CSCanada

Advances in Natural Science
Vol. 8, No. 1, 2015, pp. 33-40
DOI: 10.3968/6226

ISSN 1715-7862 [PRINT]
ISSN 1715-7870 [ONLINE]
www.cscanada.net
www.cscanada.org

Research on Tailoring Technology of Array CCD Aerial Camera Linux System

CAO Weiguo[a],*; YANG Haini[a]; DUAN Yuhan[a]; WEN Peipei[a]

[a]College of Opto-Electronic Engineering, Changchun University of Science and Technology, Changchun, China.
*Corresponding author.

Received 20 December 2014; accepted 5 March 2015
Published online 26 March 2015

Abstract
In view of the actual demand of operating system of the air plane array CCD camera, and combining with the hardware resources of the PC104 bus structure, Linux system adopted in CCD camera is cut practically, which based on the tailoring method adopting the combination of coarse-grained and fine-grained to enhance the Linux kernel preemption, improve the scheduling strategy of Linux kernel scheduler, to build a embedded system with the strong implementation capacity. The system startup and task of the response performance test in different environment shows that the cut systems is stable, reliable, and can achieve the startup time less than 5s, the performance of the task response time less than 20 millisecond.

Key words: Aviation camera; Operating system; Tailoring; Linux

Cao, W. G., Yang, H. N., Duan, Y. H., & Wen, P. P. (2015). Research on Tailoring Technology of Array CCD Aerial Camera Linux System. *Advances in Natural Science*, *8*(1), 33-40. Available from: http://www.cscanada.net/index.php/ans/article/view/6226
DOI: http://dx.doi.org/10.3968/6226

INTRODUCTION
Array CCD aerial camera is mainly used for certain type of military aircraft In the high altitude, the task is to obtain the high resolution visible light image information of target in real time from high altitude . and the control system is the control core of aerial camera, its response speed directly affects the performance of the aerial camera. Leading to the result of the influence aviation reconnaissance task completion. In order to meet the real-time design requirements for the aerial camera, usually choose Linux system as the main control system. Linux system is mature, stable, open source code and can be cut out, and there is no black box technique, While the existing Linux is a general-purpose operating system, Although the existing adopts many advanced technologies to improve the operation and the response speed of the system .but it is not essentially a real-time operating system, there are still many problems in Applied in the aerial camera, it is not meet the requirements of aerial camera embedded environment, For example, close the interrupt problem, process scheduling problem and the clock problems, these shortcomings delay of high priority interrupt request, prevents the high priority process real-time preemptive CPU and reduce the real-time periodic tasks implementation efficiency. In this paper, in view of the existing problems, cut and transformation of practical work, thereby reducing the blindness and accelerate the progress of the embedded Linux system transplantation, and its application in high altitude area array CCD aerial camera control system. With independent intellectual property rights for the development of China’s aviation embedded processor and embedded operating system and embedded high-end products laid the foundation, to improve China’s weapon design level and degree of automation, has important strategic significance and broad application prospects.

1. THE TAILORING TECHNOLOGY OF LINUX KERNEL
According to the current research status of Linux tailoring method, the existing Linux tailoring technology is mainly divided into:

1.1 Since the Distribution of Tailoring Technology Based on Linux

In the initial design of the Linux kernel, The core of all the relevant features modular design, and all function modules is used a set of independent configuration options. According to the design requirements of different functions of the embedded system, the menuconfig kernel module is compiled with Kernel compilation directives make menuconfig in practical application. This approach is the use of Linux systems with self-compiled kernel tailoring tool to achieve a object-oriented kernel system, kernel system is design according to different embedded system required.

Considering the function of embedded system is relatively single and specifically, to realize its function by supporting the peripheral hardware is relatively simple, so as long as Linux system must be Understood the configuration options for all modules, you can choose specific work platform that hardware needed to drive module, the kernel occupies a relatively small scissors of system memory with the tailoring technology of kernel system. Thus tailoring kernel occupy much of system storage space is relatively small, not only it meets the required hardware configuration requirements, but also realizes the function of feasible operation. The tailoring method of particle size is bigger, also called as coarse grained tailoring.

1.2 The Tailoring Technology Based on Source Code of Linux System

In practice of tailoring the embedded systems, through the analysis of the detailed functions of the Linux kernel source code, At the same time the specific operation requirements in reference of the embedded system, the kernel code can be Modified e.g. Reservations run operation of the kernel code, removing irrelevant code or function, providing the practical use of the code, reducing the space occupied by the irrelevant code. Because of smaller particle size of such tailoring method, it is known as a fine-grained tailoring. Its advantages are tailored the relative content in detail, but the shortcoming is that the operation is very difficult.

1.3 The Tailoring Technology Based on Call Graph of Linux Kernel

The tailoring process based on call graph of Linux kernel is divided into the following five steps:

1.3.1 Establish the Application Call Graph

The program call graph of Linux system is designed to C=(V,R).

Design the program call graph with C=(V,R), Specifically shown in Equation (1):

$$R=f(V1,V2)|V1,V2\in V. \qquad (1)$$

In the formula(1),V is the set of all functions in the program;

R is a function call relation set.

According to the formula (1) , the application code can be established, the call Then according to formula (1) and program code , The actual application call graph can be established, library functions can be directly selected , at the same time irrelevant content can be removed.

1.3.2 The Establishment of a Shared Library Call Graph

Linux systems can be realized in a call to the library function, Therefore, the establishment of call shared library is an important component of the Linux system, all procedures systems involved of the basic function is stored in the shared library. However, because of its complex content, It takes a lot of storage space. So shared libraries need to tailor.

Usually, Linux system within the shared library is given in the form of a call graph, as shown in Figure 1, it can directly display the structure of library function calls, In applications, it can be directly added to the Linux, and provides multiple entry for the application. The purpose of the establishment of a shared library call graph is to understand the related data of repeated calls, has nothing to do with the database can be directly removed, realize the library shared call out.

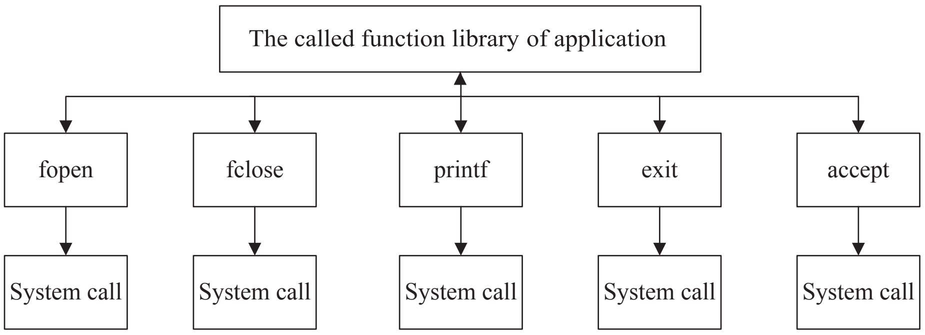


Figure 1
Schematic Diagram of the Shared Call Graph

1.3.3 The Kernel Call Graph Is Established

The method of the establish the kernel call graph and the library call graph is similar, when the system has a unusual request or interrupt signal, the kernel will interrupt the discriminant,According to the different level of interruption to the corresponding interrupt handling, at this point only set up the kernel call graph, can clear start of the specific conditions and time of the kernel function. According to the basic condition of kernel function to start, to remove the independent function in Linux tailoring.

1.3.4 The Establishment of Mixed Call Graph

By analyzing the establishment of the above three call graph theory, based on the purpose of weakening each call graph technical shortcomings, mix them up, combined with the aviation camera control system design, control system is constructed hybrid call graph image acquisition module, the mixed call graph is shown in Figure 2.

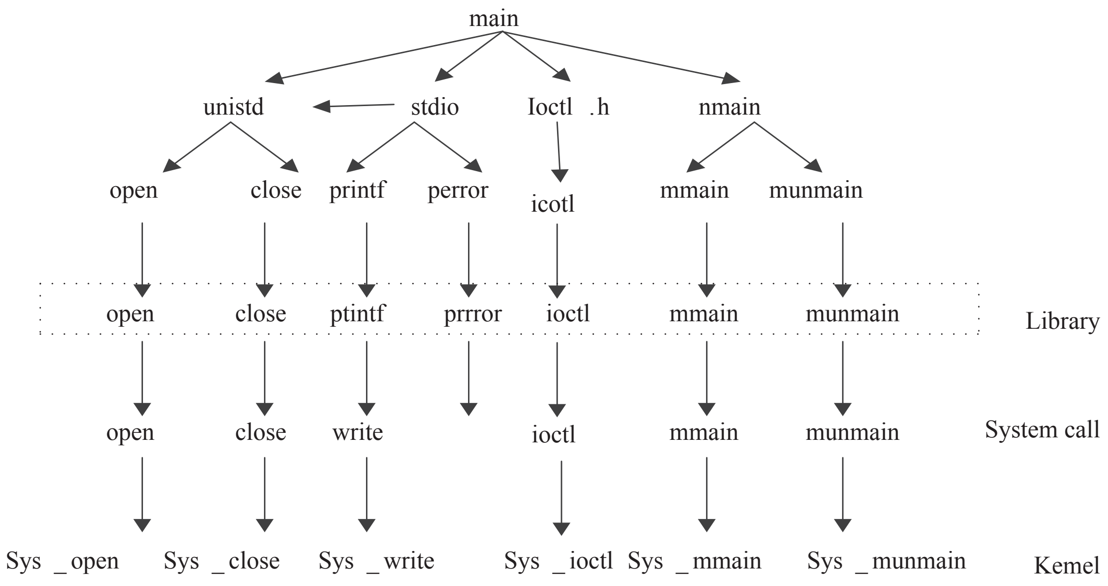


Figure 2
Mixed Call Graph Manipulation System Image Module

1.3.5 The Kernel Function and the Library Function Calls

Application of the design process, cutting the kernel function is required, at the same time, reference to the relationship between shared library function and kernel function, make actual calls to the relationship between application and the library function call, Can effectively remove irrelevant kernel function. See from figure 2 the established mixed call graph, library function that are written by the aerial camera control system of image acquisition module application calls including printf,mmap, munmap and ioctio, the kernel function includes sys_open, sys_close , sys_write, sys_ mmap, sys_ioctl, sys_muumap, and other independent library function and kernel function will be tailoring and removing.

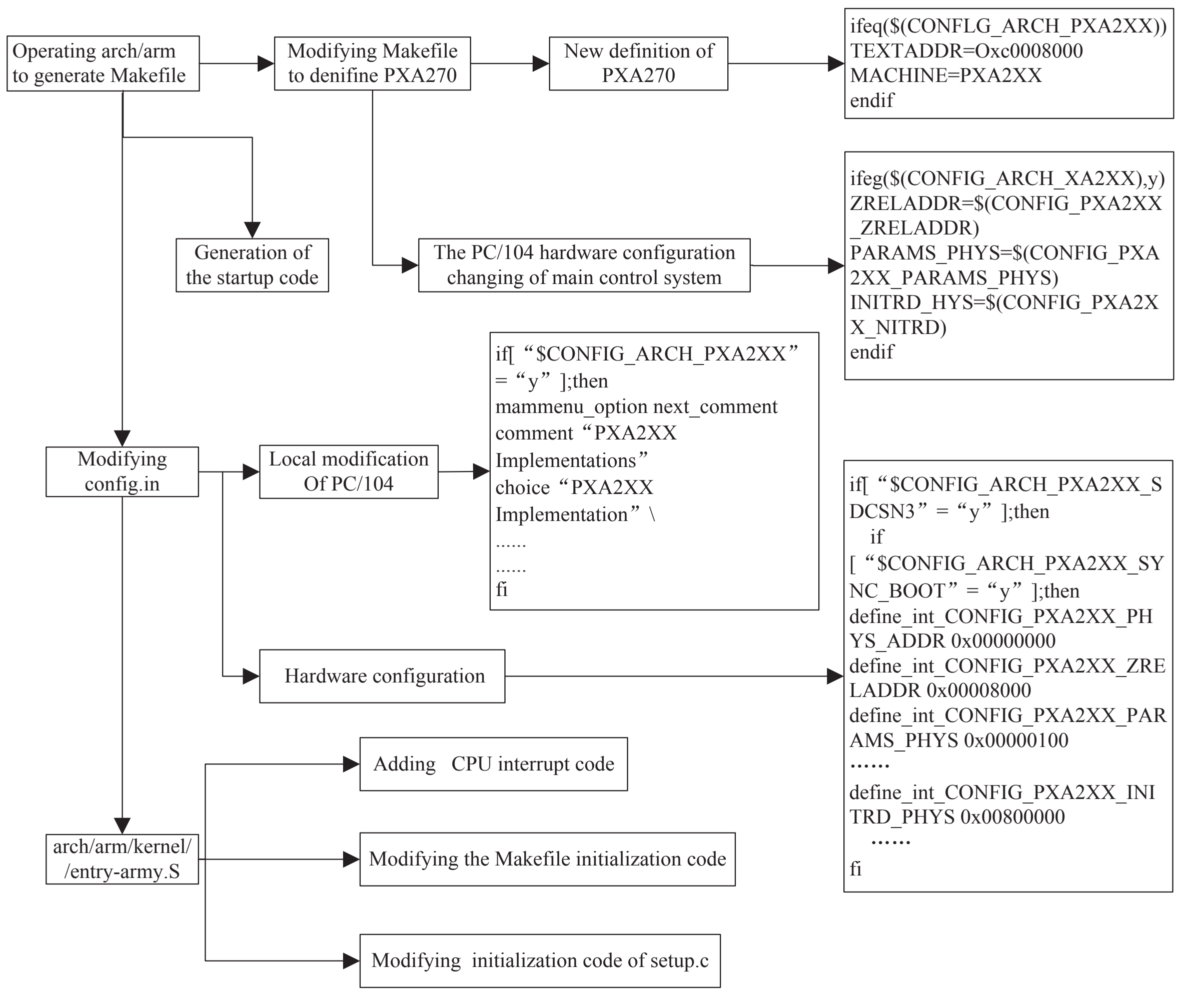
2. THE TAILORING OF LINUX KERNEL

For hardware resources based on PC104 bus is for the Boot Loader transplant, according to the application demand, will uses different tailoring method for tailoring after transplantation system.

2.1 The Change of Kernel Source Code

First of all, the startup program code modification is mainly related to the kernel code modification of entrance, the specific path is /arch/arm/kernel/head-army.s . The kernel is started, the kernel first through the entrance code to verify the consistency of the register parameters that defaulted by CPU and the Linux kernel code information by default. The register parameters CPU, the default value is the study of PC/104 control register values of each parameter pairing, which is composed of a starting procedure of Boot Loader kernel is given. By modifying the code correctly, entrance, can realize the Linux kernel and start the program, so that the normal start of the Linux kernel.

And because it involves the arch directory when Linux startup, so must the startup code for relevant documents must be modified, to meet the different requirements of embedded system startup, mainly related to the file of Make file, config.In and entry-army.s. The modify process of startup file code shown in Figure 3.


Operating arch/arm to generate Makefile
Modifying Makefile to denifine PXA270
New definition of PXA270
Generation of the startup code
The PC/104 hardware configuration changing of main control system
Modifying config.in
Local modification Of PC/104
Hardware configuration
Adding CPU interrupt code
arch/arm/kernel/ /entry-army.S
Modifying the Makefile initialization code
Modifying initialization code of setup.c

```
ifeq($(CONFLG_ARCH_PXA2XX))
TEXTADDR=Oxc0008000
MACHINE=PXA2XX
endif
```

```
ifeg($(CONFIG_ARCH_XA2XX),y)
ZRELADDR=$(CONFIG_PXA2XX
_ZRELADDR)
PARAMS_PHYS=$(CONFIG_PXA
2XX_PARAMS_PHYS)
INITRD_HYS=$(CONFIG_PXA2X
X_NITRD)
endif
```

```
if[ “$CONFIG_ARCH_PXA2XX”
= “y” ];then
mammenu_option next_comment
comment “PXA2XX
Implementations”
choice “PXA2XX
Implementation” \
......
......
fi
```

```
if[ “$CONFIG_ARCH_PXA2XX_S
DCSN3” = “y” ];then
   if
[ “$CONFIG_ARCH_PXA2XX_SY
NC_BOOT” = “y” ];then
define_int_CONFIG_PXA2XX_PH
YS_ADDR 0x00000000
define_int_CONFIG_PXA2XX_ZRE
LADDR 0x00008000
define_int_CONFIG_PXA2XX_PAR
AMS_PHYS 0x00000100
……
define_int_CONFIG_PXA2XX_INI
TRD_PHYS 0x00800000
   ……
fi
```

Figure 3
The Flow Chart of Kernel Source Code to Modify

2.2 Coarse Grain Tailoring the Kernel Configuration

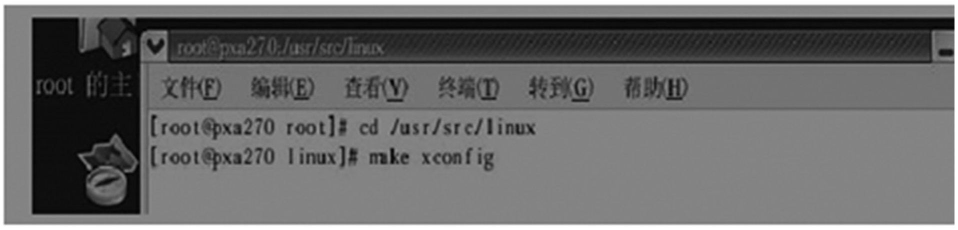
Coarse grain tailoring the kernel configuration, can improve the code reuse, to simplify the program, saving storage space. Embedded system cut and configure the kernel routine has four basic ways: Make config, make old config, make menuconfig and make xconfig. Different ways of kernel configuration, according to the code change operation process, this paper selected the make xconfig mode configuration, which has a menu configuration option, visual, simple operation process, the specific operation platform of Linux main control computer by Xwindow interface in Figure 4 (a), (b) is given.

For the Linux kernel provides configuration, gives the Y, N and M three kinds of choices, the specific meaning of the function were compiled into the kernel, do not choose to compile and when the compiler is stored as a real-time dynamic module to wait for the need to insert the kernel. In the actual process of compiling, according to the relationship between the function of the code and the kernel configuration of resource, make a reasonable choice, if the function code usage rate is very low, the compiler to load the module, in order to improve the kernel memory utilization rate, and reduce the consumption. In this paper, in order to realize the design of the embedded system kernel build a minimum, multiple function module configuration choices for *N*.

2.3 Fine Grained Tailoring Kernel Function and Function Library

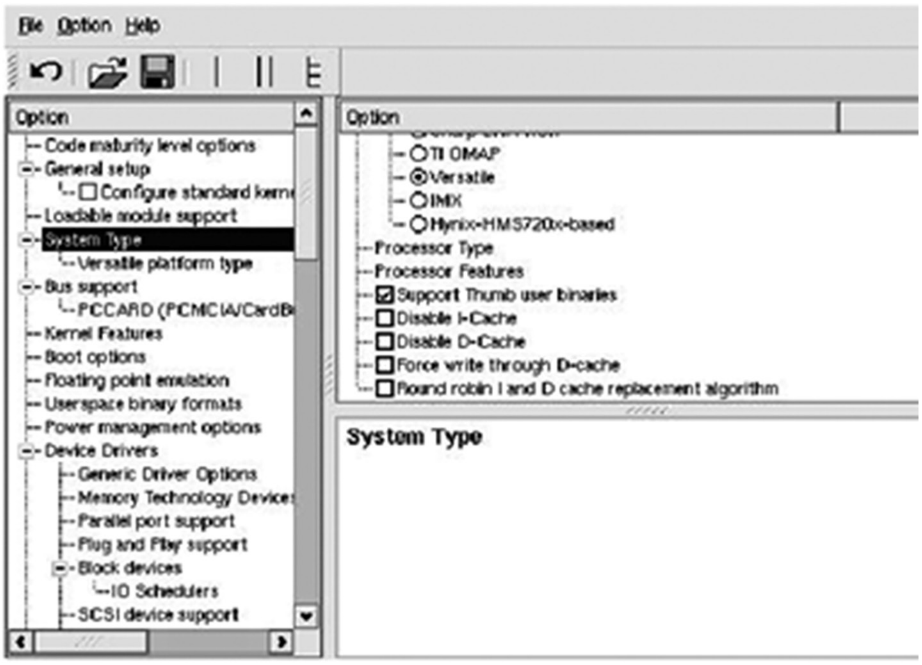
Fine grained tailoring is mainly through the function to analyze program, select the system to achieve a particular function must be a function or a library function, and eliminate system independent function package, in order to realize the kernel space optimization.

Table 1 gives the library function of embedded system application design must call, and gives the corresponding relation between the library and the kernel function. Through this table can eliminate system independent function package, such as strcpy.


root 的主
root@pxa270:/usr/src/linux
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)

```
[root@pxa270 root]# cd /usr/src/linux
[root@pxa270 linux]# make xconfig
```

(a) Start kernel configuration program



```
Option
--Code maturity level options
⊟-General setup
   └--☐ Configure standard kern
--Loadable module support
⊟-System Type
   └--Versatile platform type
⊟-Bus support
   └--PCCARD (PCMCIA/CardB
--Kernel Features
--Boot options
--Floating point emulation
--Userspace binary formats
--Power management options
⊟-Device Drivers
   ├-Generic Driver Options
   ├-Memory Technology Devices
   ├-Parallel port support
   ├-Plug and Play support
   ⊟-Block devices
   │  └--IO Schedulers
   ├-SCSI device support
```

```
Option
   --○ TI OMAP
   --◉ Versatile
   --○ IMX
   --○ Hynix-HMS720x-based
--Processor Type
--Processor Features
--☑ Support Thumb user binaries
--☐ Disable I-Cache
--☐ Disable D-Cache
--☐ Force write through D-cache
--☐ Round robin I and D cache replacement algorithm
```

(b) *x* config configuration interface

Figure 4
Coarse-Grained Clipping Kernel Configuration Interface Diagram

Table 1
The Corresponding Relationship Between the Function and the Kernel Function

	sys_brk	sys_brk	sys_brk	……	sys_brk	sys_brk	sys_brk
……	……	……	……	……	……	……	……
malloc	1	0	0	……	0	0	0
free	1	0	0	……	0	0	0
getpid	0	0	0	……	0	0	1
fork	0	1	0	……	0	0	0
exit	0	0	1	……	0	0	0
strcpy	0	0	0	……	0	0	0
printf	0	0	0	……	1	0	0
write	0	0	0	……	1	0	0
close	0	1	0	……	0	0	0
mmap	1	0	0	……	0	0	0

To be continued

Continued

3. SWITCH MACHINE AND TASK RESPONSE OF THE SYSTEM PERFORMANCE TEST

This paper is to construct a real-time response capability of embedded Linux system, the system must has the function of switch machine, restart, simple file management and task response and so on, so the system must be related to start and task response test. System under different environmental conditions, intermittent performance test carried out 720 hours of each test, at least more than 4 hours of continuous work, let the test system of intermittent perform basic function and task response, after more than a year of time for practical application test.

First to shutdown and restart the open test, system startup time through the system function of top and uptime can be obtained. In order to verify accuracy of the starting time of the system at the same time, using the third party equipment (precision timer) test. After repeated tests, get the system start time Table 2 shows; in the test of task response function , start the electric system, correct connection adapters and external test port, according to the application software of related message, testing, system files embedded timing related procedures, the program can automatically record the task response time. At the same time, plus third party device to detect specific test data, see Table 3.

Table 2
The System Data Sheet of the Starting Time

Project	Conditions	Number	Test requirements	Test date	Precision timer measurement data
Start time	Location: Common indoor Temperature: +18℃±1℃ Humidity: 50±3%	1	5s	4,600ms	4601.1ms
		2	5s	4,300ms	4300.2ms
		3	5s	4,600ms	4600.1ms
		4	5s	4,500ms	4500.9ms
		5	5s	4,600ms	4600.1ms
		6	5s	4,500ms	4500.2ms
	Location: High and low temperature laboratory Temperature: -45℃±0.5℃ Humidity: 50±3%	1	5s	4,900ms	4900.0ms
		2	5s	4,800ms	4800.3ms
		3	5s	4,700ms	4700.5ms
		4	5s	4,700ms	4700.5ms
		5	5s	4,600ms	4600.1ms
		6	5s	4,800ms	4800.2ms
	Location: High and low temperature laboratory Temperature: +60℃±0.5℃ Humidity: 50±3%	1	5s	4,800ms	4800.2ms
		2	5s	4,800ms	4800.2ms
		3	5s	4,700ms	4700.5ms
		4	5s	4,700ms	4700.5ms
		5	5s	4,600ms	4600.1ms
		6	5s	4,800ms	4800.2ms
	Location:	1	5s	4,600ms	4600.1ms
		2	5s	4,900ms	4900.0ms
		3	5s	4,300ms	4300.6ms
	heat / rain laboratory temperature: +18℃±1℃ Humidity: 95±3%	4	5s	4,700ms	4700.5ms
		5	5s	4,500ms	4500.9ms
		6	5s	4,800ms	4800.2ms
	Location: EMC laboratory Temperature: +18℃±1℃ Humidity: 95±3% Electromagnetic compatibility testing project: CE102, CS101, CS114, CS115, CS116, RS102, RS103,	1	5s	4,800ms	4800.2ms
		2	5s	4,800ms	4800.2ms
		3	5s	4,900ms	4900.0ms
		4	5s	5,000ms	4999.0ms
		5	5s	4,900ms	4900.0ms
		6	5s	4,800ms	4800.2ms

Table 3
Test Data Table Response

Project	Conditions	Testing tasks	Test requirements	Test date	Precision Timer measurement data
response time	Location: Common indoor Temperature: +18℃±1℃ Humidity: 50±3%	Self-inspection	20ms	17.8ms	17.7ms
		Communication	20ms	16.0ms	16.0ms
		Image display	20ms	18.1ms	18.0ms
		Shooting	20ms	19.0ms	19.0ms
		Fault diagnosis	20ms	16.4ms	16.5ms
		Image compression and storage	20ms	19.2ms	19.2ms
	Location: High and low temperature laboratory Temperature: -45℃±0.5℃ Humidity: 50±3%	Self-inspection	20ms	17.8ms	17.8ms
		Communication	20ms	17.2ms	17.1ms
		Image display	20ms	18.3ms	18.3ms
		Shooting	20ms	19.1ms	19.1ms
		Fault diagnosis	20ms	16.9ms	16.8ms
		Image compression and storage	20ms	19.1ms	19.1ms
	Location: High and low temperature laboratory Temperature: +60℃±0.5℃ Humidity: 50±3%	Self-inspection	20ms	17.9ms	17.8ms
		Communication	20ms	16.9ms	16.9ms
		Image display	20ms	18.2ms	18.1ms
		shooting	20ms	19.3ms	19.2ms
		Fault diagnosis	20ms	16.6ms	16.6ms
		Image compression and storage	20ms	19.1ms	19.1ms
	Location: Heat / rain laboratory Temperature: +18℃±1℃ Humidity: 95±3%	Self-inspection	20ms	17.8ms	17.9ms
		Communication	20ms	16.3ms	16.3ms
		Image display	20ms	18.1ms	18.1ms
		Shooting	20ms	19.2ms	19.3ms
		Fault diagnosis	20ms	16.0ms	16.0ms
		Image compression and storage	20ms	19.0ms	19.0ms
	Location: EMC laboratory Temperature: +18℃±1℃ Humidity: 95±3% Electromagnetic compatibility testing project: CE102, CS101, CS114, CS115, CS116, RS102, RS103	Self-inspection	20ms	18.9ms	18.8ms
		Communication	20ms	17.9ms	17.8ms
		Image display	20ms	18.2ms	18.1ms
		Shooting	20ms	19.7ms	19.6ms
		Fault diagnosis	20ms	18.6ms	18.6ms
		Image compression and storage	20ms	20.0ms	19.9ms

Known from the analysis of Tables 2 and 3 test results, The designed array CCD aerial camera embedded Linux system scheme is feasible, effective tailoring method. Through laboratory test and practical aerial examination, results show that the system is stable, reliable, and have the basic functions and tasks of embedded Linux system response function, reached the technical index requirements that start time is less than 5s and the task response time is less than 20 ms. Therefore, the tailoring system meets the design requirements.

CONCLUSION

Study of the Linux kernel tailoring technology, by adopting the combination of coarse-grained and fine-grained tailoring method, to cut operating system of the studied area array CCD aerial camera, the two aspect of the system after tailoring and kernel preemption real-time scheduler for scheduling performance has been greatly enhanced and improved. Through laboratory test and practical aerial assessment, show that the system boot time is less than 5s, the task response time is less than 20 ms, the system is running stable, reliable, and in this paper, the author studies on the tailoring technology for the same type embedded system performance also has the strong practical application value.

REFERENCES

Barbalace, A., Luchetta, A., Manduchi, G., Moro, M., Soppelsa, A., & Taliercio, C. (2008). Performance comparison of VxWorks, Linux, RTAI, and xenomai in a hard real-time application. *Nuclear Science, 6*(2), 435-439.

Daniel, P., Bovet, D. P., & Marco C. C. (2005). *Understanding the Linux Kernel* (3rd ed.). O'Reilly.

Lee, C.-T., Hong, Z.-W., & Lin, J.-M. (2003). *Linux kernel customization for embedded systems by using call graph approach.* Proceedings of the 2003 conference on Asia South Pacific design automation ASPDAC, ACM.

Lehrbaum, R. (1993). Using PC/104 embedded-PC modules for embedded applications. *WESCON, 13*(1), 633-638

Palesko, C. A., & Palesko, A. C. (2007). Understanding the cost effectiveness of embedded technology. *Electronic Manufacturing Technology Symposium, 22*(8), 114-117

Ra Jendra, Y., Prashant, P., & Raphael, F. (1994). A reservation based CSMA protocol for integrated manufacturing networks. *IEEE Transactions on Systems Man and Cybem etics, 24*(8), 1247- 1258.

Schach, A. J., Jin, S. R., & Wright, B., et al. (2002). Maintainability of the Linux kernel. *Software, IEE Proceedings, 12*(149).

von Hagen, W. (2005). Real-time and performance improvements for the 2.6 Linux kernel. *Linux Journal, 6*.