*Communications and Network*, **2013, 5, 549-553**
http://dx.doi.org/10.4236/cn.2013.53B2099 Published Online September 2013 (http://www.scirp.org/journal/cn)

# Wireless Distributed Monitoring Terminal Used for On-Line Application

**Fenggui Wang, Lin Zhang, Zhongmin Wang, Yanbo Zhang**
Institute of Automation, Shandong Academy of Sciences, Jinan, China
Email: wfgranger@gmail.com

Received June 2013

## ABSTRACT

A wireless distributed monitoring terminal was designed with embedded Linux used on the basis of ARM board, and an application program was developed for data collection, processing and transmission. Data were collected from the surrounding environment and transmitted to a server via WLAN in the form of Extensible Markup Language (XML) stream, so the terminal had good flexibility to be compatible with different sensors, network devices and information systems from multiple manufacturers.

**Keywords:** On-Line Monitoring; Distributed Monitoring; Extensible Markup Language (XML) Stream; Wireless Terminal

## 1. Introduction

On-line applications are becoming more and more widespread along with the popularity of the Internet of Things in various fields [1], such as environmental monitoring, public safety, telemedicine, etc [2,3]. In these applications, some terminals are necessary to execute data acquisition, processing and transmission. There may be different sensors, network devices and information systems from multiple manufacturers even if in one application, so it is important that every terminal has good flexibility to be compatible with them.

ARM platform has been very popular in the area of mobile devices such as smart phones, personal players of audio and video, GPS systems and so forth, and can also compete with desktop computers based on i386 architecture microprocessors in some case [4]. ARM microcontrollers are convenient and efficient for designing and developing especially with embedded operating system used, so more and more terminals begin to adopt designs based on them nowadays.

This paper deals with design of a wireless distributed monitoring terminal with embedded Linux used on the basis of ARM board, and development of an application program for data collection, processing and transmission. It collects data from the surrounding environment via RS485 or A/D convertor, carries out data analysis and processing, and transmits them to a server via Wireless Local Area Network (WLAN). The server stores these data to an oracle database and presents them through web pages. It is needed to be specially noted that these data are encapsulated into Extensible Markup Language (XML) stream by terminal before transmitting and then parsed to obtain the semantics by server after receiving [5], so as to enhance the flexibility of whole system.

## 2. Hardware Design

The hardware of the wireless distributed monitoring terminal includes ARM9 board based on S3C2440A and Wi-Fi module to send and receive data wirelessly between it and the server. Also, it connects temperature/humidity sensor and illumination sensor as example for monitoring changes of the surrounding environment. The structure diagram of the terminal is shown in **Figure 1**.

### 2.1. ARM9 Board

The board is based on S3C2440A microcontroller with 512 MB RAM and 512 MB flash memory. SAMSUNG's S3C2440A is designed to provide hand-held devices and general applications with low-power, and high-performance microcontroller solution in small die size. To reduce total system cost, the S3C2440A includes the following components: separate 16 KB Instruction and 16 KB Data Cache, MMU to handle virtual memory management, LCD Controller (STN & TFT), NAND Flash Boot Loader, System Manager (chip select logic and SDRAM Controller), 3-ch UART, 4-ch DMA, 4-ch Timers with PWM, I/O Ports, RTC, 8-ch 10-bit ADC and Touch Screen Interface, Camera interface, IIC-BUS Interface,

 

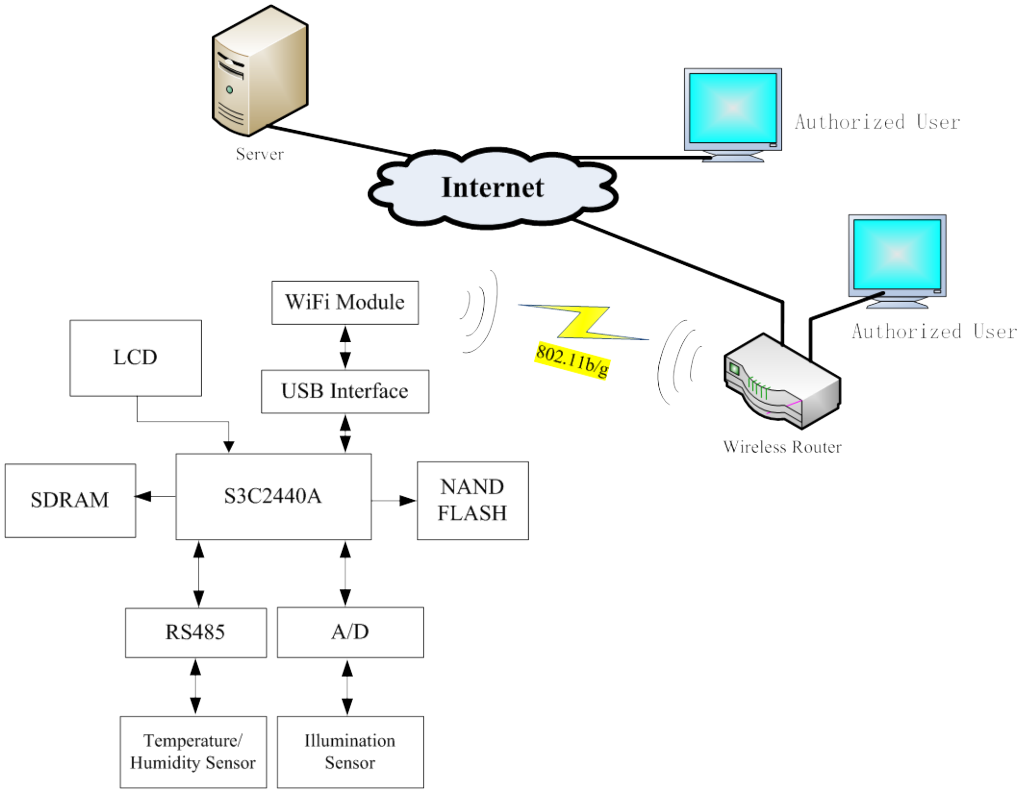


**Figure 1. Structure diagram of wireless distributed monitoring terminal.**

IIS-BUS Interface, USB Host, USB Device, SD Host & Multi-Media Card Interface, 2-ch SPI and PLL for clock generation [6]. The platform also offers a large number interface to support peripheral equipments, including USB host, USB device, LCD touch screen interface, RS232, RS485, 4 - 20 mA/0 - 5 V convertor and so on.

## 2.2. Wi-Fi Module

The Wi-Fi module adopted based on VIA Networking VT6656 is fully compliant with IEEE's security and 802.11 g standards [7]. Enabling data rates between wireless devices of up to 54 Mbps, it can deliver the bandwidth necessary for real-time streaming of high-definition digital multimedia content in home networks, and provide high-speed email, web and LAN access to multiple mobile users in corporate environments or public hotspots. Integrated WLAN into USB 2.0 with it, device also makes it more convenient to get on the net anywhere you are.

## 2.3. Sensors

The temperature/humidity sensor and illumination sensor are used to collect data from the surrounding environment in our system.

1) Temperature/humidity sensor

This sensor adopts the most advanced digital sensor technology, overcoming disadvantages of the traditional analog temperature sensor, such as instability, much deviation from accurate data, vulnerability to interference, and requiring regular calibration. It processes the function of collecting information of temperature and humidity, with calculating data stably and accurately. Its operating voltage can be as low as 12 V, accuracy of temperature ±0.7˚C, accuracy of humidity ±3% RH.

Connected to the monitoring terminal via RS485, an industrial level communication network with long-distance transmission, high speed and high stability compared with RS232, the sensor transmits acquired temperature/humidity data using Modbus standard. It is an application layer messaging protocol that is positioned at level 7 of the OSI model and widely used in industrial environment.

2) Illumination sensor

This sensor has a wide light measurement range of 0 ~ 200,000 Lux, with high precision linear amplifier circuit and high sensitive photographic detectors. Its accuracy is less than ±7%, and operating voltage is 12 - 24 V.

After data acquisition, the sensor transmits 4 - 20 mA current signal to the monitoring terminal, and then the 4 - 20 mA current signal is converted to 0 - 3.3 V voltage signal. Through an analog-digital converter, digital data are acquired and they can easily be processed and analyzed by S3C2440 to make them understood by human.

 

## 3. Software Design

The linux-2.6.28 distribution compiled for ARM9 platform is chosen as the operating system. The application program residing on the ARM platform is written in C language and compiled using the Linux GNU C compiler, which can be downloaded from the network free of charge.

### 3.1. Device Drivers

Device drivers take on a special role in Linux kernel. They are distinct "black boxes" that make a particular piece of hardware respond to a well-defined internal programming interface, and the details of how the device works are hidden completely by them [8]. User activities are performed by means of a set of standardized calls that are independent of the specific driver, mapping those calls to device-specific operations that act on real hardware is then the role of the device driver. This programming interface is such that drivers can be built separately from the rest of the kernel and "plugged in" at runtime when needed. This modularity makes Linux drivers easy to write, to the point that there are now hundreds of them available.

Linux device driver belongs to the kernel part, and there are two ways of compiling and loading it into Linux kernel. One way is to directly compile device driver into the kernel, along with Linux starts when loading. Another way is to compile device drivers as dynamic loading and deleting modules, using the command of "insmod" to load and "rmmod" to delete. This way controls the size of the kernel, while the other way increases redundancy of the kernel, so it is adopted to compile USB wireless network card driver, exit key driver, A/D driver, etc.

### 3.2. Work Flow

The monitoring terminal firstly initializes, including starting Linux system, loading USB wireless network card driver, exit key driver and A/D driver and setting serial port, and sends archived data, which is stored in a local file when the network is unconnected. Then real time data is frequently acquired from sensors, analyzed by classification, stored into local file, and sent in the form of XML stream. When sent successfully, the data will be deleted from the local file immediately. Due to being stored into a local file first, data will not be lost even if the terminal is powered down suddenly. Next, the terminal will check whether the time point set beforehand has arrived. If the time point is up, it will send archived data stored in the local file, otherwise it still conducts as before. Also, if the terminal detects the exit key has been pressed, it will exit normally including releasing memory closing devices, unloading drivers, etc. The work flow of the terminal is shown in **Figure 2**.

## 4. Testing

There is a wireless distributed monitoring terminal tested which collects data from the surrounding environment and transmits them to a server via WLAN. These data are stored to an oracle database and presented through web pages, as shown in **Figures 3** and **4**.

## 5. Conclusion

In this paper, an embedded Linux operation system combined with ARM9 platform is adopted to implement a

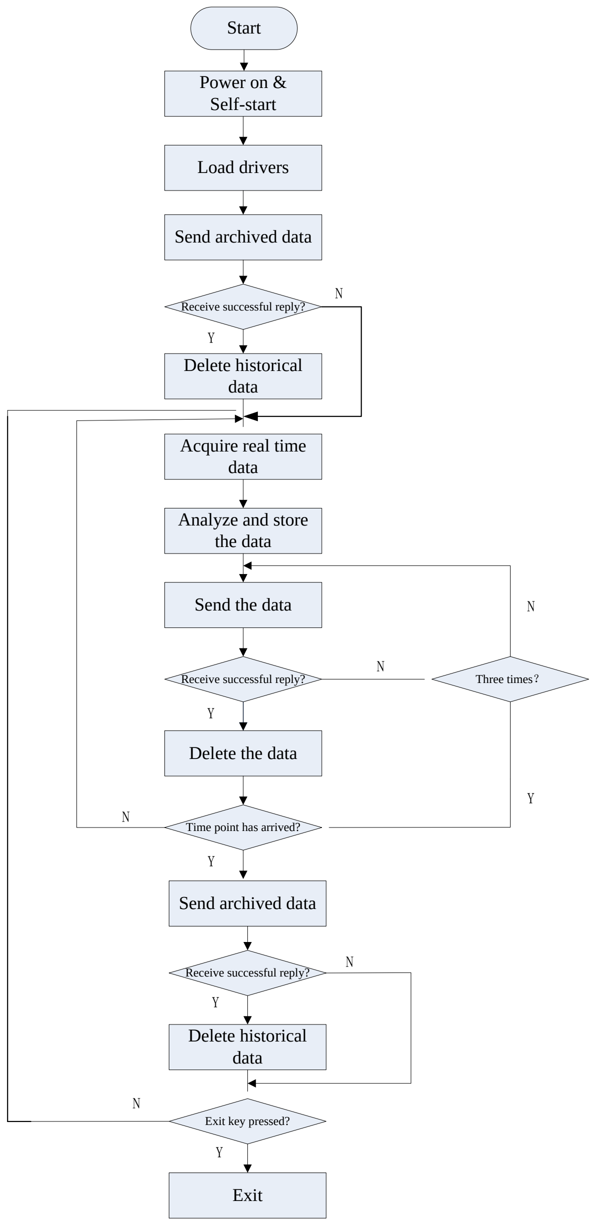


**Figure 2. Flow chart of wireless distributed monitoring terminal.**

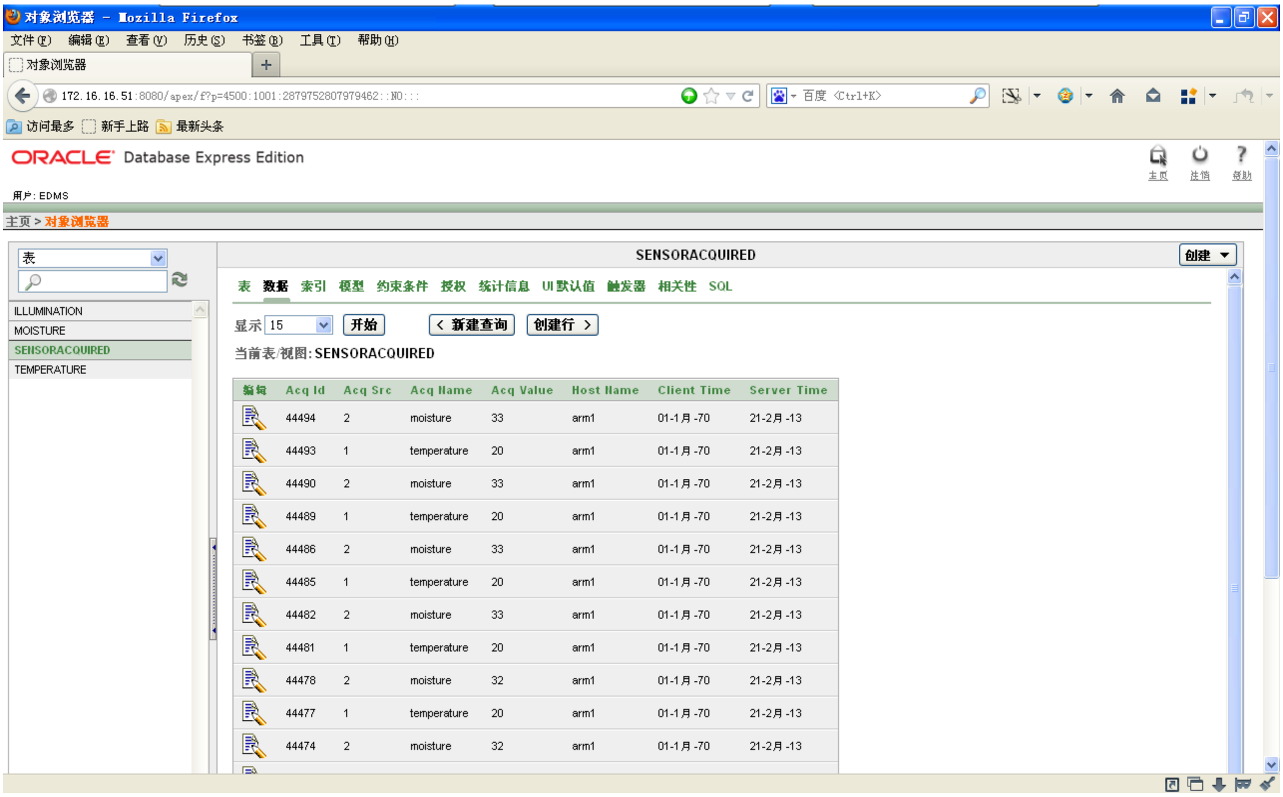


**Figure 3. Data stored to oracle database.**


**Figure 4. Data presented through web pages.**

wireless distributed monitoring terminal, and an application program is developed for data collection, processing and transmission. Data are sent in the form of XML stream, stored in oracle database and presented through web pages, so it's feasible to add any sensor to the terminal. Future possible works based on this terminal are adding more environment sensors and improving the XML streaming protocol.

## 6. Acknowledgements

The paper is supported by Youth Science and Technology Star Program of Jinan City, China (Grant No. 20100105) and Youth Science Funds of Shandong Academy of Sciences, China (Grant No. 20100106). The authors would like to thank all collaborators, especially who are in the UWB & THz Innovation Team, Shandong Academy of Sciences, China.

## REFERENCES

[1] Wikipedia Foundation, "Internet of Things," 2013. http://en.wikipedia.org/wiki/Internet_of_Things

[2] T. Qiao and L. Song, "The Design of Multi-Parameter Online Monitoring System of Water Quality Based on GPRS," *IEEE* 2010 *International Conference on Multimedia Technology* (*ICMT*), 2010, pp. 1-3.

[3] W. Fu and X. Meng, "Remote Monitoring System Research and Implementation Based on Wireless Communication," 5*th International Conference on Machine Vision* (*ICMV* 12). International Society for Optics and Photonics, 2013.

[4] Wikipedia Foundation, "ARM architecture," 2013. http://en.wikipedia.org/wiki/ARM_architecture.

[5] F. Wang, H. Yang and D. Tian, "IM-based Communication Mechanism for an On-line Monitoring System," *IEEE Proceedings of International Workshop on Intelligent Systems and Applications*, 23-24 May 2009, Wuhan, 2009, pp. 145-148.

[6] Samsung Electronics Co., Ltd., "S3C2440A 32-Bit RISC Microprocessor User's Manual (Revision 1.2)," 2004.

[7] VIA Technologies, Inc., "VIA Solomon VT6656-802.11a/b/g Wireless LAN Controller with Integrated Baseband Processor-USB 2.0 Bus Interface," 2013. http://www.via.com.tw/en/products/networking/wireless/vt6656.

[8] J. Corbet, A. Rubini and G. Kroah-Hartman, "Linux Device Drivers," 3rd Edition, O'Reilly Media, 2007.

 