

Design and Implementation of Embedded LINUX System Development Environment based on S3C2410 Chip

Xiufang Zhang^{1, a}

¹ Xi'an Eurasia University, Xi'an, Shaanxi, 710065

^a email

Keywords: S3C2410 Chip; Embedded; LINUX System Development

Abstract. With the continuous development of China's science and technology, improve people's living standards in the field of electronic information have been developed, especially for the development of S3C2410 chip embedded LINUX system, but also get a lot of attention of enterprises, the following analysis of the development environment process design, to achieve the above, hoping to promote the development of this area, to improve the level of technology in this area.

Introduction

It is no stranger to computers and networks, and their lives are closely linked, with the combination of network technology and computer technology, it is entered into the P C era. In the research and development of embedded systems become the focus of development. Here's to the embedded linux operating system, and Samsung's S3C2410 embedded processor chip, for example, detailed analysis of its design, lay the foundation for future work.

Related Part of Embedded Systems

In a computer system, the host is the main hardware platform, in addition, also constituted by a number of peripheral devices, the computer functions to be accomplished is very strong, so the processor computing speed must be fast, but is separate from the processor and other parts Designed. The embedded processor is the core component of embedded systems, the use of paper

S3C2410 embedded processor chip, which not only low energy consumption, and overall performance, low price, low-cost, low-power embedded systems is very good, in addition, there is an embedded software systems.

Design and Implementation of Analysis S3C2410 Chip Embedded LINUX System Development Environment

For embedded systems, their systems have a cross-development environment, when established will be used to host / target mode. To facilitate understanding, the following special drawings, reference analysis. For the host, refers to a general-purpose computer system, daily use of PC machine, said here the target development board, embedded system software last program run hardware environment, the host with Ethernet interface, or a string line port [1], it can be done and the target normal communication. For embedded development, it requires the host has a good development environment, so it can perform normal editing and debugging simulation for design work on the software. On this basis, after completion of compiled code, use the Ethernet interface, or a serial port can be decoded are transferred to the target machine, then use after cross debugger, the content analysis and debugging, and finally in a particular environment next, to complete the commissioning program run. CPU used in a computer system [2], are normally developed by Intel Corporation x86 series CPU, but the software needs of embedded operating environment is different, it uses chips are usually PowerPC, DragonBall, MIPS, ARM series, is a micro-processor. In operation, in order for it can be embedded in the hardware environment compiled execution. You should first establish a cross-development environment, and then use cross-development

environment, its effective links and cross-compilation.

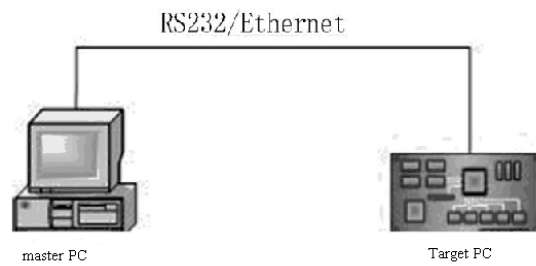


Figure 1: Embedded Linux cross-development environment

Take the arm-linux-gcc example for analysis, which is a source code that can be developed in linux operating system, and then can be compiled into binary code, such code can run on ARM architecture, but also belong to a species cross-compiler. Under normal circumstances all embedded microprocessor, its corresponding embedded system, not a single corresponding to a cross-compiler. In use, you can use the familiar cross-compiler. The computer system will be the preparation of their good [3], then this embedded software after cross-linked to form a new file, which is a binary file, you will be able to run on embedded hardware environment.

Among the computer system, the boot program mainly involves two aspects, first, a resource allocation procedure, which solidified in the BIOS, and on the other hand, the extent of which is a guide, in which different boot drive, the embedded in terms of systematic and scientific personnel in order to simplify the hardware architecture, the CPU is embedded not only in curing initiator hardware platforms, many of them embedded hardware platform, do not provide start the program, and therefore its development process, first programming embedded hardware platform boot program. So this level is also known as BootLoader [4]. The section of the program is actually in front of a computer system, run the short main program, run this small program, and its role is very large, the memory space map can be established, the hardware device may be initialized, and then the system hardware and software environment to a proper state, to facilitate future system kernel debugging, it is ready to run needed the right environment. For now, Bootloader used by the program are the following, for example Redhat company developed Redboot program, developed from PPCBoot the U-Boot, and Compaq developed the Bootldr etc., not one will be described. For one of the boot program, the South Korean company developed a Mizi Vivi, which Bootloader is a program that can support S3C2410 processor. In this study, we analyze the S3C2410 processor, so by supporting the boot program uses Vivi. In the application process, we need to Mizi company VIVI guide members download the source code, and then modify the Makefile, on this basis, the operator should also establish a board support package, the rational allocation of development board. After completion of the above-mentioned aspects of the operation of head.s file effectively modified, but it is noteworthy that we must distinguish Nor Falsh and Nand Flash [5], then VIVI recompile the source code allowed to meet the binaries demand, the final step is to carry out programming.

For Linux kernel porting, is to determine the specific embedded target platforms, it must effectively streamline Linux, and then mounted to the target platform, on this basis, should also be able to ensure its proper operation. The basic contents include the following items, following specific elaborate. First step, get a version of the Linux kernel source. The second step, combined with the objectives set before the platform, effectively streamlining the source, the main content of the work and structure of the system is to be modified, and then add the drives designed to meet a target platform of the new operating system. After the third step operation, cross-compile target platform for this system, so that it can generate a kernel image file. The fourth step, the image files obtained as described above, in strict accordance with the requirements of programming, can be installed after the completion of the target platform, and then use. Before programming the kernel, the operating system must be streamlined to do, if there are unwanted kernel may well be removed, to avoid system space is occupied, improve the efficiency of system operation and improve system storage space. After completing the streamlining of operations, designers also need to be recompiled, this will work out a new file, the file system used to meet our compressed kernel image file zImage.

In the case of the above steps are operating without problems, you can transplant the Linux kernel programming, and ensure the quality of programming.

For Linux, the use of the file system organization system in which documents and equipment to provide a uniform interface to the device, and the user program. In addition, it also supports J F F S 2 .RAMDISK, C R A M F S and many other file systems, when using the system, which can YAFFS2 root file system to read and write. For the root file system, which includes a lot of content, for example, the Linux operating system, and it needs at startup directory, key files. Here's an example, if you want to start Linux, will certainly use in the init file to a directory, if the operating system to mount a different partition, then the process is running, you must use the etc / fstab directory mount the file, all of the Linux operating system involved, during the startup process, used to file or directory, the technician can call it Linux system's root file system. For YAFFS2 file system, when the system is running, which can support NAND FLASH, and JFFS there are many similarities, and its role is to design embedded file system. But the Linux operating system and JFFS compared to the reduced part of the function, the operator can be found in actual use, the operating speed of the system is faster and uses less memory.

Conclusion

Through the above design and realization of S3C2410 chip embedded LINUX system development environment analysis, analysis of some of its components, to understand the function of the boot program, in addition, also on the embedded linux operating system YAFFS2 file system, Linux kernel porting method carried out a detailed analysis in this regard so that the operator has an overall understanding, reducing the threshold arm embedded development application, promote the development of this technology in the future development, with the related art enhance national attention in this regard, it must be a better development.

References

- [1] Zhu Xiaojun, Zhang Zhibin, Liu Huipeng. realization S3C2410 chip embedded linux system development environment based design [J]. Automation and Instrumentation, 2012 (01): 112-113.
- [2] Zangjin Mei, Tao Xun, Hu Yunling. Realization based S3C2410 embedded Linux development environment [J]. Chinese e-commerce, 2012 (22): 49-49.
- [3] Zhu Xiaojun, Zhang Zhibin, Qu Chaocheng. Based design of embedded automotive GPS positioning system S3C2410 chip [J]. Automation and Instrumentation, 2012 (02): 64-65.
- [4] Guo Guangming. Design S3C2410A embedded remote video monitoring system [J]. Science and Technology Management Research, 2010, 30 (14): 233-235.
- [5] Xu Min. design S3C2410 embedded video monitoring system [J]. Computer Knowledge and Technology, 2014 (8): 1802-1803.