Scientific Research

Research and Realization of Healthcare Monitoring System Based on ARM7 and uC/OS-II

LIU Chuan-ju, TANG Yu
1. Information College,ZhongKai University of Agriculture and Engineering, Guangzhou, China
1. e-mail liuchuanju@zhku.edu.cn, 2. e-mail tangyu_mycauc@163.com

Abstract: The research and realization of a multiple physiological parameter monitoring system based on ARM7 processor and uC/OS-II embedded operating system is proposed in this paper, which realizes real-time acquisition, display and transmit of the patient's physiological parameters such as blood pressure, pulse, ECG, blood oxygen saturation, temperature and so on. The monitoring system adopts the ARM7 as a core, which achieve real-time processing through multi-task scheduling. Monitor system comprises Ethernet communication interface, it can be conveniently access to the local hospital LAN and connects to the Internet to realize remote diagnosis and monitoring.

Keywords: ARM7; uC/OS-II; ethernet; monitoring system

1 Introduction

Patients play the most important role in medical system. For their treatment and rehabilitation, medical institutions placed on all their efforts to them.[1-3] In order to prevent and cure diseases and find diseases earlier, it is very important to real-time control some important health parameters such as blood pressure, pulse, ECG, blood oxygen saturation and body temperature. We developed a healthcare monitoring system based on ARM7 and uC/ OS-II, which includes a number of key functions, such as human physiological parameters detection, display and analysis, abnormal state alarm and network communication.[4-5] It can easily access to the Internet in the every corner of the world to achieve remote diagnosis and monitoring through real-time remote two-way communication between users and medical institutions.

2 System hardware design

The structure of hardware is shown in Figure 1. Six parameters module can get human ECG, noninvasive blood pressure, blood oxygen, pulse rate, respiration and body temperature parameters through lead port, light finger and cuff. Embedded processor, which adopts ARM7 as a core, connects to six parameters module through serial port. ARM7 central processor catches and processes the data from six parameters module through multi-task scheduling. LCD module can display all kinds of graphical and numerical signals. This system can also be controlled under an external keyboard to store and send data to the network, and signal detection alarm lines will make a warn when the detected data beyond alarm lines.

The controlling and processing core of the system is composed by the S3C44b0X from SAMSUNG based on ARM7TDMI core. S3C44b0X is a 16/32 bit high-performance RISC microprocessor, whose main frequency speeds up to 66 MHz. It incorporates 8 KB cache, LCD controller, UART 2-way, 4-channel DMA, five road with PWM timer, RTC controller, 8-1 O-bit ADC, IIC bus interface, IIS bus interface, Synchronous SIO interfaces and IO mouth including eight external interrupt source inputs. There are eight memory bank and each bank are addressable 32 M bytes. Its four operating modes diminish system power consumption and high level of integration helps system miniaturization.

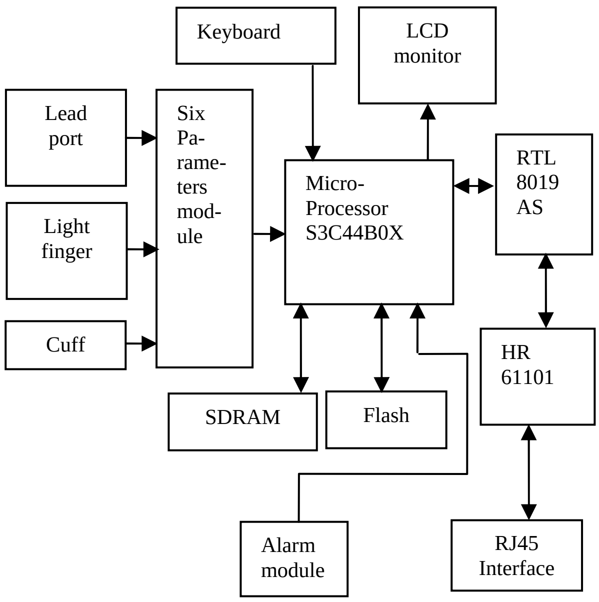

Figure 1. System hardware schematic structure

To achieve 10M Monitor Ethernet connectivity, Ethernet controller chip adopts RTL L8019AS. It supports full-duplex mode with double network bandwidth efficiency, compatibles to the NE2000, built-in 16 KB cache memory. 10M bandwidth of Ethernet controller is sufficient for the purpose of monitor's real-time data transmission rates.

Guangdong Science and Technology Plan Program (2006B21001010).

978-1-935068-01-3 © 2009 SciRes.

3 System software design

3.1 Embedded operating system u C/OS-II and system application software organization

Embedded operating system is the basic of the software system which is responsible for the allocation of system resources like multi-task scheduling. This system adopts a real-time embedded operating system uC/OS-II that includes a totally preemptive kernel. It can manage 56 missions at most and distribute each mission with a different priority. It includes a streamlined core which is high-performance, stable and outward-code.

Firstly, real-time operating system uC/OS-II is transplanted to S3C44B0X embedded microprocessor. According to their different functions, system software is divided into several parallel tasks under the control of uC/OS-II. Preemptive operating system schedules tasks depending on priority level. According to priority descending order, the system tasks are arranged as keyboard scan task, LCD display task, serial configuration task, monitoring mission task, monitoring data processing task, HTTP server task, CGI task and warning task. The judgment of the priority of the task depends on the real-time requirement of the system. Tasks which are more important to the system safe and more string enter requirement of real-time are set at high priority. In the operation of the system, the communication between tasks adopts message mailboxes mechanism provided by the operating system kernel. The system software process flow chart is shown in Figure 2.

Target_Init() function initializes processor hardware system, including general-purpose I/O port, serial port, interrupt and abnormal controller, task scheduling related interrupt vector and so on. UC/OS-II is initialized in the function OSInit(). All new missions are ready for the state after it was created. After the core is commenced, keyboard scanning task is implemented to catch the user input, and then it is translated into information packets and distributed through message mailboxes mechanism to other tasks to trigger other duties.

Monitoring data processing task receives and processes monitoring data captured by the six parameters module which is transferred serially, associates it with the current time and temporarily stores it in Flash memory for user access. HTTP server supervises port 80 to provide remote access for the Internet remote clients. CGI manages the interaction between the remote clients and servers. In accordance with the request of users, it dynamically generates script and provides the data needed by the clients. Serial port configuration and monitoring mission is used to facilitate the monitor configuration and debugging. Through the serial port, engineering personnel can send order-debug software and users can monitor configuration of the IP address, subnet mask, etc..

3.2 LwIP TCP/IP protocol stack and RTL 8019 AS driver

LwIP is an open-source TCP/IP protocol stack used for the embedded system, which is developed by Switzerland Computer Science College (Swedish Institute of Computer Science), Adam Dunkels.[6-7] LwIP is the meaning of Light Weight IP protocol, and can be operated without the operating system, although it has some difficulties when running in this manner. LwIP discussed in this system operates with the support of μ/OS- II. Different from other TCP / IP protocol stack, all the TCP / IP protocol stacks are in the same process (tcpip_thread) in LwIP that separate protocol stack from operating system kernel. But the application program can work in two statements. As an individual process, it can communicate to tcpip_thread through e-mail and information provided by the operating system. If it presences in TCP/IP protocol, the communication to tcpip_thread is through internal callback function interface (Raw API). LwIP work process is shown in Figure 3.

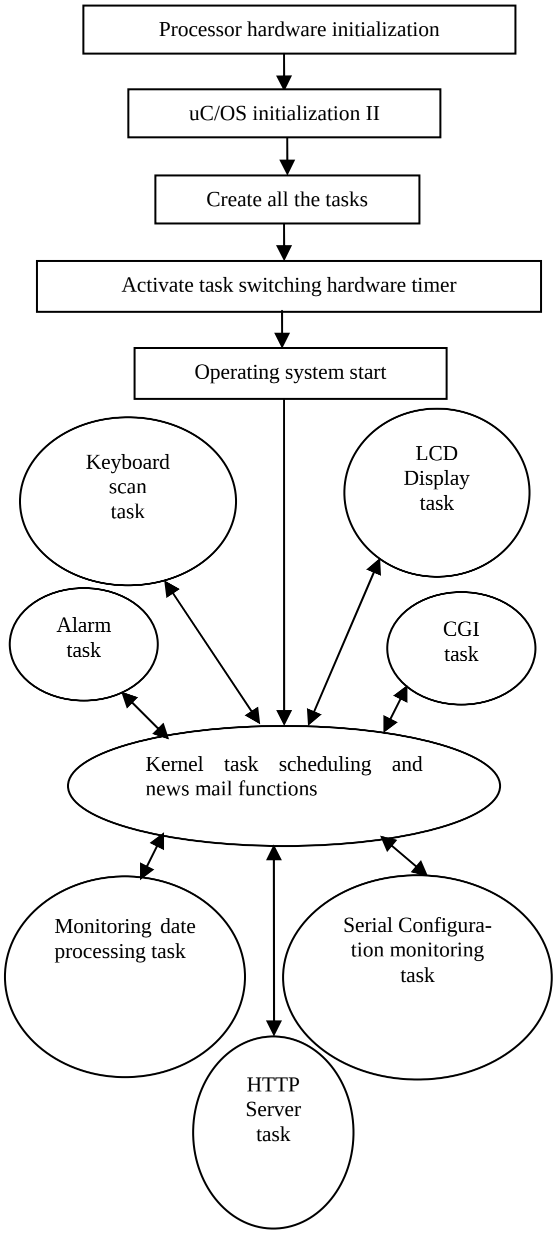

Figure 2. System software flow chart

Network driver mainly includes two parts. The first part is ehthernetif.c document which is a model provided by LwIP for network drive. It performs the function of distributing pbuf data structure for receive-data frame and firstly unpacking the data transmitted to the upper-class together with adding the first part of Ethernet to the sending data packets. The second part is a lower layer for the network hardware processing directly; it manages the set of MAC and PHY layer related control register and initialization of all kinds of data structure, as well as sending and receiving routines. As the first part's source code has been given in LwIP, it can be used without too many changes. The second part of the driver RTLL8019AS mainly comprises three parts: Mac_Init () function manages the initialization of LAN chip electrification, including setting NIC physical address and location and size of the transceiver buffer; Mac _Send () performs the function of data transmission; Mac_Rec () function is called by data packets receive interrupt handling function when reading the receiver buffer zone text and transmitting it to the upper function.

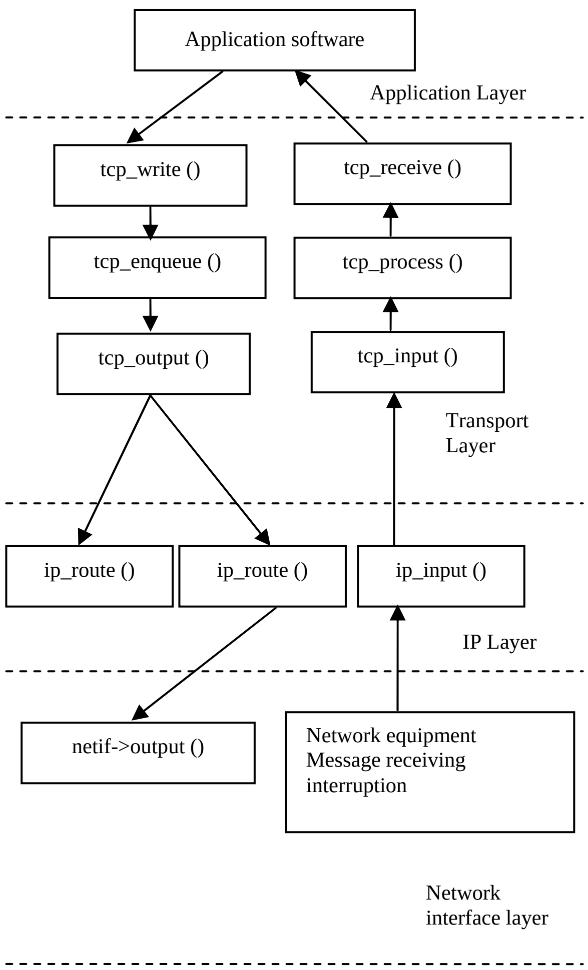

Figure 3. TCP/IP protocol stack function organization structure of LWIP

4 Conclusion

The smooth operation of monitor console depends on a good network environment including medical institutions LAN construction, a large medical databases construction, accesses to the Ethernet switches, wards and clinics room terminal querying and so on. Scene description of the system application is shown in Figure 4.

The success of the healthcare monitoring system can help healthcare workers get monitoring data not only from the LCD screen at the scene but also from the local medical units LAN or even any computer accessing to the Internet. Monitoring data can be real-time or saved in the medical unit database as historical data. It facilitates monitoring and diagnostic work and provides immediate protection for the wearer.[8-10]

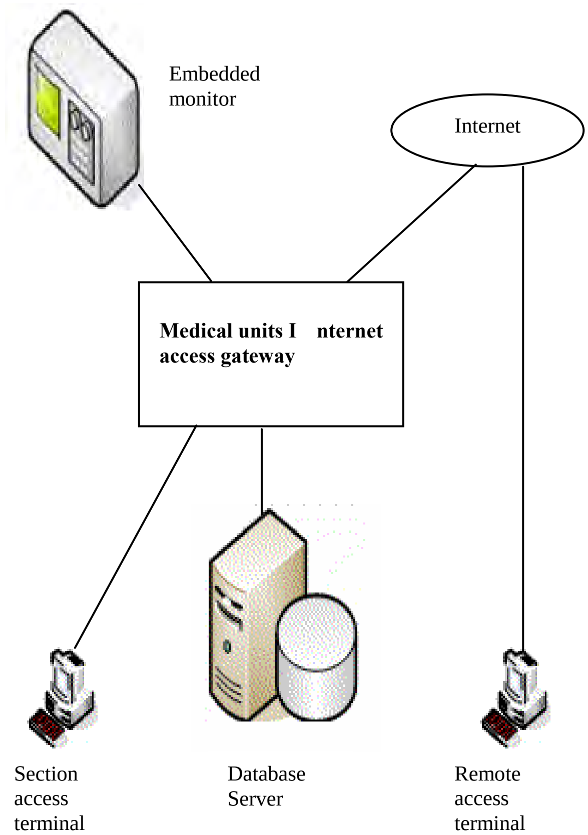

Figure 4. Monitor application scene graph

With the continuous development of informationization of informationization of society, Internet has penetrated into every corner of daily life; Ethernet interface design in health monitoring system makes remote monitoring and diagnosis anywhere and anytime possible.

5 Acknowledgements

This paper is supported by Guangdong Science and Technology Plan Program (2006B21001010), Guangdong and Hong Kong Focus on the Key Technology Breakthrough Projects (200649863107).

978-1-935068-01-3 © 2009 SciRes.

Reference

[1] Jessica A. Barber, Robert A. Resenheck, et al., “Monitoring the Dissemination of Peer Support in the VA Healthcare System”, Community Ment Health, Vol 44, pp. 433−441, 2008.

[2] Resnick S. G., Rosenheck R. A., “Recovery and positive psychology: Parallel themes and potential synergies”, Psychiatric Services, Vol 57, pp. 120−122, 2006.

[3] Stewart, M. J., “Professional interface with mutual aid self-help groups: A review”, Social Science & Medical, Vol 31, pp. 1143−1158, 1990.

[4] Dai yong-lu, tang xiao-ying, et al., “Portable multi-parameter health Monitor Design based on embedded systems”, Application of Electronic Technique, Vol.33, pp. 55−57, 2006.

[5] Wang Yun-xin, He Wei-xing, et al., “The Design of A New Kind of Portable ECG Monitor Based on ARM7”, Control & Automation, Vol.12, pp. 97−99, 2006.

[6] Dai Jin-min, Cao Xin-rong, Sun Xiao-gang, “Development of the Infrared Radiation Human Body Thermometer”, Infrared Technology, Vol. 24, pp. 52−54, 2002.

[7] Bao Xu-he, “Design of Portable Electronic Blood Pressure Monitor”, Modern Electronic Technique, Vol.30, pp. 7−10, 2007.

[8] Chinman M., “Service providers’ views of psychiatric mutual support groups”, Journal of Community Psychology”, Vol 30, pp. 349−366, 2002.

[9] Balas EA, Weingarten S, Garb CT, et al., “Improving preventive care by prompting physicians”, Arch Intern Med, Vol 133, pp. 761−769, 2000.

[10] Audet AM, Davis K, Schoenbaum SC, “Adoption of patient centered care practices by physicians: results from a national survey”, Arch Intern Med, Vol 166, pp. 754−759, 2006.