TELKOMNIKA, Vol. 11, No. 6, June 2013, pp. 3259 ~ 3264
e-ISSN: 2087-278X

The Transplantation of BOA Based on Linux3.0.1 and S3C6410

Jianmin Wang*, Haibo Wei
College of Communication and Electronics, Jiangxi Science & Technology Normal University
Corresponding author, e-mail: wjm92005@163.com, loverichardlove@yeah.net

Abstract

The web server is a tool of information queries and information dissemination based on hypertext. Special software didn't need to be installed by the users, and can access the embedded devices by accessing the browser through a general-purpose PC. In this paper, the embedded Web Server BOA transplantation, in the latest ARM11 architecture S3C6410 processor and version of the Linux3.0.1 operating system and gave an example of how to access the browser to control the development Board as the instance for LED lights.

Keywords: *ARM11, Linux3.0.1, common gateway interface, BOA server, S3C6410*

1. Introduction

With the development of embedded technology and the popularization of broadband network, remote control has been widely accepted. Embedded network control technology has been gradually developed on this condition.

We can install the appropriate equipment on the web server side, and the control of the corresponding embedded equipment can be achieved by local client through the ordinary browsers. And remote control of the objective system can be achieved by using the CGI gateway procedures to send command to monitoring site [1]. One of the hotspots of the embedded technology is how to make the embedded device access to the internet effectively, and to make the embedded equipment interact with the control center through web server .This paper will discuss how to achieve the design of web server based on the S3C6410 and Linux 3.0.1 embedded system, and further discuss its application.

2. The Introduction of Development Board Platform

ARM platform developed rapidly with the excellent performance, the extended ability of rich interface. At the same time, the application of the Linux operating system has a very strong flexibility because of the characteristics of open source. The platform that combines arm with Linux becomes the ideal development platform of many special applications.

In the embedded processor market, popular processors were, ARM, MIPS, PPC and other architectures. The documentation of ARM processor is rich, and ARM processor supported by most popular embedded operating system. In the choice of operating system, the most popular embedded operating system like Vxworks, Windows CE, UC/OS-II, Linux [2]. Compared to other commercial operating system, the Linux open source network operating system develops to a new stage with its open source, low cost, excellent network performance and stability. Therefore, the ARM processor and Linux 3.0.1 operating system been used in this design.

2.1. Hardware Development Platform

The hardware platform used in this article is the TE6410 development board. SAMSUNG's newest S3C6410 processor used in the platform. SAMSUNG's S3C6410 is designed to provide hand-held devices and general applications with low-power, and high-performance micro-controller solution in small die size. It offers outstanding features with tits CPU core, a 16/32-bit ARM920t RISC processor, and the internal resources are very rich,

Received January 21, 2013; Revised April 9, 2013; Accepted April 23, 2013

integrated with many powerful hardware accelerators ,like video processing, audio processing, two-dimensional graphics, display operation and scaling. Memory system with dual external memory port, DRAM and FLASH/ROM/DRAM port. By providing a complete set of common system peripherals, the S3C6410 minimizes overall system costs and eliminate the need to configure additional components. Compared with other processors in the same series, ARM11 enhances the processing speed of exception and interrupt, and the media processing extension makes MPEG4 encoding/decoding speed up [3].

2.2. Software Development Platform

Embedded software development uses the pattern of the host computer with development board. First, we write program in the host computer, then generate binary code that can be executed in the development board through the cross compiler, and then download it to the development Board to operate. The release version of REDHAT Enterprise Linux 5 has been selected as the host operating system because of its rich resources and superior performance.

The ARM-Linux-Cross Compiler environment should be established due to the application program is running on the ARM development Board. In this paper, the version of BOA source code is boa-0.94.13, the version of cross compiler tool chain package is cross-4.2.2.

2.3. BootLoader Design

The BootLoader is the codes that run firstly when power on. BootLoader program was run before the operating system kernel is running. From this program, we can initialize the hardware equipment and set up the image of memory space, and bring the environment of the system's software and hardware to an appropriate environment, to prepare a feat environment for the using of operating system kernel at last.

2.4. Transplantation of Embedded Linux Kernel

Linux kernel has five main functions: process scheduling, inter-process communication, memory management, network and file interface. Linux kernel is huge. Kernel transplantation is a complex task and also a very important process during the development of embedded system. Kernel transplantation mainly contains four parts: modify related codes according to its own hardware platform, kernel configuration, kernel compiling and kernel download.

2.5. File System Transplantation

File system is a part of embedded operating system, and its main mission is to manage the logical file, including the provision of file operations(create, modify, delete, etc.)interface, user-friendly operation of files and directories, and responsible for the security of documents.

2. Transplant of Embedded Web Server

3.1. Introduction to BOA Server

The current mainstream web server mainly have APACHE, HTTPD and BOA.etc. BOA is a lightweight single task http server, has characters such as open source, high performances, and the space needed by operating is only about 140KB, which has the best support for CGI. Different from other traditional web servers, when a connection request comes in, dealing with multiple HTTP connection request through establishing of HTTP request lists , at the same time it will create a new process for the CGI program, which can maximally save system resources. It is of importance for embedded system [4]. And BOA also has some functions such as generating directory automatically and extracting files.

The biggest feature of Boa is high speed, it is different with the traditional Web server, and it will not open process for every connection only, not by copying their own process to deal with many links. Through the establishment of a list of HTTP requests to handle multi-HTTP connection requests. Boa treats all the live HTTP connections internally, only open new process for every CGI connection, so this can save the system resources effectively. So, Boa has high degree of HTTP request processing speed and efficiency, with very good real-time, with a high application value in embedded system.To sum up, BOA has a very high speed and efficiency of HTTP request processing which has the high application value in the embedded system.

3.2. Transplant of BOA in the TE6410 Development Board

The transplant of BOA mainly involves the following aspects:

Open the website www.boa.org, download the source code of boa-0.94.13.tar.gz, and decompress under the virtual machine [5].

Open the file "src/defines.h", modify the define SERVER_ROOT "etc/boa". So boa in the development Board will start to read "boa.conf" document under the direction "etc/boa".

Carry out "./configure",and modify the file " Makefile". Then we will change in proper order "cc = GCC cpp = GCC-E" to "cc = arm-linux-gcc cpp = arm-linux-gcc-E".

Then execute "make". Generate BOA executable program in the current directory. Then perform "arm-Linux-strip boa" command that can reduce the debugging information.

Modified the file "boa.conf", and configure related directory, and configure the "errorlog " error log and the "accesslog" access log separately.The corresponding configuration is as follows:

Port 80 # set web port
User 0 # open the restrictions to user
User 0 # open the restrictions to user
The directory that story html file is: Documentroot /usr/local/boa
Cgi script directory is: Scriptalias /cgi-bin/ /usr/lib/cgi-bin
The directory that srory mime.types is :Mime.types /etc/mime.types
Comment out about ninety-fourth lines of boa.conf is :Servername www.your.org.here

Create the corresponding directories under the root file system, and then copy the relevant documents to the corresponding directories [6]:

Copy the boa executable files that built above to the "/etc/boa" directory ,and modify permissions.

Copy the "boa.conf" file to the" /etc/boa" directory.

Copy the webpage test file " index.html" and "leds.html" to the "/usr/local/boa" directory.

Copy the file "/etc/mime.types" in the virtual machine to the directory "/etc" in the development board.

Copy the file "cgi" to the directory "/usr/lib/cgi-bin".

Establish "/var/log/boa/error_log",and modify the relevant permission.

Establish "/var/log/boa/access_log",and modify the relevant permission.

Then we can set the network environment of the development board, in order to access by host computer conveniently.

Connecting the host computer with the development board through the router by using the network cable. The host computer's IP is 172.16.48.120, the development board's IP would be changed to the same network segment with the host computer. With related testing , the host computer can access to the development board.

3. Achieve Technology of the Web Page CGI

BOA is ported to the TE6410 development board that runs the Linux operating system, and built embedded web server to provide with network access mode based on HTTP protocol for users. The appropriate CGI procedures can be write to achieve the transmission of the video data and the processing of control instructions, because of BOA supports the dynamic webpage based on CGI [7]. This article can achieve the control to the LED lamp on the development board through the webpage, through the Linux pipe mechanism and a simple CGI program.

4.1. The Introduction to CGI

CGI is short for common gateway interface. It is the standard interface of interacting the external application of extended application with WWW server. Its main function is to pass the information from the client to web server to start the specified program to accomplish specific tasks in the WWW environment.

Through the interface, the CGI program can be used in the web server, adding dynamic content. The browser will send an HTTP request of an executable application program, not only static HTML files. The server runs the specified application that can read the information associated with the request , and can get the numerical from the request [8].

CGI application program transmits the corresponding implementation result to the client browser after completing the relevant operations and the operations to the bottom equipment to let the users see the results of the implementation of operation intuitively.

The basic workflow of CGI as shown in Figure 1:

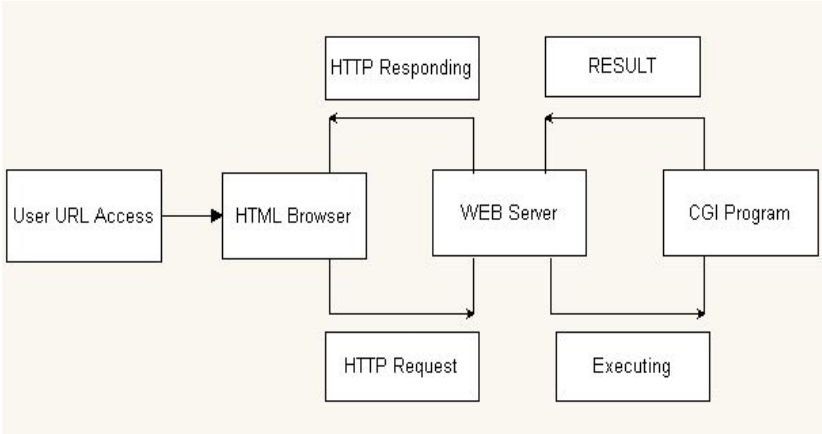

Figure 1. The Basic Workflow of CGI

4.2. Compile to Generate CGI Executable Program

It can obtain the procedure that can be ran in the development board by using the cross compiler to compile the CGI under the Linux console.

4. Application Examples

In this paper, the example of controlling LED blinking pattern through the webpage be given. In fact, through the Linux system inter-process communication mechanism to realize the pipeline, wherein LED is a shared resource, by running a background process to create a named pipe led-control in the directory “/tmp”, and has been detected the pipeline input data, to change led display mode according to different parameters [9].

“leds.cgi” is a gateway program that under the “/usr/lib/cgi-bin” in the webpage server file directory, it is called as an implementation action by webpage leds.html.it receives the character instruction that send from the browser, and these instructions are copied into the actual numbers, and then the instructions delivery to the “/tmp/led-control” pipeline by calling the echo command so as to realize the control of LED.

5.1. Testing for the BOA Operation

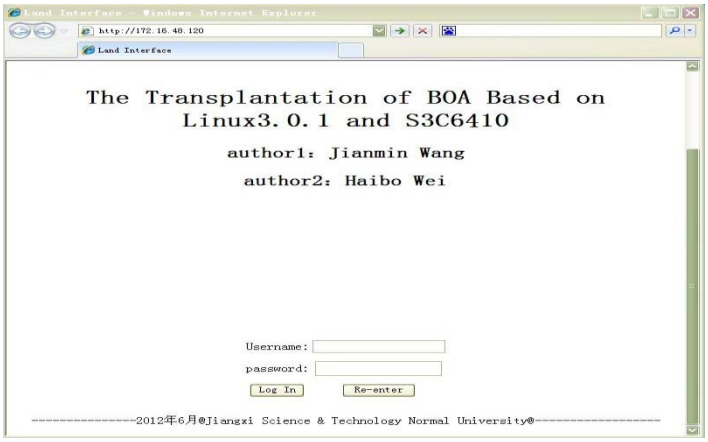

Figure 2. The Login Interface in the Development Board Web Server

We can test whether the BOA server can be ran successfully after transplantation. Executing the command to run the BOA in the development board directories “/etc/boa”.

Open an IE browser and type in the URL:http://172.16.48.120, that can access the dynamic webpage in the development board through the browser after operating the boa [10]. As shown in Figure 2.

The LED lamp control interface can login into by entering the user name and password. As shown in Figure 3.

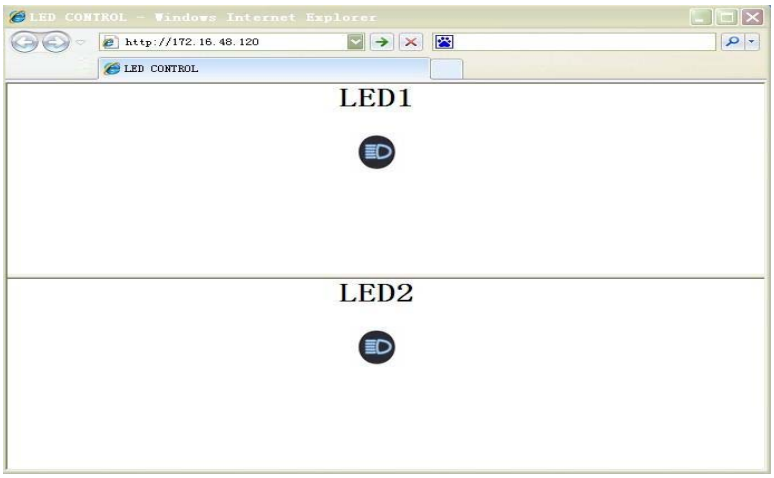

Figure 3. The LED Login Interface

We can see the change of the LED bright lights to eliminate flicker in the development after undertaking related operations. Effect diagram as shown in Figure 4:

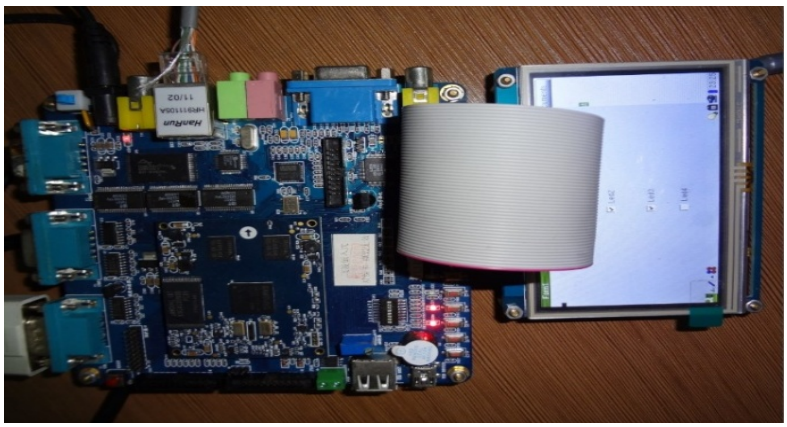

Figure 4. LED Lamp in the Development Board Practical Effect Diagram

5.2. The Analysis to the Test Result

After the actual test, the system is equipped with a ARM11 processor that has more advantages than the ARM9 in test, both in the data transmission and in the data acquisition process, showed better performance than the ARM9,and to achieve the desired results [11]. The processor will show more obvious advantages if it is used in video surveillance, speech processing and other fields.

5. Conclusion

In the basis of successful of transplantation of the server ,using the embedded web server in the area of network monitoring, which can greatly save system resources and improve the system efficiency, and give full play to the advantages of network. The web server can be widely applied to the industrial equipment, automation of agriculture, home medical equipment, information appliances, intelligent community, remote monitoring alarm system, the embedded GSM short message platform, the maintenance of communication base station. Embedded web server has the widespread application prospects.

Acknowledgements

First, I would like to extend my thanks to my tutor, Mr. Xie, who gave me numerous encouragements and help. Second, I also want to thank my beloved girlfriend Miss Zhou. Without their generous advices and supports, this paper cannot be accomplished. Thank you very much.

Supported by the Graduate Innovative Special Foundation of Jiangxi Science & Technology Normal University

References

[1] Karim Yaghmour. Building Embedded Linux Systems. O'Reilly. 2003: 206-238.
[2] Du Chunlei. *ARM Architecture and Programming*. Beijing: Tsinghua University Press. 2003: 173-175.
[3] Wei Dongshan. Embedded Linux Application Development Complete Handbook. Beijing: People's Post and Telecommunications Publishing House. 2008: 368-377
[4] C Andres, N Sloss, Dominic Symes. ARM System Developer's Guide. *Morgan Kaufmann*. 2004: 328-341.
[5] Huang Guangqi. CGI Programming Guide. Beijing: Publishing House of Electronics Industry. 1999: 201-203.
[6] Szymensky AW. Embedded Internet Technology in Process Control Device. *IEEE*. 2000.
[7] Ding Tianran, Wang Le. Based On Embedded Linux Web Remote Control System Design and Implementation. *Journal of Communication University of China*. 2009; 9(5): 135-137.
[8] Song Wen, Zhong Xiangqu. *Web Programming Guide*. Beijing: Tsinghua University Press. 1999: 411-423.
[9] Nasution H, Jamaluddin H, Syeriff JM. Energy Analysis for Air Conditioning System Using Fuzzy Logic Controller. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2011; 9(1): 145-150.
[10] Ziegler JG, Nichols NB. Optimum Settings for Automatic Controllers. *Journal of Dynamic Systems, Measurement, and Control*. 1993; 115 (2B): 210-215.
[11] Jiejia LI, Dai Yan LIU, Rui QU. The Strategy of Construction Equipment Energy-Saving Control. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2012; 10(4): 706-712.