

Development of Home Remote Monitoring System Based on Mini2440 Board

Hongliang Song ^a, Jinxiao Cao ^b, Tongzhuang Liu ^c, Jinsong Li ^d

College of Mechanical and Electronic Engineering, Shandong University of Science and Technology, Qingdao, Shandong 266590, China

^askdshl@163.com, ^b2322716244@qq.com, ^c1085418527@qq.com, ^d2315901638@qq.com

Abstract

In order to develop intelligent home remote online monitoring system, can be realized function of real-time monitoring and automatic. The system adopts Linux system and Mini2440 development board building a complete system with the help of USB camera and general external audio system. It monitors the room through motion mobile object monitoring program, and uses embedded Web server to realize remote monitoring. Experimental results show that the system can not only realize real-time scene monitoring and alarm, but also realize remote Web server monitoring.

Keywords

Remote monitoring, S3c2440 microprocessor, Motion, Web services.

1. Introduction

With the progress of society and the continuous development of urbanization, the occurrence of burglary in the community is also more common, which makes the residents of residential communities pay more attention to their own personal safety, the safety of private property and public property, and people's awareness of security is also on the rise[1]. The traditional residential security and access control system can't realize the monitoring of the home environment of a single-family resident in a community, thus resulting in the theft of private property of some residents[2].

Aiming at the security problem of single-family residential home environment in residential area, this paper presents a method to monitor the home environment and develops its own monitoring system. The Mini2440 board used s3c2440 microprocessor as the core and the USB Camera, external speakers and other peripherals, through the embedded Linux operating system to build a living room mobile monitoring system. The residential mobile object monitoring system can realize on-site monitoring and remote monitoring functions. Residents who are not at home can remotely monitor and control their own home through the network; not only can home security be realized, but also electrical equipment can be controlled to meet people's intelligence Home needs.

2. System design

According to the above-mentioned home remote monitoring system hardware design, use mini2440 development board, ZC301P camera and general audio constitute a hardware device, the system software using embedded Linux system. The overall system design structure see Fig 1.

According to the system's overall design structure, the main hardware components of the monitoring system are: microprocessor, onboard peripherals, USB camera, LCD screen, audio, stepper motor, power supply and remote PC. Which mini2440 development board peripherals include: FLASH, SDRAM, JTAG download interface, on-site control buttons and DM9000 network card. In the whole system S3C2440 microprocessor as the core, USB camera image and video capture, and use SDRAM for data storage, the camera lens and stepper motor can be combined to achieve the monitoring area, remote monitoring through the remote PC access to the Web Browser way to achieve. For the monitoring site, through the LCD display mobile objects in the room, alarm equipment, audio interface through the use of peripherals to connect the keys used to achieve on-site control. In addition,

the power supply for the development version of power supply, FLASH used to load the system running program, monitoring system program downloaded to the development board through JTAG.

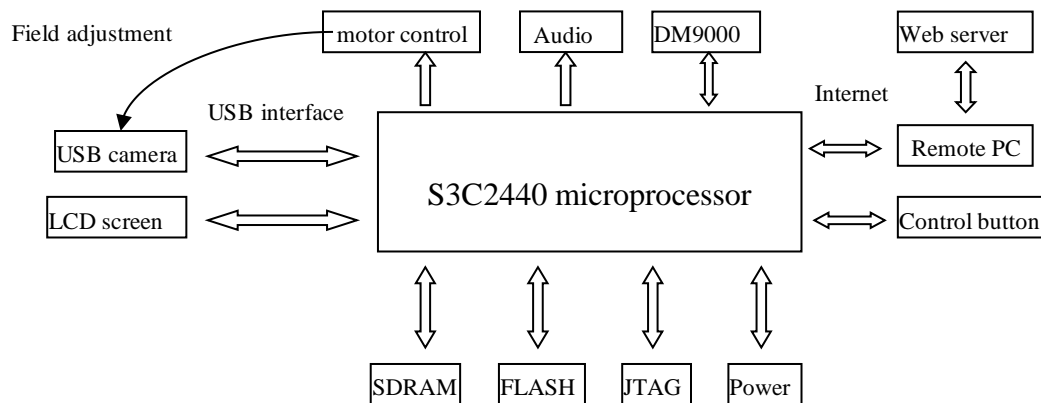


Fig. 1 The overall system design structure

3. Device Drivers Transplant

This part of the system used for the main equipment: USB camera and audio equipment, the two related drivers for transplantation and configuration introduced in order to achieve peripheral camera and audio control of the control functions.

3.1 USB camera driver transplant

The system selects the ZC0301P camera to realize the image video acquisition. The development board operating system uses the linux-2.6.32.2 kernel, which can integrate the ZC0301 camera driver to facilitate the camera-driven transplantation [3]. Inux-2.6.32.2 kernel can well support the use of USB, the ZC0301 camera driver configuration:

Use #make menuconfig enter the configuration menu;

Use the configuration menu followed by the following configuration:

Device drivers--->

<*>Multimedia support--->

[*]Video capture adapters --->

[*]V4L USB devices(NEW)--->

<*> USB ZC0301[P] Image Processor and Control Chip support

Save the configuration and exit, the above compiled kernel downloaded to the development board for testing.

3.2 Sound card driver and the transplant of Madplay player

Sound card drivers are mainly OSS and ALSA two architectures, as OSS as an open audio system can be used for audio playback. However, due to the flaw of OSS design, its poor support for mixing playback, coupled with OSS part of the code is not open source, there is a big disadvantage; ALSA Linux system as a senior audio architecture, which is mainly provided for the sound card driver components. In order to achieve better playback quality [4], the ALSA-OSS package is a combination of two architectures for the sound card driver. The main operation of the sound card driver transplantation is as follows:

Use the configuration menu for sound card configuration: # make menuconfig;

Device Drivers -> <*> Sound card support ->

-Sound card support

<*> Advanced Linux Sound Architecture ->

-Advanced Linux Sound Architecture

```

<*> OSS Mixer API
<*> OSS PCM (digital audio) API
[*] Verbose proofs contents
<*> ALSA for SoC audio support ->
-ALSA for SoC audio support
<*> SoC Audio for the Samsung S3C2440 chips
<*> SoC I2S Audio support UDA1341 wired to a S3C2440
Re-execute # make zImage, generate arm-linux kernel sound card driver downloaded to the mini2440
development board test.
For the audio player using the traditional Madplay player, its four packages constitute [5]. Audio
equipment through the audio interface cable and development board connected to achieve audio alarm.
Madplay player migration steps are as follows:
Running mkdir/mp3, extract the Madplay Player installation package to the mp3 directory:
tar -xzvf madplay-0.15.2b.tar.gz -C /mp3
tar -xzvf libid3tag-0.15.1b.tar.gz -C /mp3
tar -xzvf ibmad-0.15.1b.tar.gz -C /mp3
tar -xzvf zlib-1.1.4.tar.gz -C /mp3
Editing zlib:#cd/mp3/zlib-L1.4 --> #./configure
--prefix=/opt/EmbedSky/4.3.3/arm-none-linux-gnueabi/libc/usr/lib
CC=/opt/EmbedSky/4.3.3/bin/arm-linux-gcc
AR=/opt/EmbedSky/4.3.3/bin/arm-linux-arcc
RANLIB=/opt/EmbedSky/4.3.3/bin/arm-linux-ranlib
#make --> #make install
Editing libid3tag:#cd/mp3/libid3tag-0.15.1d
#./configure-host=arm-linuxCC=arm-linux-gcc--disable-debugging
--disable-shared
prefix=/opt/EmbedSky/4.3.3/arm-none-linux-gnueabi/libc/usr/lib
#make -->#make install
Editing libmad:#cd/mp3/libmad-0.15.1b
#./configure--enable-fpm=arm/host=arm-linux--disable-shared
-disable-debugging
prefix=/opt/EmbedSky/4.3.3/arm-none-linux-gnueabi/libc/usr/lib
modify the Makefile 129 line removed -fforce-mem
#make --> #make install
Editing madplay:#cd /mp3/madplay-0.15.2b
#./configure-host=arm-linux-gcc --disable-debugging-disable-shared
#make --> #rm madplay
Get statically linked program, download madplay to the development board test results.

```

4. Realization of moving object detection

4.1 Moving object detection algorithm

For the moving object detection using frame difference method, which has the advantages of small amount of calculation, fast speed, high real-time, not easily affected by light [6]. The basic principle of the frame difference method: the use of the camera to capture a number of frames of the image, set

a fixed threshold in advance T , and then adjacent two frame-by-pixel subtraction of images, the subtraction results with the pre-set T comparison subtraction. If the value is larger than the threshold, the object is moved, otherwise, the object is not moved. The corresponding calculation formula is as follows:

$$d(x, y) = \begin{cases} 1, & |f_{i+1}(x, y) - f_i(x, y)| \geq T \\ 0, & |f_{i+1}(x, y) - f_i(x, y)| < T \end{cases} \quad (1)$$

In the formula, $d(x, y)$ represents the image-processed binarized image, $f_{i+1}(x, y)$, $f_i(x, y)$ represents the adjacent image before and after images; T for the threshold set in advance. For the setting of the threshold T must be reasonable, if T is too small will lead to misjudgment of white Gaussian noise; In contrast, the image will be empty and missed inspection.

4.2 Motion main program transplant

The motion is a set of free and open source dynamic image monitoring algorithm library of UNIX operating system, which uses the V4L/V4L2 interface function provided by Linux operating system to capture and encode video equipment. Since the main motion program is completely written in C language, the high quality of the generated object code and the high efficiency of program execution are especially suitable for the occasions when the hardware resources of the embedded system are scarce [7]. The main steps of its migration procedures:

- 1) `#mkdir /home /motion --> #cp motin-3.3.2.tar.gz /home/motion`
- 2) `#tar xzvf motion-3.3.2.tar.gz --> #mkdir/opt/motion`
- 3) Running `#make` generate executable files; download the executable file to the mini2440 development board testing.

4.3 Transplanting Boa and CGIC

As a compact embedded Web server, Boa's executable code is around 60KB and performs single-task operations. Boa can only complete the user's request in turn. Boa can fork a process for CGI program to execute. Boa is designed to be fast and secure; web pages can be accessed by IP address [8]. Specific transplantation steps are as follows:

`tar -xzvf boa-0.94.13.tar`

Generate Makefile: Enter the src directory of Boa source code, execute `./configure` command to generate Boa configuration file and some source code and Makefile.

Modify Makefile: Find `CC = gcc` Modify to `FCC = gcc = arm> limix-gcc`, change `FCPP = gcc-EF` to `FCPP = arm-linux-gcc-E` and save and exit.

Fix bugs in Boa.

Boa server configuration: the use of `boa.onf` as a basis for modification, and finally the revised configuration file copied to the development board root file system `/etc/boa` directory.

For CGI common gateway interface programs need to rely on CGIC library, CGIC as a powerful support for CGI development of the standard C library, can support Linux, Unix and windows and other operating systems. The CGIC library migration process is as follows :

1) `tar -xzvf cgic205.tar.gz`,

2) Modify the `cgic205` source directory Makefile file:

`CC = gcc` changed to `CC = arm-linux-gcc`;

`AR = ar` changed to `AR = arm-linux-ar`;

`RANLIB = ranlib` changed to `RANLIB = arm-linux-ranlib`;

`gcc cgictest.o -o cgictest.cgi $ {LIBS}` changed to `$ (CC) $ (CFLAGS) cgictest.o -o cgictest.cgi $ {LIBS}`;

gcc capture.o -o capture \$ {LIBS}, changed to: \$ (CC) \$ (CFLAGS) capture.o -o capture \$ {LIBS}, and save and exit.

3) Compile and install: Copy the compiled cgic library file (libcgic.a) and the test files capture, cgic.html, and cgictest.cgi to the / lib directory and / web / cam of the development board's file system respectively.

4) Configure Boa: modify the CGI program path, so that the web server supports CGI, the ScriptAlias boa.conf file is modified to ScriptAlias / cam / / web / cam /; browser / cam directory and the input file system / web / cam directory corresponding.

5) Run boa program, enter the mini2440 development board ip address /cam/cgictest.cgi in the remote PC browser browser, run the show to see if the result is normal.

5. System web interface and testing

5.1 Web interface production

Remote monitoring for the home remote monitoring system requirements, the web interface has four main modules: Home, system introduction, monitoring images and video surveillance. The page "Home" displays the current monitoring result. The system introduces the function of the system briefly. The monitoring picture displays the collected picture information of the moving object. The monitoring video stores the moving object in the form of video for the user Watch, the interface shown in Figure 2.



Fig. 2 System Web interface

5.2 System test

First use the JTAG download interface to download the application to the development board, and then access the various peripherals mini2440 development board, to the development board power supply system running; In front the USB camera field of view waving hand, observe the effect.

After the test, when the hand waving sound will sound an alarm, indicating that the monitoring system to the normal operation of the site; the same time, enter the development board in the remote PC link to the web site to view the monitoring picture and monitoring video in the display area will have the appropriate information, Indicating that its remote monitoring system to normal operation, Figure 2 shows the monitoring area of a picture effect.

6. Conclusion

For home security issues, design and development based on the mini2440 home remote monitoring system. The system can rely on the USB camera to monitor the moving objects in the room in real

time, and can capture the video and image information uploaded to the remote embedded web server. The homeowner can log in to view detailed information in the webpage. Meanwhile, When the monitor will sound the alarm sound. The system is proved to be working normally and achieve the expected result.

References

- [1] Hou J L, Zhang Q C, Jun F U. Design of Wireless Remote Monitoring System for Water Quality Based on OneNet Platform[J], Journal of Huaiyin Institute of Technology, 2016.
- [2] Niti Witthayawiroj,Pongpon Nilaphruek. The Development of Smart Home System for Controlling and Monitoring Energy Consumption using WebSocket Protocol[J]. Materials Science and Engineering,2017,185(1).
- [3] Duoyou, Zhensheng, WANG. The Development of Intelligent Home Remote Monitoring System based on B/S Framework[J]. International Journal of Technology Management, 2015(2):63-65.
- [4] Yi Feng Hou. Research on Remote Monitoring and Control System Based on Communication Technology by Power Line Carrier[J], Applied Mechanics and Materials, 2013,2748(427).
- [5] Shang-Liang Chen,Shu-Kai Chang,Yun-Yao Chen,Zheng Xu. Development of a Multisensor Embedded Intelligent Home Environment Monitoring System Based on Digital Signal Processor and Wi-Fi[J]. International Journal of Distributed Sensor Networks,2015,2015.
- [6] Wei Li,Er Qing Lu. Design of Femtocell-Based Smart Home System Gateway[J]. Applied Mechanics and Materials,2014,3047(530).
- [7] Wei H X, Zhan W J, Kang Y M, et al. Embedded system designer course [M]. Beijing: Qinghua University Press, 2006: 2.
- [8] Wei W, Jiang A P, Lu T J, et al. Design of an 8-bit low-cost embedded MCU core[J]. Microcomputer Information, 2007, 29: 1-3.