is given. Section 6 gives the experimental results and analysis [5]. Conclusions are summarized in section 7.

The Overall Structure: The Wireless Meter Reading System based on WI-FI communication technology is made up of Intelligent Meter, Wireless Terminal, Wireless Repeater, Access Point and Control Center. Figure1 illustrates the overall structure of the system. The system can be divided into three main parts: Intelligent meter, Control Center and WI-FI Wireless Network which is made up of Wireless AP, Wireless Repeater and Wireless

Corresponding Author: S. Philomina, Departmemt of ECE, Bharath University, Chennai, India.

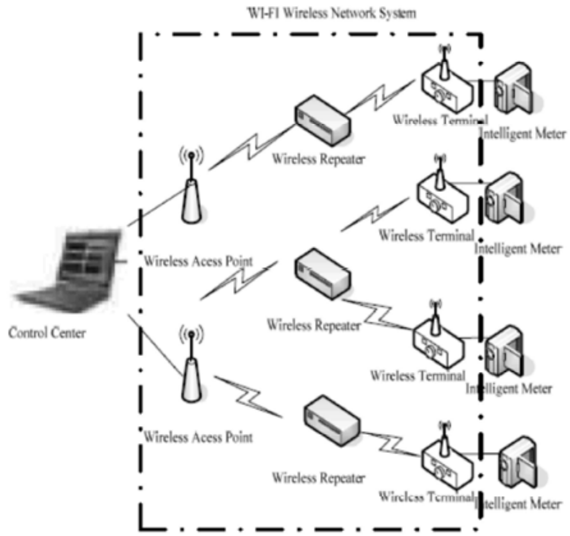

Fig. 1: Meter Reading System Architecture

Terminal. The WI-FI wireless terminal which is connected to intelligent Meter by RS485 collects the information of the intelligent meter via Modbus/TCP protocal. The data is then sent to the Access Point after being processed. The communication between Access Point and control center is via TCP/IP protocal. The access point sends the data to the control center where the data is processed and stored into the database. Users can get the information by using the management software which is the core of control center. Control Center sends command such as data procurement, data storage, alarming and so on to control the whole system [4].

The function of each part of the system is as follows: The main function of wireless terminal is connecting the intelligent meter and the WI-FI Wireless LAN, transmitting the meter data and information to the Wireless LAN and receiving commands. To guarantee the communication quality and rate, the communication rate is set to 100kpbs and the size of data is limited to 512byte/point. The system specifications are described in details in Table 2.Considering the privacy and security, WPA and Address Code Check is used [6].

The Design of Wireless Terminal

A. The Design of Hardware: The WI-FI wireless terminal communicates with the intelligent Meter by RS485 interface and sends data to wireless access point. The hardware mainly includes core processor, power supply circuit, Wi-Fi wireless module, storage unit, LED indicator light, reset circuit, RS485 interface and ethernet interface. The hardware structure is illustrated in Figure 2.

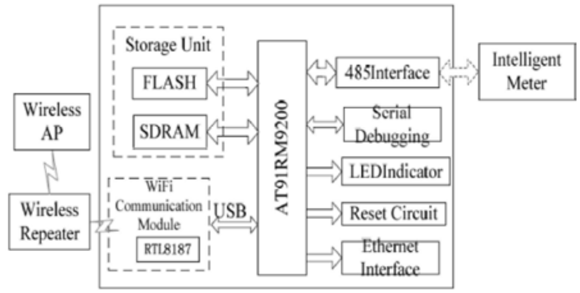

Fig. 2: Hardware structure diagram

AT91RM9200 processor of Atmel Company is used for the core processor. AT91RM9200 microprocessor enlarges the SDRAM memory and FLASH memory outside. SDRAM memory stores the program running code and FLASH memory stores data that is collected. When the task is running, the program code from FLASH memory will be copied to the SDRAM memory for operation and this can improve the speed of the system. The reset circuit is made up of hardware reset circuit and software reset circuit. Hardware reset circuit uses IMP811S chip which can ensure the reliable reset while software reset circuit adopts the most simple RC reset circuit which is mainly used to restore initial set. Serial interface is mainly used for debug [5].

Power supply system is the key of the device and it is the basic guarantee of the stable operation of the system. This system adopts the regulated 24V DC source. Power supply module adopts 24S05-6W isolation power module whose input voltage is 18 - to 36VDC and output voltage is 5VDC. 5V voltage is converted to 3.3V by 1117M-3.3.

Wi-Fi wireless module adopts G-SKY wireless transmitting module whose transmitting chip is RTL8187 chip of REALTEK Company. TL8187 chip uses IEEE802.11g^draft2.0?standard and can be compatible well with IEEE802.11g/IEEE802.11b standard.USB2.0 physical layer is integrated in this module which can be connected to the core processor directly through USB interface. It has the advantages of reducing the blind spots of the coverage of the area and expanding the coverage of wireless signals effectively which makes it a low cost and high level of integration wireless module.

The Design of Embedded Software: The design of wireless terminal software mainly includes three parts: the transplantation of operating system, the transplantation of Wi-Fi wireless module drive and the design of application.

Operating System Transplantation: Wi-Fi wireless terminal uses Linux2.6 as operating system. Linux has the features of widely hardware support which means it can run on every popular CPU. It also has the advantages of perfect network communication and file management mechanism, open source and efficient real-time kernel cutting. Operating system transplantation includes loader (Bootloader), the kernel cutting, configuration and compile and root file system transplantation. Guide loader is in charge of the target boards initialization when powered up, loading kernel to memory, transferring guide parameters to kernel and operating kernel. The kernel cutting and configuration must ensure system functions firstly and ensure the normal operation of every hardware circuit including the use of I/O port and the drive of serial port, USB and Wi-Fi wireless module. “Make menuconfig” is used to configure the kernel and add the support of the file system Ramdisk, module loading and unloading, network optimization, PCMCIA bus and some basic equipment driver support. After the configuration, “make uImage” command can be used to compile to generate ‘uImage’ files which are cross compiled to generate binary files. The binary files are compressed by zip and then translated to identifiable ‘uImage’ such as U-Boot which then can be download to hardware platform to run. Considering that AT91RM9200 platform only neither has “NOR Flash”, the root file system adopts JSSF2 file system which is developed for flash and have permanent features. The bin/,dev,/etc,/lib,/proc,/sbin and /usr of root file system is necessary while others can be adjusted according to the requirements. Considering the partition uses of NFS file system and NOR Flash need hard points, /mnt directory is kept and var directory is used to store temporary data and root directory stores some root user configuration files and tmp directory stores temporary files[6].

Qt ("cute", or unofficially as Q-T cue-tee) is a cross-platform application framework that is widely used for developing application software with a graphical user Interface (GUI) (in which cases Qt is classified as a widget toolkit) and also used for developing non-GUI programs such as command-line tools and consoles for servers. Haavard Nord and Eirik Chambe-Eng (the original developers of Qt and the CEO and President, respectively, of Trolltech) began development of "Qt" in 1991.

Qt is free and open source software. All editions support many compilers, including the GCC C++ compiler and the Visual Studio suite.

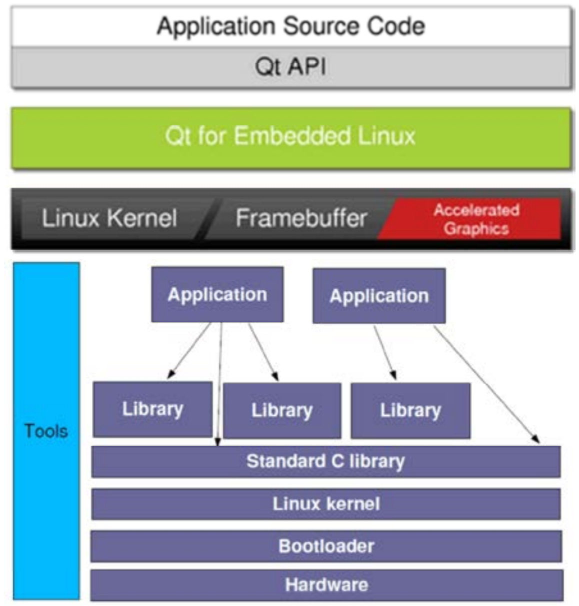

Fig. 3:

On August 9, 2012, Digia acquired Qt software technologies from Nokia. About 125 Qt developers will be transferred to Digia, with the immediate goal of bringing Qt support to android, iOS and Windows 8 platforms.

Qt is most notably used in VLC media player, Safari Browser, Autodesk Maya, The Foundry's Nuke, Adobe Photoshop Elements, Skype, Virtual Box and Mathematica and by the European Space Agency, DreamWorks, Google, HP, KDE, Lucas film, Panasonic, Philips, Samsung, Siemens, Volvo, Walt Disney Animation Studios and Research In Motion. The Opera web browser also uses Qt, but only as an interface to the Linux platform.

Embedded Linux is the use of Linux in embedded computer systems such as mobile phones, personal digital assistants, media players, set-top boxes and other consumer electronics devices, networking equipment, machine control, industrial automation, navigation equipment and medical instruments which have

- Low Memory Resources – RAM, Hard Disk
- Low Processing Power – CPU (400 MHz, <= 1GHz)
- Compact Size
- Low Cost Limitation
- Higher Performance and Efficiency in Speed, Power

Due to its low cost and ease of customization, Linux has been shipped in many consumer devices.

Linux is available for many architectures and an obvious candidate for an embedded system and it already is being used widely in this area. Its open nature makes it particularly attractive to developers. Development tool suites have begun to appear in response to the perceived need, although one can work without such luxury and employ less integrated tools already available in Linux. New embedded systems companies using Linux have opened for business and various older embedded systems companies have added Linux to their product line.

Qt for Embedded Linux is a C++ framework for GUI and application development for embedded devices. It runs on a variety of processors. Qt for Embedded Linux provides the standard Qt API for embedded devices with a lightweight Graphics system.

Qt is a cross-platform application and UI framework for writing web-enabled applications for desktop, mobile and embedded operating systems. This page contains links to articles and overviews explaining key components and techniques used in Qt development.

One of the most important issues with embedded systems is the need for a real-time operating system. The definition of real-time here varies quite a bit. To some people, real-time means responding to an event in the one-microsecond range, to others it is 50 milliseconds. The hardness of real-time also varies quite a bit. Some systems need hard real-time response, with short deterministic response latencies to events. However, on many systems, when analyzed closely, we see a response time requirement that is actually near real-time.

Often the real-time requirement is a tradeoff of time and buffer space. With memory getting cheaper and CPUs getting faster, near real-time is now more typical than hard real-time and many commercial operating systems that claim to be real-time are far from being hard real-time. Usually, when you get into the detailed design of these systems, there are warnings that the drivers' interrupts and applications must be very carefully designed in order to meet real-time requirements.

RT-Linux (Linux with real-time extensions) contains time critical functions to provide precise control over interrupt handling, through the use of an interrupt manager and does a good job of making sure that critical interrupts get executed when needed. The hardness of this approach depends mostly on the CPU interrupt structure and context-switch hardware support. This approach is sufficient for a large range of real-time requirements. Even without the real-time extensions, Linux does pretty well at keeping up with multiple streams of events. For example, a Linux PC system on a low end Pentium is able to keep multiple 10BaseT interfaces executing effectively, while simultaneously running character-level serial ports at a full 56KBPS without losing any data.

Some real-time hardware and software Linux APIs to consider are RTLinux, RTAI, EL and Linux-SRT. RTLinux is a hard real-time Linux API originally developed at the New Mexico Institute of Technology. RTAI (DIAPM) is a spin-off of the RTLinux real-time API that was developed by programmers at the Department of Aerospace Engineering, Polytechnic Politecnico di Milano (DIAPM). EL/IX is a proposed POSIX-based hard real-time Linux API being promoted by Red Hat. And Linux-SRT is a soft real-time alternative to real-time APIs, which provides performance-enhancing capabilities to any Linux program without requiring that the program be modified or recompiled.

Transplantation of Wi-fi Wireless Module: Wi-Fi wireless module uses RTL8187L module of Realtek. Linux-2.6.32.2 kernel provides the drive support of RTL8187 and RTL8187B wireless module, but this dive dose not perform well on RTL8187L wireless module and the function support is not perfect. At the same time, Realtek official provides the support of Linux of this wireless module and RTL8187L driver source code, making the support of RTL8187L more perfect. As a result, drive provided by Realtek official is used and only needs some modification. RTL8187L driver source code is downloaded and rt18187 and ieee80211 files are copied to driver/net/wireless. Makefile and Kconfig file is modified making it added to kernel and compiled.

Add "obj-$(CONFIG_RTL8187L)+= rtl8187/" to "driver/net/wireless/Makefile" file and add "config RTL8187L" and "tristate "Realtek 8187L USB support"" to "driver/net/wireless/Kconfig" file. New "Makefile"file under "drivers/net/wireless/rtl8187" directory and add compiler marks and target file of core program of RTL8187L drive to it. IEEE802.11 protocal support must also be added and configuration module is named "RTL8187L", the same as that of Kconfig. Configuring RTL8187 drive by "Make menuconfig" and compiling to generate.ko module file by "make module" and downloading.ko drive program to hardware platform can support the operation of Wi-Fi module very well.

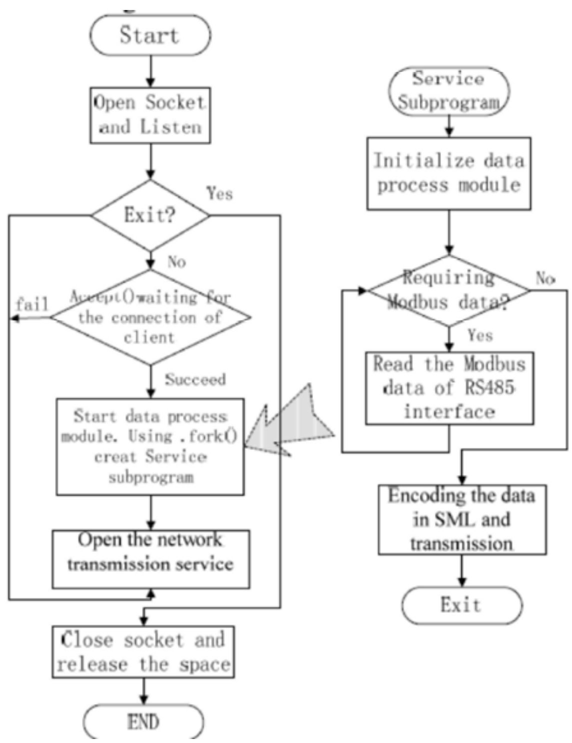

Fig. 4: Flowchart of Application Program

The Design of Application: The design of application mainly includes network communication and RS485 communication. TCP/IP is used in the network communication and data structure is designed based on SML which will be introduced in section4. Socket is used for network programming. Open the socket first and initialize it to data stream socket and listen to it. Accept () function then waits for the connection of the client. If the connection is successful, the data will be sent. The flowchart of the program is illustrated in Figure4.

RESULTS

In order to test the communication performance of the system, network experiment system as Figure 8 shows is used. The equipment's are PC, wireless AP TL-WA501G+, wireless Wi-Fi terminal and intelligent meter. Open management software and configuration the wireless access point and the open the DHCP dynamic IP service functions, the IP of Wi-Fi terminal finally got is 192.168.1.101.

CONCLUSION

The IP of management software is 192.168.1.101 and the port is 4000. The data can be received after the server being connected. The data is shown in Figure 9. The voltage that the meter connected to fluctuates in a small scope around 220V and is the same as meter data. Embedded software and management software runs normally and the data acquisition module and network communication module are reaching the requirements.

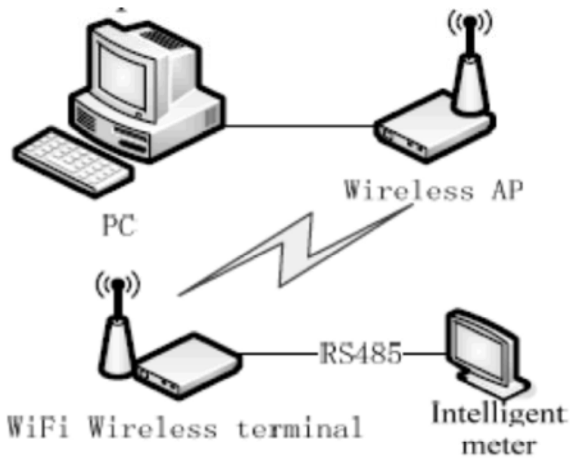

Fig. 5: Experiment System

REFERENCES

1. Duan Qichang and Dong Ping, March, 2009. Network Data Transmission and Information Exchange of the Wireless Meter Reading System, Techniques of Automation and Application, 28: 39-41.
2. Das, V.V., 2009. Wireless Communication System for Energy Meter Reading, Advances in Recent Technologies in Communication and Computing, 2009. Abstract| Full Text:PDF(439KB), pp: 896-898.
3. Chen Wenzhou, 2008. Research and Application of Wi-Fi Technology, Data Communication, 2: 14-17.
4. Fangchang Zhang, Xiaodan Liu and Weiyan Hou, 2010. Design and Implementation of Measurement Meter Reading System Based On Wi-Fi," Automation and instrument, 25: 18-21.
5. Marvell Technology, February2007.88W8686 Datasheet Rev D.
6. Chen Haiyan and Ren Songyan, 2010. Research of Embedded Linux Root File System, Modern Electronics Technique, 4: 73-75.