



TQ2440 Development Platform Manual





Directory	
Directory	
Chapter 1 Introduction	7
1.1 Appearance of TQ2440 development platform	7
1. 2 Hardware resource of TQ2440	
1.3 Software introduction of TQ2440 platform	
1.4 Introduction of TQ2440 appendix CD-ROM	
Chapter 2 Hardware and Software Environment of Platform	
2. 1 Hardware structure	
2.1.1 Hardware resources allocation	
2.1.2 Instruction of jumper	
2. 1. 3 Instruction of TQ2440 development platform interface	
2. 2 TQ2440 schematics	
2. 2. 1 power-supply circuit	
2. 2. 2 System reset circuit	
2.2.3 System clock circuit	
2.2.4 JTAG interface circuit	
2, 2. 5 Serial port circuit	
2. 2. 6 key-press circuit	
2. 2. 7 Buzzer PWM driver circuit	
2. 2. 8 IIC circuit	
2. 2. 9 SD card interface circuit	
2. 2. 10 Real-time clock standby battery power circuit	
2. 2. 11 USB interface circuit	
2. 2. 12 IIS audio data interface circuit	
2. 2. 13 ADC circuit	
2. 2. 14 Ethernet interface circuit	
2. 2. 15 System bus interface	
2. 2. 16 Camera interface	
2. 2. 17 LCD/STN interface circuit	
2. 2. 18 Instruction of Core Board interface	
2. 3 Software characteristics	
2. 3. 1 Linux characteristics	
2. 3. 2 WinCE characteristics	
2. 3. 3 U-Boot characteristics	
2.4 Windows environment configuration	
2. 4. 1 Hyper-terminal configuration	
2. 4. 2 DNW software configuration	
2. 4. 3 GIVEIO driver intallation	
2. 4. 4 USB download-driver installation	
2. 5 Linux environment configuration	
2. 5. 1 Build cross-compile environment	
2. 5. 2 Network File Service (NFS) configuration	





2. 5. 3 PC Linux FTP service configuration	59
2. 5. 4 PC Linux Telnet service configuration	
2. 6 WinCE environment configuration	
Chapter 3 Platform Utilization	
3.1 Introduction of wire connection on platform and PC	
3. 2 Burning u-boot by SJF2440	
3. 2. 1 Executing SJF2440 with batch-processing files	
3. 2. 2 Running SJF2440	
3. 2. 3 Burning u-boot to Nor Flash	
3. 2. 4 Burning u-boot to Nand Flash	
3.3 Burning u-boot with H-Itag	
3. 3. 1 H-JTAG installation	
3. 3. 2 H-JTAG configuration	
3. 3. 3 H-Flasher configuration	
3.4 u-boot application	
3.4.1 Function introduction	
3.4.2 burning u-boot	
3, 4. 3 Configuration and estimation of u-boot start-up state	
3. 4. 4 Introduction of u-boot utilization	
3.5 Burning system	
3. 5. 1 Burning Linux OS	
3. 5. 2 Burning WinCE OS	
3. 5. 3 Burning TQ2440-Test_xxxx.bin file	
3. 6 Linux experiment	
3. 6. 1 Experiment of program termination	
3. 6. 2 Experiment of program auto-run configuration	
3. 6. 3 Experiment of setting and saving system real-time clock	
3. 6. 4 Experiment of USB mobile storage device test	
3. 6. 5 Experiment of SD card test	
3. 6. 6 Experiment of mounting NFS	
3. 6. 7 Experiment of USB camera capturing screen test	
3. 6. 8 Experiment of sound card test	
3. 6. 9 Experiment of files transmission with PC via serial port	
3. 6. 10 Experiment of screen capture	
3. 6. 11 Experiment of user LED test	
3. 6. 12 Experiment of user keyboard test	
3. 6. 13 Experiment of log in BBS in telnet	
3. 6. 14 Experiment of remote login platform by telnet	
3. 6. 15 Experiment of FTP remote file transmission	
3. 6. 16 Experiment of Web server test	
3. 6. 17 Experiment of USB camera remote control	
3. 6. 18 Experiment of user LED remote control	
3. 7 WinCE experiment	





3. 7. 1 Experiment of USB mobile storage device test	
3. 7. 2 Experiment of SD card test	
3. 7. 3 Experiment of Flash power failure protection	
3. 7. 4 Experiment of player utilization	
3. 7. 5 Experiment of 100M network card test	
3. 7. 6 Experiment of telnet remote log-in	
3. 7. 7 Experiment of FTP remote file transmission	
3. 7. 8 Experiment of Web server test	
3.7.9 Experiment of touch-screen correction	
3. 7. 10 Experiment of USB camera test	
3. 7. 11 Experiment of image rotation in LCD	
3. 7. 12 Experiment of 3 serial ports test	
3. 7. 13 Experiment of recording test	
3. 7. 14 Experiment of surfing by IE explorer	
3. 7. 15 Experiment of USB synchronization by ActiveSync	
3.7. 16 Experiment of WinCE self-carried game test	
3.7.17 Experiment of Zuma game test	
3. 8 Experiment of non-OS testing Demo	
3. 8. 1 Experiment of PWM function test	
3. 8. 2 Experiment of real-time clock test	
3. 8. 3 Experiment of ADC conversion test	
3. 8. 4 Experiment of external interrupt test	
3. 8. 5 Experiment of touch-screen test	
3. 8. 6 Experiment of LCD test	
3. 8. 7 Experiment of IIC interface test	
3. 8. 8 Experiment of audio output test	
3. 8. 9 Experiment of audio input test	
3. 8. 10 Experiment of SD card test	
3. 8. 11 Experiment of camera test	
Chapter 4 Linux Development Manual	
4. 1 Compiling bootloader	
4. 2 Compiling Linux-2. 6 kernel	
4. 2. 1 Use configuration file of EmbedSky to compile Linux kernel	
4. 2. 2 Customizing Linux kernel	
4. 3 Making root file system	
4. 3. 1 Components of root file system	
4. 3. 2 BusyBox compiling	
4. 3. 3 Building root file system	
4. 3. 4 Making Yaffs root file system image	
4.4 Compiling u-boot	
4.4.1 Decompressing u-boot	
4.4.2 Configuring u-boot	
3.4.3 Compiling u-boot	





Chapter 5 WinCE development manual	
5. 1 Installing BSP of TQ2440	
5. 2 Compiling the example projects in CD-ROM	
5. 3 Customizing user project files	
5. 4 Export SDK	
5. 4. 1 Configure SDK	
5. 4. 2 Compiling SDK	
5. 4. 3 Finishing the compiling	
5.5 Communicating with PC synchronously by using ActiveSync	
5. 5. 1 Installing ActiveSync	
5. 5. 2 Installing synchronizing communication USB driver	
5. 5. 3 Utilize ActiveSync synchronizing software	
5.5.4 Transferring file with ActiveSync	
5.6 Capturing screen with ActiveSync and Platform Builder connection	
5. 6. 1 Configuring platform	
5. 6. 2 Configuring PC	
5.7 Cheek the platform register information based on connection between ActiveSync	and Platform Builder
Chapter 6 Qt/Embeded graphics development	
6.1 Simulating Qt/Embeded in PC	
6. 1. 1 Configuring the running evironment	
5. 1. 2 Installation and compiling	
5. 1. 3 Enable the created library	
5. 1. 4 Simulating Qtopia in PC	
6. 2 Programming	
6. 2. 1 Hello routine	
6. 2. 2 Configuring environment parameters	
6. 2. 3 Making desktop starter file	
6. 2. 4 Making the icon	
6. 2. 5 Copying Hello.desktop	
6. 2. 6 Running hello solely	
6. 2. 7 Running hello in Qtopia	
6. 3 Transplanting Web explorer	
6. 4 Building development environment based on ARM	
6. 4. 1 Installing development environment	
6. 4. 2 Make running script	
6. 4. 3 Compiling hello for ARM	
6. 4. 4 Installing hello	
6. 4. 5 Running hello solely on the platform	
6. 4. 6 Running hello in Qtopia on the platform	
6. 4. 7 Make desktop starter file	
6. 4. 8 Make file system containing Qtopia	





7.1 Application development in Linux	
7. 1. 1 Hello EmbedSky experiment	
7.1.2 Experiment of calling math function	
7.1.3 Experiment of thread programming	
7.1.4 Experiment of UDP network programming	
7.1.5 Experiment of controlling LED	
7.1.6 Experiment of user button controlling	
7.2 Example of driver development in Linux	
7. 2. 1 Hello EmbedSky experiment	
7.2.2 Experiment of LED driver	
7.2.3 Experiment of user keyboard driver	
7.3 Experiment of Non-OS application	
7.3.1 Configuring the experiment environment	
7. 3. 2 Experiment of test	
7. 3. 3 Burning TQ2440-Test.bin file	
Chapter 8 Transplantation of Web server	
8.4 Transplanting boa software	
8-1.1 Building compiling environment	
8. 1. 2 Configuring the compiling condition	
8. 1. 3 Compiling and optimizing	
8. 2 Transplanting cgic library	
8. 2. 1 Building compile environment	
8. 2. 2 Configuring compile condition	
8. 2. 3 Compiling and optimizing	
8. 3 Configuring Web server	
8. 3. 1 Configuring boa	
8. 4 Testing	
8. 4. 1 Static web page test	
8. 4. 2 CGI script test	
8. 5 Solving the error	
8. 5. 1 Error 1	
8. 6 Some source codes	
8. 6. 1 cgictest.cpp source code	
8. 6. 2 Source code of home page	
8.7 Environment variable	
8.7.1 The environment variable relevant to the server	
8.7.2 The environment variable relevant to the customer	
8.7.3 The environment variable relevant to the request	
Chapter 9 Embeded database transplantation (SQLite)	
9.1 Introduction of SQLite database	
9.2 Transplanting SQLite database	
9. 2. 1 Get SQLite source code	
9. 2. 2 Transplanting SQLite	





9. 2. 3 Using SQLite	
9. 3 Testing SQLite database	

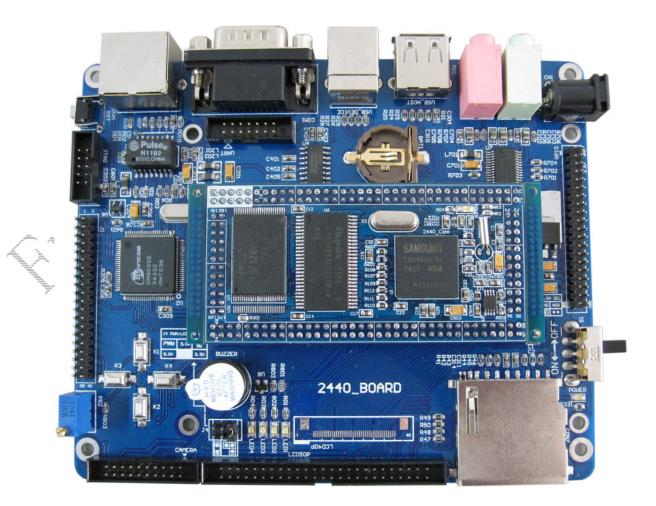






Chapter 1 Introduction

1.1 Appearance of TQ2440 development platform



1.2 Hardware resource of TQ2440

TQ2440 development platform is composed of a Core Board and Mother Board, to make secondary development easier, The CD-ROM contains reference PCB files.



TQ2440 Core Board specifications



CPÚ

Samsung S3C2440AL, 400MHz, highest frequency: $533MHz_{\circ}$

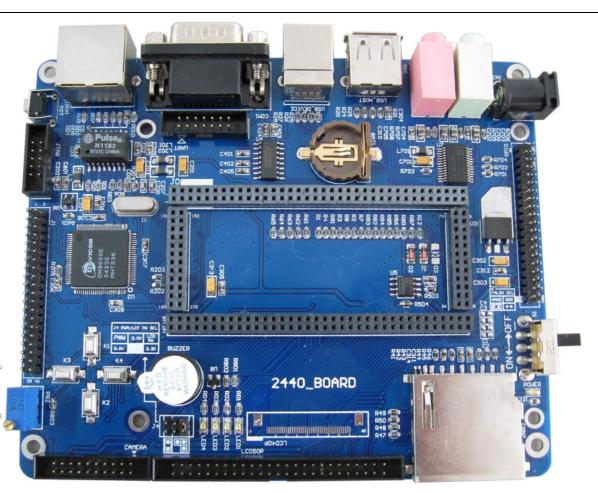
SDRAM

- On-board 64MB SDRAM
- 32bit data bus
 - SDRAM clock frequency as high as 100MHz
- ♦ Flash ROM
 - On-board 256MB Nand Flash, nonvolatile
 - On-board 2MB Nor Flash
- 1.25V main power supply , fully solving the CUP heating problem
- A power indicator light
- 3.3V Core Board power supply

TQ2440 Mother Board specification







- Provide 3.3V power supply, load current up to 1.5A, excellent power supply can avoid issues resulting from power instability;
- ♦ LCD interface

- On-board LCD interface contains a 4-line resistance type touch-screen interface which can connect to a 4-line resistance type touch-screen directly.

- Supporting STN LCD of monochrome, 4-level gray, 16-level gray, 256-level gray and 4096-level color, sized from 3.5 inch to 12.1 inch, resolution up to 1024×768 pixels.

- Supporting TFT LCD of monochrome, 4-level gray, 16-level gray, 256-level gray and 64K-level color and true color, sized from 3.5 inch to 12.1 inch, resolution up to 1024×768 pixels.

- Standard TQ2440 suite contains TFT true color TFT LCD of 256K-level color, 3.5 inch dimension and 240×320 pixels resolution. And a touch-screen.

- A 100M Ethernet RJ-45 interface.
- 3 serial port interfaces. COM1 is a DB9 interface with voltage converted by RS3232 (Including interface of CMOS voltage level). The UART interface on-board is namely the extended interface of serial port with CMOS voltage level.
- A USB type Host A interface (supporting USB1.1 protocol).
- A USB type Slave B interface (supporting USB1.1 protocol).
- A SD memory card interface, supporting DMA transmission mode.
- 1-way stereo audio output interface, 1-way audio input interface.
- A 2.0mm, 10-Pin Jtag interface, which can be used for software simulation and step debugging and





download u-boot.

- 4 User Buttons.
- ♦ 4 on-board user LED lights.
- On-board AD test unit
- On-board PWM function test unit (buzzer).
- On-board EEPROM test unit.
- A 20-Pin, 130 megapixels CMOS camera interface.
- A 40-Pin GPIO extended interface.
- A 40-Pin bus extended interface.
- On-board real-time clock battery.
- Power switch and indicator light

Specification size

- ◆ 78mm×37mm (Core Board)
- ▶ 128mm×105mm (Mother Board)

1. 3 Software introduction of TQ2440 platform

Operating system provided

- ♦ Window CE 5.0
- ♦ Linux-2.6.13
- ♦ uCOS-II

Bootloader provided

• u-boot-1.1.6 (supporting TFTP transmission and burning yaffs file system)

Test program provided

• Non-OS test program

Schematics provided

- Schematics of Core Board (PDF format).
- Schematics and PCB diagram of Mother Board (original diagram for board manufacturing).

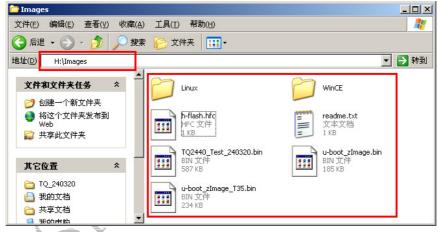
1.4 Introduction of TQ2440 appendix CD-ROM

Images directory

The directory contains compiled image files, including uboot image, non-os test program image, linux image and WinCE image (under the directory WinCE)







 \rightarrow Linux directory

The compiled Linux image files, including: kernel image, simplified file system image, Qt image supporting touch-screen version and Qt image of mouse version.

	🚞 Linux			
	文件(E) 编辑(E) 查看(/) 收藏(<u>A</u>)	工具(I) 帮助(H)	2
A > I	🕞 后退 🔹 💮 🕤 🏂	● 搜索	* 🍺 文件夹 🛛 🎹 •	
	地址(D) H:\Images\Linux	<		▼ 🗲 转到
	文件和文件夹任务	*	readme.txt 文本文档 1 KB	root_condence.img IMG 文件 14,024 KB
	创建一个新文件夹 裕这个文件夹发布部 Web 学共享此文件夹	ðj	root_qt_mouse.img IMG 文件 57,158 KB	THUCE TRUE THUCE TRUE THUCE TRUE THUCE TRUE THUCE TRUE
	其它位置	*	u-boot_zImage.bin BIN 文件 185 KB	u-boot_zImage_T35.bin BIN 文件 234 KB
	 Images 我的文档 共享文档 我的史帖 	•	Image 2.6_T35_uboot 6_T35_UBOOT 文件 1,547 KB	

→ WinCE directory

Í

Compiled WinCE image file, including: eboot image and NK image.

Caution: the resource code of uboot and eboot image are not contained in CD-ROM; If there is a need of eboot resource code, please contact us.

🗁 WinCE		
文件(E) 编辑(E) 查看(V) 收藏(A	A) 工具(<u>T</u>) 帮助(<u>H</u>)	R
🕝 后退 🔹 🕥 🖌 🏂 🔎 捜索	素 🕞 文件夹 🛄 -	
地址(D) H:\Images\WinCE		转到
文件和文件夹任务 ☆	EBOOT_dm9000_u-boot.nb0 NB0 文件 88 KB NB0 文件 23,278 KB	
 创建一个新文件夹 格这个文件夹发布到 Web 共享此文件夹 	readme.txt 文本文档 1 KB	
	u-boot_zImage_T35.bin BIN 文件 234 KB	
 ☐ Images ☐ 我的文档 ☐ 共享文档 3 我的电脑 	1	

➢ Linux directory

Containing: linux kernel, file system, Qt/Embeded, Busybox, uboot, database, web server, cross-compiler





and example program.

Caution: The source code of uboot contained in CD-ROM does not support USB download mode, but only TFTP download mode.

WinCE directory

Contains: BSP package, project files and SDK package.

→ SMDK2440 directory

WinCE 5.0 BSP package

 \rightarrow TQ2440 directory

Project file sample, the file NK.bin under the directory Images\WinCE\ is compiled from it.

 \rightarrow SDK directory

The SDK package exported from project file sample.

Windows platform tools

Containing general softwares and drivers.

 \rightarrow ActiveSync

WinCE synchronizing software, needs to be intalled to activate WinCE synchronization function.

→ DNW

Running in PC server, used for transmitting data to platform in USB download function.

→ GIVEIO

A driver used to virtualize the parallel port into normal IO driver when using Jtag.

→ H-JTAG

Used for software simulation and burning uboot into Nor Flash.

 \rightarrow Jave patch

This patch is indispensable when accessing USB camera via web explorer.

→ SJF2440

A program running in PC for Jtag download.

 \rightarrow TFTP_Server_TFTPDWIN_v0.4.2

A proxy software runnig in PC for TFTP download.

 \rightarrow USB driver

The driver used for USB download function.

 \rightarrow VMware

A software used for installing Linux in Windows

 \rightarrow WinCE synchronization driver

A USB driver used for synchronization between WinCE and PC

```
Circuit diagram
```

Containing Core Board schematic, Mother Board schematic and Mother Board PCB diagram.

Information from Samsung website

Containing some resources downloaded from Samsung company website.

Non-OS test program





The test program running independent from OS.

➢ Chip manual

Containing information of all chips appearing in development platform.

➢ User manual

Í

Containing development information of TQ2440







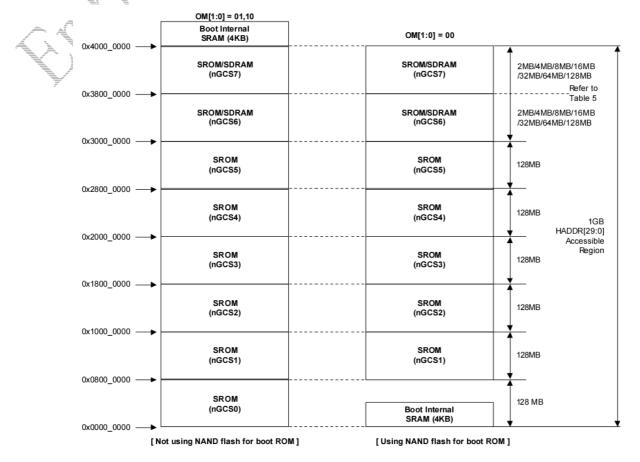
Chapter 2 Hardware and Software Environment of Development Board

2.1 Hardware structure

2.1.1 Hardware resources allocation

Address space allocation and chip selection signal definition

S3C2440 supports 2 kinds of start-up modes: Start up from Nand Flash, and start up from Nor Flash. The address allocations of chip selection are different in these 2 modes, as shown in the following diagram:



The upper chart on the left indicates the address allocation in Nor Flash start-up mode of nGCS0; The upper chart on the right describepts the address allocation in Nand Flash start-up mode.

(Note: Use the jumper OM0 in Mother Board to select Nand Flash and Nor Flash start-up mode. Nand Flash start-up mode is selected when the jumper is fit on, and Nor Flash start-up mode is selected when the jumper is removed.)

In the upper graph, the Nand Flash start-up mode, 4KB Boot Internal SRAM in CPU is mapped into the

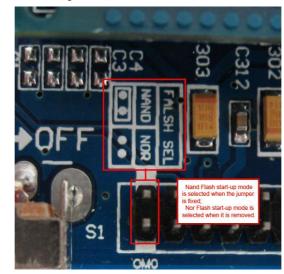


chip-selection space of nGCS0. Before a program starts, CPU copies the first 4KB part of the proram into this space. Then the program starts to run. If a program is larger than 4KB, the first part of 4KB needs to include the code initializing Nand Flash and other devices, and copyingies data from Nand Flash into SDRAM on-board, and jumping to SDRAM space on-board from the 4KB space;

Under the Nor Flash start-up mode, nGCS0 is mapped from the beginning address of Nor Flash. SDRAM address space on-board: 0x3000000~0x34000000.

2.1.2 Explanations of jumper

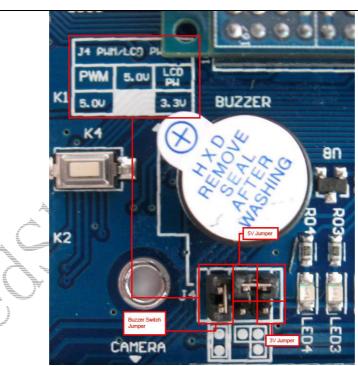
There are 2 jumpers in TQ2440 platform (J4 in Mother Board of TQ2440 and OM0 of GPIO interface): The jumper OM0 of GPIO extended interface refers to Boot Select, the Nand Flash start-up mode is selected when the jumper is fixed. Nor Flash start-up mode is selected when it is removed.



The role of jumper J4: Switch on LCD power, and switch on the buzzer, as shown in the following diagram.

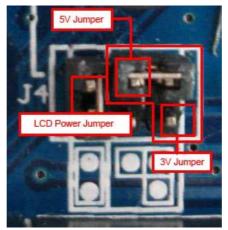






The "L" shaped jumper interface refers to switch on LCD powering, selecting 3.3V or 5V voltage to power LCD.

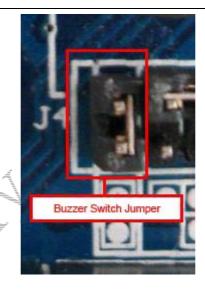
Samsung 3.5 inch LCD and Donghua 3.5 inch LCD needs a 3.3V power; Toshiba 3.5 inch LCD and Samsung 7 inch LCD needs a 5V power. As shown in the following diagram.



The "I" shaped jumper interface plays a role of power switch of the buzzer. The power is on when the jumper is fixed, and is off when it is removed.

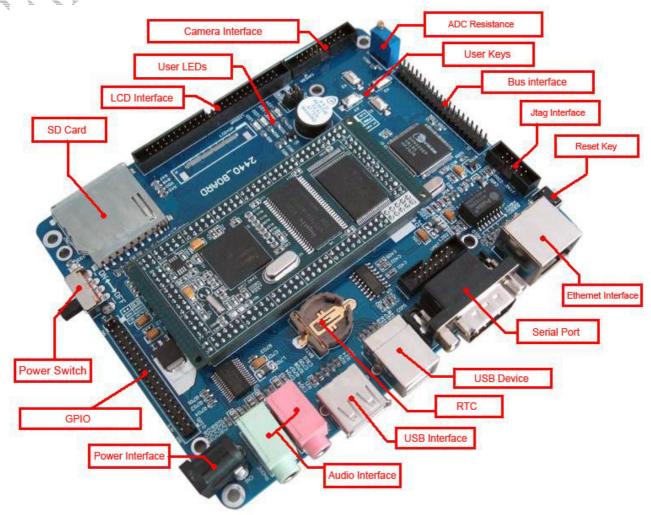






2.1.3 Explanations of TQ2440 development platform interface

The main interfaces of TQ2440 is shown in the following picture:







• Power interface

Caution: the input power needs to be less than 7V. The output voltage of power adapter provided is 5V. Make sure to confirm your own power-adapter output by consulting the supplier or by measuring with a reliable device, in order to prevent from an excessive high input.

Caution: A reliable 5V power-adapter, like the one provided by us is the best choice for the power source.

♦ Audio interface

TQ2440 provides standard audio interfaces, the green one is for output, and the red one for input.

• USB interface

TQ2440 development board provides 2 USB interfaces, one is USB A (USB_HOST interface, be used to connect to the U disk, USB camera and other devices); The other is USB B (USB_Deive interface, connect to PC with a USB extension-line to transmit data).

When using USB for download function, using a standard USB extension-line to connect PC and development platform. Advice: Remove the extension-line immediately after the download is complete.

♦ Serial port

Serial port is one of the most important interfaces of TQ2440 development board. It is used for interaction and data transmission between platform and PC, and debugging.

We offer a standard direct-connecting serial port line to connect development platform and PC for interaction.

Network card interface

TQ2440 carries a 100M network card interface. The user can connect the platform with a net line and go surffing in operating system; In uboot-download mode, the user could download data to development platform by using TFTP via network card interface.

♦ Jtag interface

In TQ2440, if there is no uboot in Nand Flash or in Nor Flash. Using Jtag to burn uboot; Or using Jtag for simulation

Connect Jtag interface and PC parallel port with Jtag line, using Jtag software to burn program or to simulate. Note: Remove the Jtag line if not using it.

♦ Camera interface

When connecting the camera, keep lens outward.

♦ LCD interface

Beware of the direction of LCD interface and don't make it reversed when connecting.

Caution: When using Toshiba LCD, be careful don't touch the high-voltage region on backside of the LCD driver board, In order to prevent from electronic shock.

♦ SD card interface

Insert the SD card with its interface side downwards.

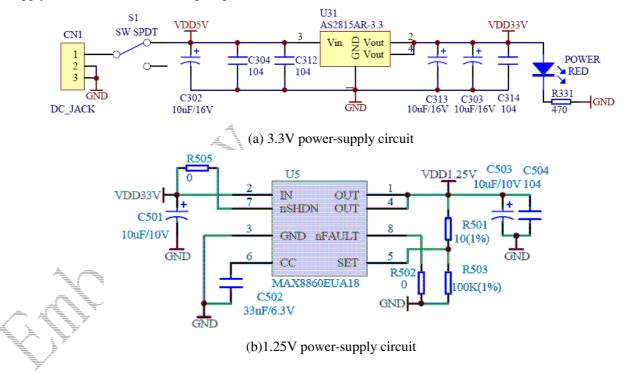
2.2 TQ2440 schematics

2.2.1 power-supply circuit

TQ2440 supports a 5V input. The platform contain power switch and indicator light. A Low Dropout Line Regulator AS28 M76D7V33915AR-3.3 IC which could carry 1.5A load provides platform with 3.3V power supply.



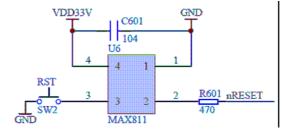
A Low noise and Low Dropout Line Regulator MAX8860EUA provides CPU on Core Board with 1.25V power supply. As shown in the following diagram:



2. 2. 2 System reset circuit

,

A system reset chip MAX811S is selected to enhance the power supervision ability. If the system voltage is lower than the threshold 2.93V, MAX811S resets the system immediately. As shown in the following diagram:



2. 2. 3 System clock circuit

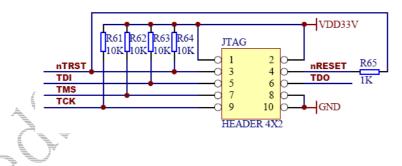
An external oscillator is intended for the source of system clock; The internal PLL circuit could adjust the system clock to speed up the system operation. The system clock circuit is shown in the following diagram:





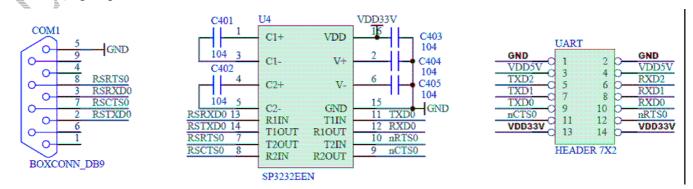
2. 2. 4 JTAG interface circuit

The 10-pin, 2.0mm clearance Jtag interface takes up only a small area on platform. S3C2440 supports JTAG function. The user can use external JTAG debug line or simulator for debugging. The circuit is shown in the following diagram:



2. 2. 5 Serial port circuit

There is a 5-line asynchronous serial port interface and a UART extension interface. The circuit is shown in the following diagram:

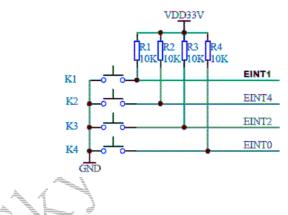


2.2.6 key-press circuit

TQ2440 provdes a 4-way key-press circuit, 4 pull-up $10k \Omega$ resistances are connected to 4 GPIO pins on CPU. The voltage level of GPIO turns from high to low when the key is pressed. By using polling program or interrupt the user can be acquainted with the voltage change of GPIO pins. The circuit is shown in the following diagram:

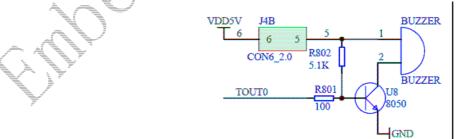






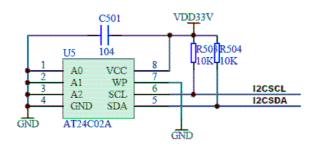
2. 2. 7 Buzzer PWM driver circuit

The buzzer on platform is driven by PWM, and produces sound with different frequencies. The circuit is shown in the following diagram:



2.2.8 IIC circuit

 $I^2 C$ circuit supports a multi-master-controlling $I^2 C$ bus serial interface. A serial data bus SDA and a serial clock bus SCL transmit data between master and slave device. SDA and SCL are both duplex and are used for reading and writing operation of AT24C02A. $I^2 C$ circuit is shown in the following diagram:



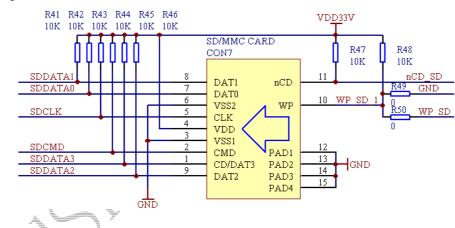
2.2.9 SD card interface circuit

SD (Security Digital) card is a kind of widely applied card. A specified interface circuit on platform supports reading and writing function of SD card. S3C2440 integrates SD module in itself. The circuit is shown in



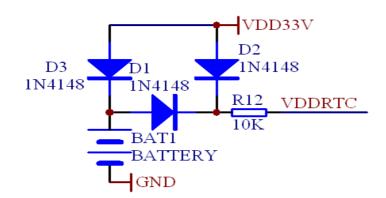


the following diagram:



2. 2. 10 Real-time clock standby battery power circuit

The standby battery protects system state data during power-off. The reference circuit is shown in the following diagram:



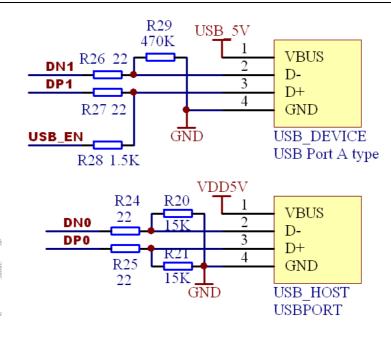
2. 2. 11 USB interface circuit

Í

S3C2440A integrates USB module itself, containing a HOST USB1.1 interface and a Device USB1.1 interface. The circuit is shown in the following diagram:







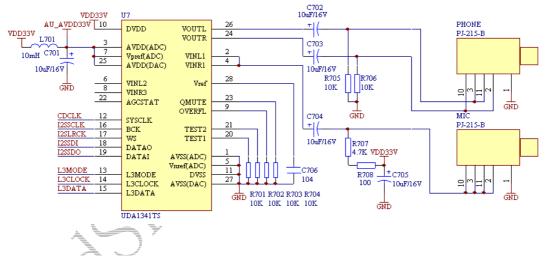
2. 2. 12 IIS audio data interface circuit

 I^2 S (Inter-IC sound bus), is a interface standard introduced by Philips and Sony company. The I^2 S circuit is shown in the diagram 2-12. The system connects I^2 S interface and UDA1341TS (provided by Philips company), to build SPEADER audio ouput channel and MICROPHONE audio input channel. UDA1341TS can convert digital signal into analog signal, and can convert analog stereo signal into digital signal, and can also use AGC (Auto Gain Control), PGA (Programmable Gain Adjust) to process analog signal. The chip supports DSP function in order to process digital signal. In practical application, UDA1341TS can be widely used in CD, MD and digital camera. The bus L3 of UDA1341TS is used when the chip works in micro-controller-input mode. L3 includes L3DATA (data line of interface), L3MODE (mode line of interface) and L3CLOCK (clock line of interface). The processor can configure the audio-processing parameter and system-controlling parameter of UDA1341TS via this interface.

As shown in the following diagram:

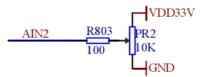






2.2.13 ADC circuit

S3C2440A has an 8-way, 10-bit CMOS A/D converter, and 3.3V reference voltage. The platform carries a DC voltage test circuit. The resistance PR2 is used to adjust input voltage. AIN2 is the analog voltage input. As shown in the following diagram:

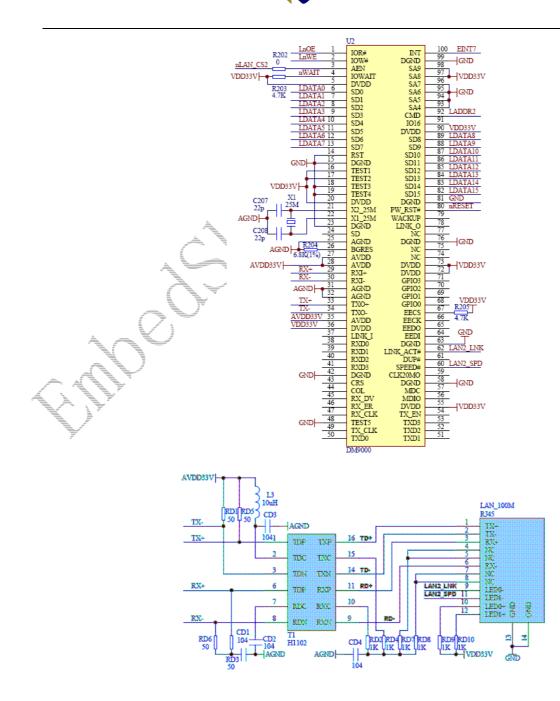


2. 2. 14 Ethernet interface circuit

S3C2440A carries no network interface. By adopting extension-network-interface mode, the platform provides a DM9000E 100M network interface. The circuit is shown in the following diagram:





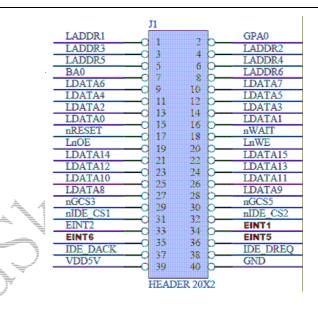


2. 2. 15 System bus interface

System bus interface is made up of 8 address-buses (address 0~6 and address 24), 16 data-buses, 4 interrupt-buses and 4 chip-selection buses. System bus integrates leading points needed by IDE for extension. The circuit is shown in the following diagram:







2. 2. 16 Camera interface

1

S3C2440A integrates the CAMERA module in itself. The user can connect different types of camera to the interface (like OV9650). The circuit is shown in the following diagram:

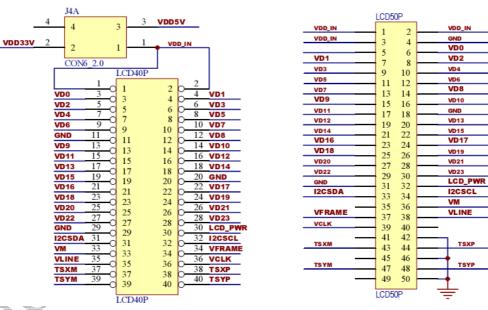
	CAMERA	
I2CSCL	2 1	12CSDA
CAMRST		EINT19
CAM_HREF		CAM_PCLK
CAMCLK	8 7	CAM_VSYN
CAMDATA6		CAMDATA7
CAMDATA4		CAMDATA5
CAMDATA2		CAMDATA3
CAMDATAO	16 15	CAMDATA1
VDD33V		VDD5V
GND		GND
******	20 19 CAM 2.0	

2. 2. 17 LCD/STN interface circuit

TQ2440 support 2 types of interface, 50-pin 2.0mm-clearance and 40-pin 0.5mm-clearance, which can support monochromatic, pseudo-color, true-color and touch-screen LCD interface. The circuit is shown in the following diagram:







Caution: Use jumper J4 to select 3.3V or 5V power for LCD.

Í

2.2.18 Instruction of Core Board interface

pin	function	pin	function	pin	function	pin	function
1	DATA6	2	DATA7	3	ADDR7	4	ADDR8
5	ADDR5	6	ADDR6	7	ADDR3	8	ADDR4
9	ADDR1	10	ADDR2	11	DATA30	12	DATA31
13	DATA28	14	DATA29	15	DATA26	16	DATA27
17	DATA24	18	DATA25	19	DATA22	20	DATA23
21	DATA20	22	DATA21	23	DATA18	24	DATA19
25	DATA16	26	DATA17	27	nTRST	28	nRESRT
29	TDO	30	TDI	31	TCK	32	TMS
33	RXD2/nCTS1/GPH7	34	TXD2/nRTS1/GPH6	35	RXD1/GPH5	36	TXD1/GPH4
37	RXD0/GPH3	38	TXD0/GPH2	39	nRTS0/GPH1	40	nCTS0/GPH0
41	EINT0/GPF0	42	EINT1/GPF1	43	EINT2/GPF2	44	EINT3/GPF3
45	EINT4/GPF4	46	EINT5/GPF5	47	EINT6/GPF6	48	EINT7/GPF7
49	EINT8/GPG0	50	EINT11/nSS1/GPG3	51	EINT14/SPIMOSI1/GPG6	52	EINT13/SPIMISO1/GPG5
53	EINT19/TCLK1/GPG11	54	EINT15/SPICLK1/GPG7	55	EINT18/nCTS1/GPG10	56	EINT9/GPG1
57	EINT20/GPG12	58	VDD_RTC	59	DP1/PDP0	60	AIN3
61	DN1/PDN0	62	AIN2	63	DN0	64	AIN1
65	DP0	66	AIN0	67	EINT13/SPIMISO1/GPG5	68	EINT10/nSS0/GPG2
69	SPICLK0/GPE13	70	SPIMOSI0/GPE12	71	EINT22/GPG14	72	EINT21/GPG13
73	Vref	74	EINT23/GPG15	75	OM2	76	OM3
77	OM0	78	OM1	79	EINT16/GPG8	80	SDDAT2/GPE9





In the upper definitions, "/" means the multi-function pin. The user can use software to define the pin function; The red letters indicate the default function chosen by platform; The blue letters indicate the audio function chosen by platform; The green letters indicate the network function chosen by platform.

2.3 Software characteristics

2.3.1 Linux characteristics

- Version: 2.6.13
- Support file system
- Yaffs readable and writable file system, default selection.
- Cramfs compressed read-only file system, not recommended.





- Ext2 available when connecting a hard-disk
- Fat32 be used in mobile storing device.
- NFS network file system, can be used for debugging applications and drivers
- The drivers included in CD (Caution: Camera drivers are provided in the form of module, and

others are provided in source code.)

- 3 standard serial port drivers.
- DM9000 driver.
- Sound card driver (support playing and recording).
- RTC driver (preserve time record when power-off).
- User LED driver.
- User key-press driver.
- Ordinary LCD drivers (including: resolution 320×240 , 240×320 and 640×480).
- Camera driver.
- Touch-screen driver.
- USB camera (including: OV511 camera and others).
 - USB mouse, USB keyboard, U disk, mobile hard disk and so on.
 - SD card driver.
 - ⁷ Linux application and service program.
 - busybox1.2.0 tool kit (includes general Linux instructions).
 - Telnet, FTP, inetd (remote login tools and services).
- Boa (Web server application).
- Madplay (MP3 player in console).

- Servfox (the camera application in console is able to access USB camera via network and display

the image captured by camera in LCD).

- Spcacat (the shooting application of camera in console).
- rz and sz (file receiving and sending application via serial port in console).
- Snapshot (image-capture software in console).
- camera_test (test program of camera in console).
- Graphic interface (the source code is provided).
- Qt/Embedded

2.3.2 WinCE characteristics

Windows CE 5.0

- Drivers in CD.
- DM9000 network card driver.
- USB wireless lan driver (VNUWLC41).
- Camera driver.
- USB mouse, USB keyboard, U disk and mobile hard disk driver.
- USB camera driver.
- USB synchronization driver.





- 3 serial ports driver.
- Sound card driver.
- SD card driver.
- real-time clock driver.
- register table preservation.
- Flash free space power-off preservation, about 30M free space.
- System characteristic (simplified Chinese system)
- Windows XP interface style.
- Windows Media Player 9.0 (supports MP3, MPEG2, MPEG4, WMV, WAV and so on).
- Picture explorer, wordpad.
- IE6 explorer.

2.3.3 U-Boot characteristics

The uboot of this platform support both USB download and WinCE download function. The other features supported are list in the following:

- Support Nand Flash and Nor Flash self-adaptive start up.
 - Support USB data transmission. The transmission rate reaches as high as 850KB/S.
 - Support TFTP network data transmission.
- Support start-up logo.
- Support WinCE and Linux start-up scroll bar.
- Support writing Yaffs file system.
- Support analysis and writing NK.bin file.
- Support WinCE and Linux self-adaptive start up.
- Support writing user program to Nand Flash.
- Support download user program to SDRAM.

2.4 Windows environment configuration

The following captured images might have some difference with the OS you have used in PC. If you have questions, please contact us.

2.4.1 Hyper-terminal configuration

We recommend using Window self-carried hyper-terminal for interaction between PC and TQ2440. Here we introduce the configuration based on Windows XP hyper-terminal.

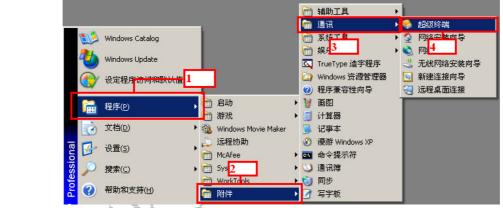
The following diagrams introduce the steps configuring the hyper-terminal:

Step 1, open "Start->Programs->Accessories->Communications->Hyper Terminal":



Í





A pop-up window "Default Telnet Program" appears, choose "No":

	は Telnet	程序?		? ×
	<u>.</u>	我们推荐您将 HyperTerminal 样做吗?	作为默认 telne	t 程序。要这
		□ 请不要再问这个问题(D)		
N^{-}		是①	否例	

A pop-up window "位置信息", fill in the blank "您的区号(或城市号)是什么?" with your district number, click "确定" and continue:

位置信息	<u>? </u> ×
	在您做任何电话或调制解调器连接之前,Windows 需要知道关于您当前位置的信息。 目前所在的国家 (地区) (w): 中华人民共和国
	」中华八天兴和国 您的区号(或城市号)是什么(C)?
	020 1
	您拔外线需要先拔哪个号码 @)?
	本地电话系统使用:
	● 音频拨号 (E) ● 脉 <mark>2</mark>)

A dialog box as the following one appears, choose "确定":



ź



电话和调制解调器选项		? ×
拔号规则		
下列显示了您指定的	的位置。选择您拨号的位置。	
位置(L):		_
位置 ③ 我的位置	区号 020	- 1
	晝 ⑭) │ 编辑 诓) │ 删除 ⑪)	
	确定 取消 应用 ()	()

Step 2, an interface as the following one appears. Name your hyper-terminal and select an icon. Then click "确定" to continue:

连接描述	? ×
新建连接	
输入名称并为 <mark>该连接选</mark> 择图标:	
名称 00: 1	
超级终端	
图标(工):	
确定 取消	i i

Step 3, an interface "连接到" as the following one appears. Select comN you connecting. N represents the serial port number of PC you are using. The example uses COM1 of PC. click "确定" and continue:

连接到	<u>? ×</u>
🗞 超级终端	
输入待拨电话的详细	田信息:
国家(地区)(C):	中华人民共和国(86)
区号(28):	020
电话号码(P):	
连接时使用 (2):	COM1
2	新 定 取消



Step 4, an interface "COM1 属性" appears, setting: "波特率: 115200, 数据位: 8, 奇偶校验: 无, 停止 位: 1, 数据流控制: 无". Click "确定" to continue:

c c	COM1 雇性	<u>?</u> ×
	端口设置	
	毎秒位数 (B): 115200 1	
	数据位 @): 8	
À	奇偶校验 (£): 无 💽	
	停止位 (2): 1	
	数据流控制 (E): 无2	
	3 还原为默认值 (8)	
	<u> </u>	(A)

Step 5, the Typer Terminal window appears. Click menu "文件" and select "保存" to save your configuration:

	呼叫(<u>C</u>) 传送(<u>T</u>) 养	
新建连接(N) 打开(0)		
保存(5)	2	
另存为(A) 页面设置(U) 打印(P)		
属性(<u>R</u>)		
退出(X) Alt+F4]	
保存当前会话		

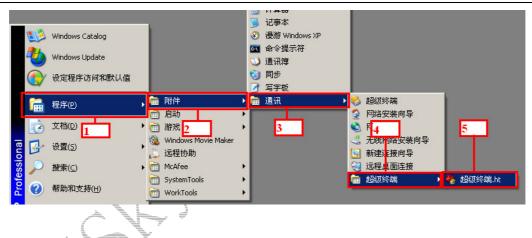
Step 6, in the future use, you can find the saved option "超级终端.ht" under "开始->程序->附件->通讯->超级终端". Click it as follows:

(you can create a shortcut on your desktop for convenience.)

ź



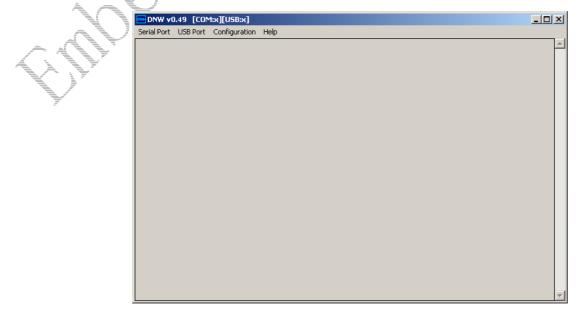




2.4.2 DNW software configuration

Í

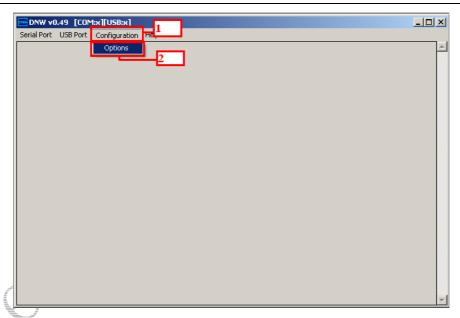
Find DNW software under directory "Windows 平台工具\DNW". Double-click to open it:



Step 1, click "Configuration -> Options", the configuration table "UART/USB Options" appears.







Step 2, choose "115200" of "Baud Rate", choose "COM1" of "COM Port" (choose the right one according to actual situation), fill in "0x32000000" of "USB Port", click "OK" to finish the DNW configuration:

UART/USB Options		
Serial Port		
Baud Rate	COM Port	ОК
① 115200	○ COM 1	Cancel
C 57600	C COM 2	
C 38400	C COM 3	
C 19200	C COM 4	
C 14400		
° 9600		
USB Port		
Download Address 0×32000000		

2.4.3 GIVEIO driver intallation

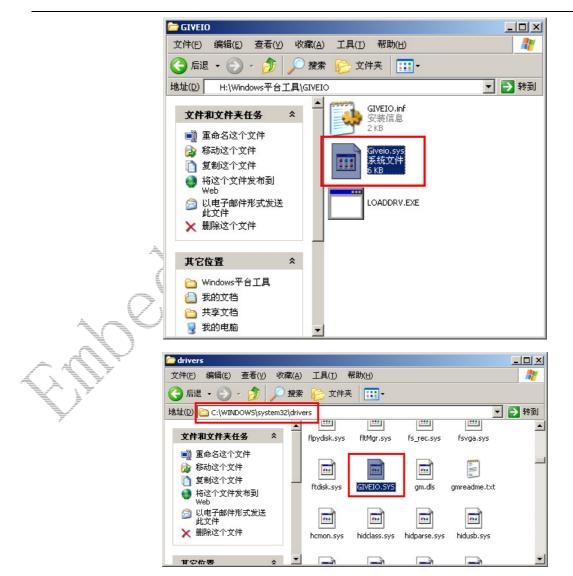
If the user wants to burn u-boot with the Jtage software SJF2440.exe, a driver needs to be installed to virtualize parallel port into IO port. Pay attention to the parallel port configuration under BIOS, SPP and EPP mode is recommended, but ECP mode is not recommended.

The steps how to install GIVEIO is shown in the following.

Step 1, Find giveio driver in the CD under the directory "Windows 平台工具\GIVEIO". Copy the file "giveio.sys" to your system disk, under the directory "WINDOWS\system32\drivers"







Step 2, open "控制面板" on your PC, double-click the icon "添加硬件" and enter the interface 添加硬件, click "下一步" to continue:







Step 3, system will find the hardware automatically. When the search is finished, the next window appears. Select "是,我已经连接了此硬件", and click "下一步" to continue:





	漆加硬件 向导	
	向导正在搜索,请稍候 ♥♥	2
	向导正在搜索最近连接到计算机但尚未安装的硬件。	
	< 上一步 (B) 下一步 (B) > 取消	
	添加硬件向导 硬件连接好了吗?	2
¢	 您已经将此硬件连接到计算机了吗? ① 是,我已经连接了此硬件(Y) ① 否,我尚未添加此硬件(H) 	
	2	
	〈上一步@ 下一步 @ 习 取消	1

Step 4, select the option as the following one in the appearing interface and click "T - #" to continue:





从以下列表,选择一个已安装的硬件设备,然后单击"下一步",检查属性或 解决您遇到的问题。 要添加列表中没有显示的硬件,请单击"添加新的硬件设备"。 已安装的硬件 (g):
要添加列表中没有显示的硬件,请单击"添加新的硬件设备"。
USB Root Hub
😴 USB Root Hub
♦ USB Root Hub ♦ USB Root Hub

Step 5, the interface "安装向导" appears. Select "安装我手动从列表选择的硬件(高级)" and click "下一步" to continue:

添加硬件向导
这个向导可以帮助您安装其他硬件
这个向导可以搜索其他硬件并为您自动安装。或者,如果您知道要安装哪个型 号的硬件,您可以从列表选择。
您期望向导做什么? 搜索并自动安装硬件(推荐)(S) 受装我手动从列表选择的硬件(高级)(M) 1
2 < 上一步 (B) 下一步 (B) 下一步 (B) 取消

Step 6, select "端口(COM 和 LPT)" among the hardware list and click "下一步" to continue:





装計	雨雨外台巴
0457	11211月子

从以下列表,选择要安装的硬件类型	
如果看不到想要的硬件类别,诸单击"显示所有设备"。 常见硬件类型(Y):	
 ● 打印机 ● 週制解调器 ● 第四 (COM 和 LPT) ■ 35 串ロ卡 ● 5 音、視測和游戏控制器 ● 声音、視測はため 	<u> </u>
● 图像处理设备 ■● 网络适配器 2	_
〈上一步 (8) 下一步 (8) 〉	取消

Step 7, click "从磁盘安装" and continue:

_

添加硬件向导		
选择要为此硬件安装的设备引	题动程序	
了 诸选定硬件的厂商和型号 程序的磁盘,请单击"从	,然后单击"下一步"。如 磁盘安装"。	果手头有包含要安装的驱动
<u>「商</u> ※ (标准端口类型) Cyclades Corporation Microsoft RATOC Systems, Inc.	型号 Communications Port 译 ECP Printer Port 译 ECP 打印机端口 计印机端口	
○ 这个驱动程序已经过数字签署 告诉我为什么驱动程序签名很		<u>从撤盡安装 (H)</u> 2
	< 上一步 ®)	下一步(12) > 取消

Step 8, select "浏览" in "从磁盘安装" interface:

从磁盘安	装	×
J	插入厂商的安装盘,然后确定已在下面选定正确 的驱动器。	
	厂商文件复制来源 (C): ▲: \	(XIK B)



1



Locate the previous GIVEIO directory. Find the file "GIVEIO.inf" and click " $\mathfrak{T}\mathfrak{H}$ " to continue:

查找文	件					<u>? ×</u>
查抄	滤围(重): ┃	CIVEIO		•	G 🖻 🖻 🖪	
我最	这 近的文档	GIVEIO.inf	L <u>1</u>			
	<mark>)</mark> 桌面					
我	分 的文档					
我	的电脑					
	夏 上邻居					2
		文件名 @):	GIVEIO. inf		-	打开(0)
ANL		文件类型 (I):	安装信息 (*.inf)		V	取消
Back to the inter	rface "从禄	菡盘安装", aı	nd click "确定" to	continue:		

从磁盘安	装	×
J	插入厂商的安装盘,然后确定已在下面选定正确的驱动器。2	 取消
	厂商文件复制来源 (C): 1 H:\Windows平台工具\GIVEIO	浏览 (2)

Step 9, back to the interface concerning installing device drivers. Select "giveio" device and click "T - #" to continue:





	添加硬件向导
	选择要为此硬件安装的设备驱动程序
	请选定硬件的厂商和型号,然后单击"下一步"。如果手头有包含要安装的驱动 程序的磁盘,请单击"从磁盘安装"。
	型묵 giveio
	♪ 这个驱动程序没有经过数字签署! 告诉我为什么驱动程序签名很重要
	< 上一步 @ 下一步 @ 下一步 @ 取消
Step 10, "1	可导准备安装您的硬件" interface appears. Click "下一步" to continue: 添加硬件向导
	承加使件目录 自导准备安装您的硬件
	要安装的硬件:
	giveio
	了一个 要开始安装您的新硬件,请单击"下一步"。

An interface appears warning that the driver has not been authenticated by Microsoft. Click "仍然继续" to continue:





硬件安装	
1	正在为此硬件安装的软件: giveio
	没有通过 Windows 徽标测试,无法验证它同 Windows XP 的相容性。(<u>告诉我为什么这个测试很重要。</u>) 继续安装此软件会立即或在以后使系统支得不稳定。 Bicrosoft 建议您现在停止此安装,并同硬件供应商 联系,以获得通过 Tindows 数标测试的软件。
<u> </u>	

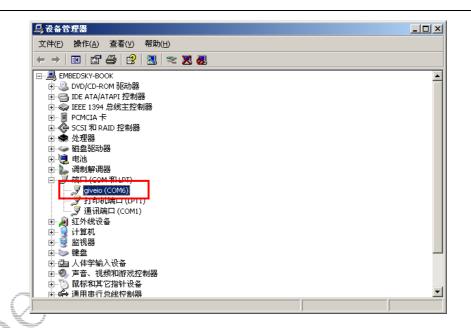
Step 11, click "完成" to finish the installation:



Step 12, the newly installed device could be found in "设备管理器":







2. 4. 4 USB download-driver installation

The following steps introduce how to install USB download-driver. The driver is located under the directory "Windows 平台工具\USB 驱动":



Step 1, open hyper-terminal, and link the serial port line and power line; press the space-key of PC and hold, and Switch on the power. The hyper-terminal will display the u-boot console (instruction: USB download-driver needs to be installed in u-boot console).



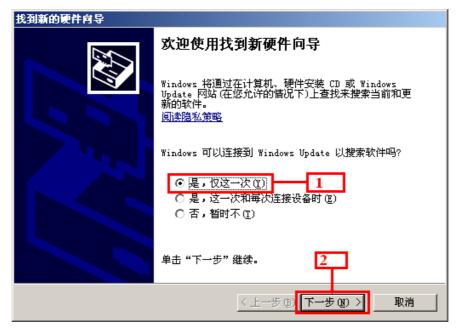


[1] Downloa [2] Downloa [3] Downloa [4] Downloa [5] Downloa [6] Downloa [7] Downloa [8] Boot th [9] Format	d Linux Kernel d WinCE NK.bin d CRAMFS image d YAFFS image d to SDRAM & Run e system the Nand Flash boot parameters	

Step 2, when linking the USB wire, Windows XP can recognize the new device automatically as the following diagram:



The interface "找到新的硬件向导" pops up. Select "是,仅这一次" and click "下一步" to continue:



Step 3, select "从列表或指定位置安装(高级)" in the next interface and click "下一步" to continue:



Step

Í



找到新的硬件向导		
	这个向导帮助您安装软件:	
664	TQ2440 Board	
	如果您的硬件带有安装 CD 或软盘,请现在将 其插入。	
	您期望向导做什么?	
	○ 自动安装软件(推荐)(I)	
	○ 从列表或指定位置安装(高级)(S) 1	
	要继续,请单击"下一步"。 2	
	〈上一步 (8) 下一步 (8) 〉 取消	
select"在搜索中包括这个伯	立置" in "在这些位置上搜索最佳驱动程序" menu and click	"浏览
找到新的硬件向导		

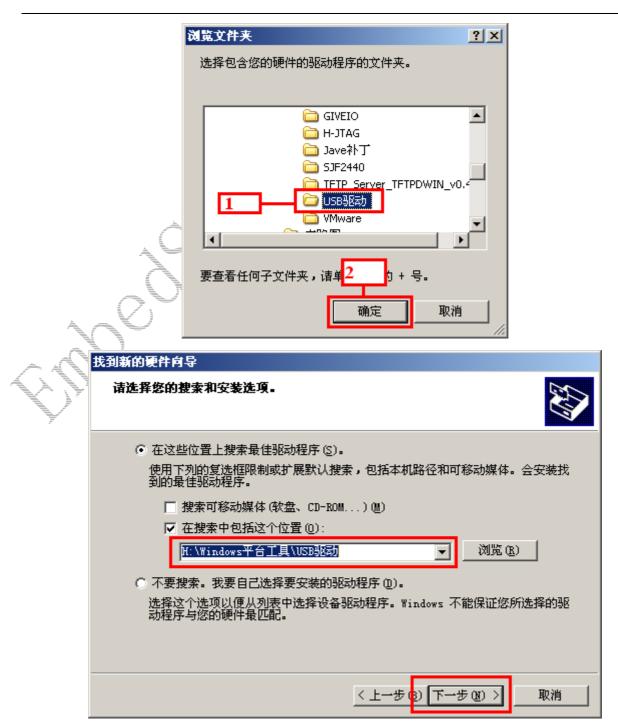
我到新的家件內守
请选择您的搜索和安装选项。
 ● 在这些位置上搜索最佳驱动程序(⑤)。 ● 伸下列的复选框限制或扩展默认搜索,包括本机路径和可移动媒体。会安装找到的最佳驱动程序。 ● 搜索可移动媒体(软盘、CD-ROM)(M) ● 在搜索中包括这个位置(0): ● 2 ● 不要搜索。我要自己选择要安装的驱动程序(0)。 3 选择这个选项以便从列表中选择设备驱动程序。Windows 不能床证怎些选择的驱动程序与您的硬件最匹配。
< 上一步 (B) 下一步 (B) > 取消

Locate the driver and click "确定" to go back to the upper diagram. Click "下一步" to continue:



_





Step 5, the installing guide begins to search hardware device:



1



	找到新的硬件向导	
	向导正在安装软件,请稍候	
	TQ2440 Board	
	6	\triangleright
		< 上一步 (B) 下一步 (B) > 取消
The fo	llowing diagram appears. Select "SEC SC	DC Test Board" and click "下一步" to continue:

找到新的硬件向导			
从下列表中选择与您的	的硬件的最佳匹配。		
SKY2440 Bo	pard		
描述	版本 制造商	位置	
TQ2440 Board	未知 天嵌科技,广州	h:\windows平台工具\usb跽	b∖s
✓ 这个驱动程序 告诉我驱动程序	 没有经过数字签署! - <u>签署的重要性</u>		Þ
	<	(上步@) 下步@) >	取消

Step 6, when installing the driver, the following interface appears. Click "仍然继续" to continue:





硬件安装	t de la companya de la
1	正在为此硬件安装的软件: TQ2440 Board
	没有通过 Windows 徽标测试,无法验证它同 Windows XP 的相容性。(<u>告诉我为什么这个测试很重要。</u>) 继续安装此软件会立即或在以后使系统变得不稳定。
	Bicrosoft 建议您现在停止此安装,并同硬件供应商 联系,以获得通过 Windows 数标测试的软件。
<u> </u>	
	「仍然继续で」」 停止安装 (5)

Step 7, the interface "所需文件" appears:

Í



Click "确定". Then the following interface appears. Click "浏览" to locate the driver:



Locate the file "secbulk.sys", and click "打开" to continue:





查找文件					<u>? ×</u>
查找范围(L):	🗀 ण्डा अक्रिक्री		• •	• 🖽 🔰 🏂	
我最近的文档	secbulk.sys	<u>L1</u>			
「「」「」「」「」」「」」「」」「」」「」」」「「」」」」」」」」」」」」」					
多 我的文档					
我的电脑					_
四月二日 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1995 - 1 1995 -					
	文件名(N):	secbulk. sys		•	打开(0)
	文件类型(I):	SECBULK. sys; SECBULK. sy		V	取消
Back to "所需文件	" interface, click	"确定" to continue:			
所需	文件				×
<i>w</i>	需要 USB D 列 TQ2440 上自	ownloader Installation I 句文件 'SECBULK.sys'。)isk for	确定	



输入文件所在的路径,然后单击"确定"。

取消

Step 8, click "完成" to finish USB download-driver installation:







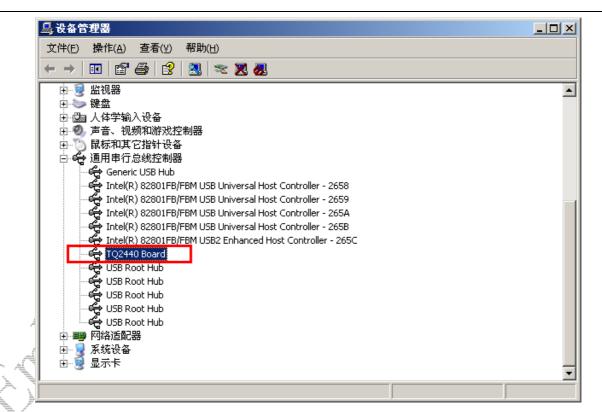
Step 9, after the USB download-driver has been installed, open DNW software. The mark "[COM:x][USB:OK]" could be found on top of the window:

		1:x][USB:OK]	
Serial Port	US <mark>B Port</mark>	Configuration	Help
			A

The USB driver installed previously could be found in "设备管理器":







Now the user can use USB to download u-boot, operating system and file system.

2.5 Linux environment configuration

It is suggested to refer to 《TQ2440 之 RedHat9 安装 step 》 before read this part

Instruction: In the latter Linux operation, the PC command line is executed on ordinary terminal; while the platform command line is executed on hyper-terminal. We attach symbol "#" to each PC operation instruction in order to distinguish these 2 kinds of command line. Please be aware of the difference.

The Linux compression package must be decompressed under Linux. The filename and command are case sensitive which is different from Windows. Please be aware of that.

2.5.1 Build cross-compile environment

The cross_compiler is the main environment under Linux. We introduce a process building a development environment which can compile arm-linux kernel, driver and application under RedHat 9.0.

Copy the compression package "crosstools_all.tar.bz2" from the directory "Linux" to the directory "/opt/EmbedSky/" in Linux system, and decompress the package under the current directory: (the following commands are executed in PC).

#cd /opt/EmbedSky

#tar xvfj crosstools_all.tar.bz2 -C /

After the upper operation, the compiler has been installed under the directory "2.95.3" and "3.3.2" of "/usr/local/arm/" and the directory of "/opt/EmbedSky/crosstools_3.4.1_softfloat/". The makefile will be installed





automatically under the directory "/usr/local/sbin/":

- The cross-compile compiler of 3.3.2 version is used to compile Qtopia/Embedded.
- The cross-compile compiler of 2.95.3 is used to compile VIVI and transplant boa.

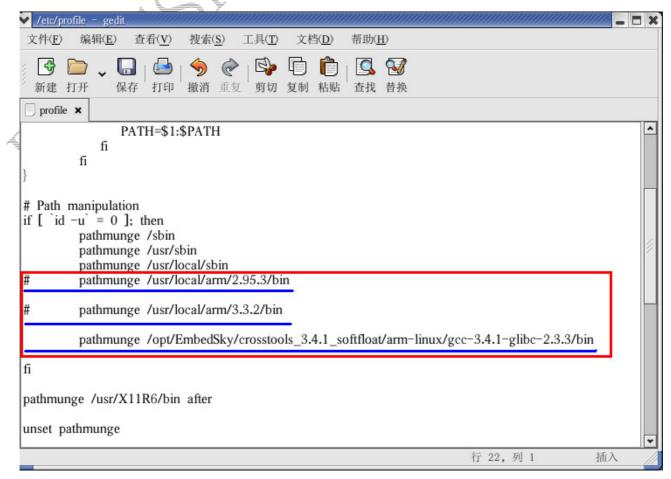
- The cross-compile compiler of 3.4.1_softfolat is used to compile kernel, busybox, u-boot and application.

Execute the command:

#gedit /etc/profile

Add the following information in "profile":

(the following frame contains the added information. If the user tries to use the cross-compiler of a certain version, please remove its prefix "#", and add "#" to the head of other versions. The lines highlighted with blue underline as following diagram is required to be added the prefix "#")



Save "profile" and execute the following command to set a default cross-compiler:

#source /etc/profile

Execute the following command to check if the cross-compiler has been installed successfully and check the revised version:

#arm-linux-gcc -v

Get the following information:





文件(E) 编辑(E) 查看(V) 终端(T) 转到	
[root@EnbedSky root]# gedit /etc/profile [root@EnbedSky root]# source /etc/profil [root@EnbedSky root]# arm-linux-gcc -v	At present the cross-complier
Reading specs from /opt/EnbedSky/crossto glibc-2.3.3/lib/gcc/arm-linux/3.4.1/spec	ols_3.4.1_softfloat/arm-linux/gcc-3.4.1- s
.3.3/gcc-3.4.1/configuretarget=arm-li ix=/opt/EmbedSky/crosstools_3.4.1_softfl th-float=softwith-headers=/opt/EmbedS gcc-3.4.1-glibc-2.3.3/arm-linux/include ools_3.4.1_softfloat/arm-linux/gcc-3.4.1 nable-threads=posixenable-symvers=gnu s=c,c++enable-sharedenable-c99e	ol-0.28/build/arm linux/gcc-3.4.1-glibc-2 nuxhost=i686-host_pc-linux-gnupref oat/arm linux/gcc-3.4.1-glibc-2.3.3wi ky/crosstools_3.4.1_softfloat/arm linux/ with-local-prefix=/opt/EnbedSky/crosst -glibc-2.3.3/arm linuxdisable-nlse uenablecxa_atexitenable-language nable-long-long
Thread model: posix gcc version 3.4.1	

The cross-compiler version might be frequently changed. Use the previous command "gedit /etc/profile" to modify the file "/etc/profile", and use it to validate "source /etc/profile". Then execute the command "arm-linux-gcc -v" to check the revised version of cross-compiler.

文件E 编辑E 查看(V) 终端(T 转到(G) 帮助(E) [root@EnbedSky root]# gedit /etc/profile [root@EnbedSky root]# arm linux-gcc -v Reading specs from /opt/EnbedSky/crosstools_3.4.1_softfloat/arm linux/gcc-3.4.1- glibc-2.3.3/lib/gcc/arm linux/3.4.1/specs Configured with: /opt/crosstool/crosstool-0.28/build/arm linux/gcc-3.4.1-glibc-2 3.3/gcc-3.4.1/configuretarget=arm linuxhost=i686-host_pc-linux-gnuprefix=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm linux/gcc-3.3wi th-float=softwith-headers=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm linux/ gcc-3.4.1-glibc-2.3.3/arm linux/includewith-local-prefix=/opt/EnbedSky/crosst ools_3.4.1_softfloat/arm linux/gcc-3.4.1-glibc-2.3.3/arm linuxdisable-nlse nable-threads=posixenable-symvers=gnuenablecxa_atexitenable-language s=c,c++enable-sharedenable-c99enable-long-long Thread model: posix gcc version 3.4.1 [root@EnbedSky root]# gedit /etc/profile [root@EnbedSky root]# gedit /etc/profile [root@EnbedSky root]# gedit /etc/profile [root@EnbedSky root]# arm linux-gcc -v Reading specs from /usr/local/arm/3.3.2/lib/gcc-lib/arm-linux/3.3.2/specs Configured with:/gcc-3.3.2/configure target=arm linuxwith-cpu=strongarm 100prefix=/usr/local/arm/3.3.2 i686-pc-linux-gnuwith-headers=/work/kernel, h3900/includeenable-threads=pthreadsenable-sharedenable-staticenable -languages=c,c++ Thread model: posix gcc version 3.3.2 [root@EnbedSky root]# []	✓ root@EmbedSky: [^]		1 3
[root@EnbedSky root]# source /etc/profile [root@EnbedSky root]# arm linux-gcc -v Reading specs from /opt/EnbedSky/crosstools_3.4.1_softfloat/arm linux/gcc-3.4.1-glibc-2. 3.3/lib/gcc/arm linux/3.4.1/specs Configured with: /opt/crosstool/crosstool-0.28/build/arm linux/gcc-3.4.1-glibc-2. 3.3/gcc-3.4.1/configuretarget=arm linuxhost=i686-host_pc-linux-gnuprefix=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm linux/gcc-3.4.1-glibc-2.3.3wi th-float=softwith-headers=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm linux/gcc-3.4.1-glibc-2.3.3wi th-float=softwith-headers=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm linux/gcc-3.4.1-glibc-2.3.3/arm linux/includewith-local-prefix=/opt/EnbedSky/crosst ools_3.4.1_softfloat/arm linux/gcc-3.4.1-glibc-2.3.3/arm linuxdisable-nlse nable-threads=posixenable-symers=gnuenablecxa_atexitenable-language s=c,c++enable-sharedenable-c99enable-long-long Thread model: posix gcc version 3.4.1 [root@EnbedSky root]# gedit /etc/profile [root@EnbedSky	文件(E) 编辑(E) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>)	帮助(<u>H</u>)	
[root@EnbedSky root]# gedit /etc/profile [root@EnbedSky root]# source /etc/profile [root@EnbedSky root]# arm-linux-gcc -v Reading specs from /usr/local/arm/3.3.2/lib/gcc-lib/arm-linux/3.3.2/specs Configured with:/gcc-3.3.2/configure target=arm-linuxwith-cpu=strongarm1 100prefix=/usr/local/arm/3.3.2 i686-pc-linux-gnuwith-headers=/work/kernel. h3900/includeenable-threads=pthreadsenable-sharedenable-staticenable -languages=c,c++ Thread model: posix gcc version 3.3.2	[root@EnbedSky root]# source /etc/profile [root@EnbedSky root]# arm-linux-gcc -v Reading specs from /opt/EnbedSky/crosstools_ glibc-2.3.3/lib/gcc/arm-linux/3.4.1/specs Configured with: /opt/crosstool/crosstool-0. .3.3/gcc-3.4.1/configuretarget=arm-linux ix=/opt/EnbedSky/crosstools_3.4.1_softfloat/ th-float=softwith-headers=/opt/EnbedSky/c gcc-3.4.1-glibc-2.3.3/arm-linux/includewi ools_3.4.1_softfloat/arm-linux/gcc-3.4.1-gli nable-threads=posixenable-symvers=gnue s=c,c++enable-sharedenable-c99enabl Thread model: posix	cross compiler at version 3.4.1 3.4.1_softfloat/arm linux/gcc-3.4.1-glibc-2 host=i686-host_pc-linux-gnupre farm linux/gcc-3.4.1-glibc-2.3.3w crosstools_3.4.1_softfloat/arm linux, th-local-prefix=/opt/EnbedSky/cross bc-2.3.3/arm linuxdisable-nlso nablecxa_atexitenable-language	2 f / t
and the second	[root@EnbedSky root]# gedit /etc/profile [root@EnbedSky root]# source /etc/profile [root@EnbedSky root]# arm-linux-gcc -v Reading specs from /usr/local/arm/3.3.2/lib/ Configured with:/gcc-3.3.2/configure ta 100prefix=/usr/local/arm/3.3.2 i686-pc-li h3900/includeenable-threads=pthreadsen -languages=c,c++ Thread model: posix gcc version 3.3.2	compiler at version 3.3.2 gcc-lib/arm-linux/3.3.2/specs rget=arm-linuxwith-cpu=strongarm nux-gnuwith-headers=/work/kernel	



(caution: After update, you might find the version is not changed in current terminal. You can open a new terminal to check the revised version.)

2.5.2 Network File Service (NFS) configuration

When installing RedHat 9.0, if you choose complete installation, all the relevant components will be installed by default. Configure the Network File Service as the following diagram:

(caution: Close the firewall of Linux as following diagram, otherwise the NFS can't work possibly)

Use the following command to close firewall:

#/etc/init.d/iptables stop

Configure the sharing directory:

Execute the command "#gedit /etc/exports", edit the configuration files of NFS (caution: The file is empty in the first opening), and add the following contents:

	✓ /etc/exports - gedit					/// - 🗆 x
-	文件(<u>F</u>) 编辑(<u>E</u>)	查看(<u>V</u>) 搜索(<u>S</u>) 工具(<u>T</u>)	文档(<u>D</u>)	帮助(<u>H</u>)	
	→ □ → □ → → □ →		~ -	[] []	公 登 查找 替换	
	exports ×					
	/opt/EmbedSky/roo	t_nfs	*(rw,syn	c,no_root_	squash)	
				行 1,	列 59	插入 🥢

- "/opt/EmbedSky/root_nfs" represents NFS sharing directory, it can be used as root file system of platform and be mounted by NFS:

- "*" represents that all the customers have the right to mount this directory.

- "rw" represents that the customer which mounting this directory is authorized to write and read this directory.

- "no_root_squash" represents that the customer who mount this directory could be treated as the root of server.

Build the sharing directory:

Find the file "root_nfs.tar.bz2" under the directory "Linux" in CD, and decompress it under the directory "/opt/EmbedSky/" in linux. The decompression command is "#tar xvfj root_nfs.tar.bz2 -C /".

Start and stop NFS:

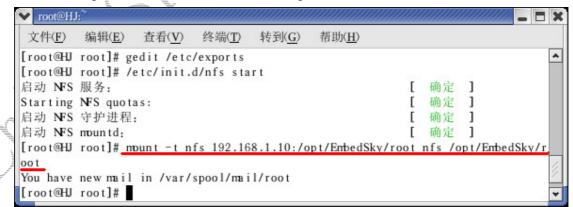
Execute the command "#/etc/init.d/nfs start" to start NFS:





root@HJ:~			_ 🗆 🗙
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转	到(<u>G</u>) 帮助(<u>H</u>)		
root@HJ root]# gedit /etc/exports root@HJ root]# /etc/init.d/nfs start 計动 NFS 服务: tarting NFS quotas: 計动 NFS 守护进程: 計动 NFS mountd: root@HJ root]#		[确定 [确定 [确定	

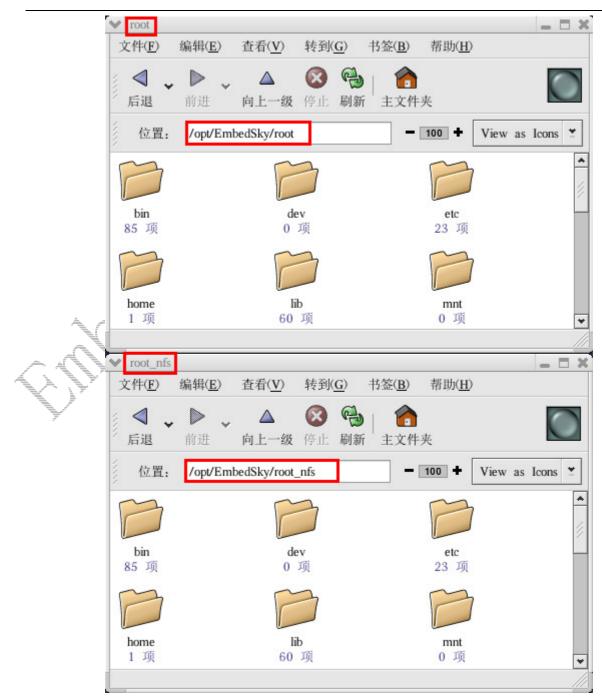
Use the following command to check the NFS running or not:



The contents under the directory "/opt/EmbedSky/root/" and "/opt/EmbedSky/root_nfs/" are the same. Operation on any one of the two will lead to change of the other. (caution: the directory "root" and "root_nfs" needs to be built by the user yourself. Use the command "mkdir xxx, xxx" to build a directory). The upper diagram indicates the user needs to input IP address of Linux on PC when mounting NFS.







Use the command "#/etc/init.d/nfs stop" to stop NFS.

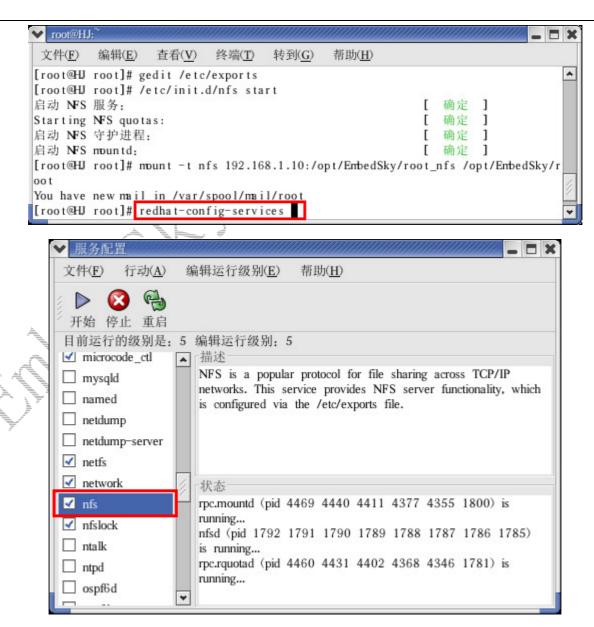
1

You can make NFS auto-start when PC starts up by following operation:

Use the command "#redhat-config-services" to open system service configuration window:







Find and select NFS option box:





文件(<u>F</u>) 行动(<u>A</u>)) 编辑运行级别(<u>E</u>) 帮助(<u>H</u>)
 一 一 一 一 一 一 一 一 一 二 二	3
目前运行的级别是 ✓ microcode_ctl	上:5 编辑运行级别:5 ▲ 描述
mysqld named netdump	NFS is a popular protocol for file sharing across TCP/IP networks. This service provides NFS server functionality, wh is configured via the /etc/exports file.
netdump-server	
 netdump-server netfs network 	
✓ netfs	↓ 「状态- rpc.mountd (pid 4469 4440 4411 4377 4355 1800) is
✓ netfs✓ network	≶ 「状态

Click "保存改变" in menu "文件(F)" to save configuration.

2. 5. 3 PC Linux FTP service configuration

As similar as NFS configuration, use the command "#redhat-config-services" to open system service configuration window and find the option "vsftpd". Select it and save the configuration:





文件(E) 行动(A)	编辑运行级别(<u>E</u>) 帮助(<u>H</u>)
▶ 🐼 <table-cell> 开始 停止 重启</table-cell>	
目前运行的级别是:	
🗌 time-udp	描述
🗆 tux	Vsftpd is a ftp daemon, which is the program that answers incoming ftp service requests.
🗆 ups	B of active reducers
vmware-tools	
vncserver	
✓ vsftpd	
winbind	状态
✓ xinetd	vsftpd (pid 1810) is running
yppasswdd	
□ ypserv	
ypxfrd	
🗌 zebra	

2. 5. 4 PC Linux Telnet service configuration

As similar as NFS configuration, use the command "#redhat-config-services" to open system service configuration window and find the option "telnet". Select it and save the configuration:





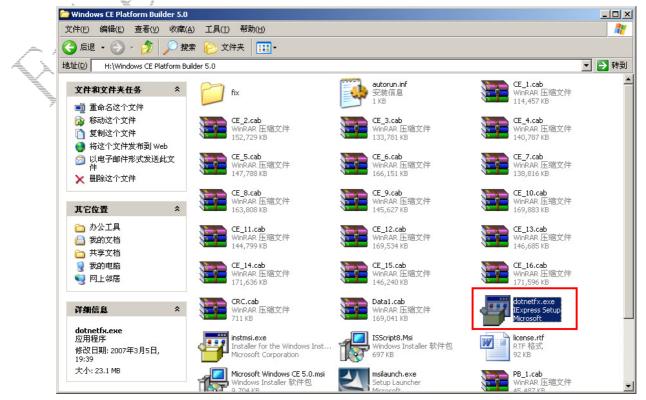


2. 6 WinCE environment configuration

This section introduces the process installing Platform Build 5.0 (call it PB for short) in WindowsXP. We use the PB for developing and configuring WinCE kernel and debugging. A space larger than 5GB is needed by all PB files. In addition, an ordinary project needs about 500MB space. Therefore, it is suggested to provide a space no less than 7GB for PB installation.

Caution: Copy the PB setup file to hard disk before installation. Otherwise the installing might fail.

Step 1: Open the WindowsCE CD, and install Framework net 1.0 first. Find "dotnetfx.exe" and double-click it to begin installation:

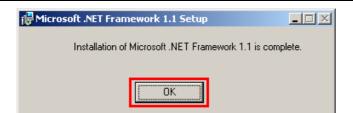


The following boxes pop up, keep clicking "是(Y)" button until the Framework net 1.0 installation finished.







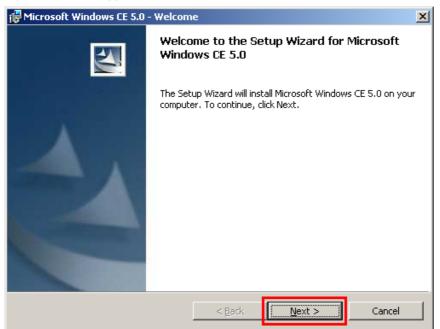


Step 2: Double-click "Setup.exe". The following interface appears. Click "Install" to continue:

🐞 Welcome to Microsoft Windows CE 5.0	
Start	
 → Install → Release Notes → What's New? 	
Microsoft	

Step 3: The Welcome interface appears, click "Next" to continue:

, di



Step 4: The interface "License Agreement" appears, select the option "I accept the terms in the license agreement" and click "Next" to continue:





🖟 Microsoft Windows CE 5.0 - License Agreement 🛛 🔀				
License Agreement Please read the following license agreement carefully.				
END-USER LICENSE AGREEMENT FOR				
MICROSOFT SOFTWARE				
MICROSOFT WINDOWS CE 5.0 TOOLKIT				
These terms are an agreement between Microsoft Corporation (or, if applicable because you distribute run-time licenses, one of Microsoft's affiliates) and you. Please read them. They apply to the software named above ("SOFTWARE") which includes the media on which you received it, if any. The terms also				
I accept the terms in the license agreement I I do not accept the terms in the license agreement (Exit Setup) InstallShield				
Print < Back Next > Cancel				

Step 5: The interface "Customer Information", enter the correct product key and click "Next" to continue:

🚰 Microsoft Windows CE 5.0 - Customer Information	×
Customer Information Please enter your information.	
User Name: Arthur Organization: 广州天嵌计算机科技有限公司 Please enter the product key:	-1
InstallShield	Cancel

Step 6: Select a setup type. Select "Custom (Tools and OS)" and click "Next" to continue:





17 N	1icrosoft Wind	ows CE 5.0 - Setup Type	×
5	etup Type Choose the seti	up type that best suits your needs.	
	Please select a		
	C Tools Only	Installs IDE and debugger support, but not CPU support for building images.	
Inst	allShield	2 < <u>B</u> ack <u>N</u> ext > Cancel	

Step 7: Select the installation directory. The installation path here is F disk (The default directory is recommended). Click "Next" to continue: (The following two diagrams give two choices: choosing default C disk or choosing user-defined directory F disk)

Hicrosoft Windows CE 5.0 - Destination Folders	×
Destination Folders Click Next to install to these folders, or click Change to install to a different folder.	
Install Platform Builder 5.0 to: C:\Program Files\Windows CE Platform Builder\5.00\	<u>C</u> hange
Install Microsoft Windows CE 5.0 Operating System to: C:\WINCE500\	Change
InstallShield	Cancel



🙀 Microsof	t Windows CE 5.0 - Destination Folders	×
	on Folders It to install to these folders, or click Change to install to a different	
	Install Platform Builder 5.0 to: F:\Program Files\Windows CE Platform Builder\5.00\	Change
	Install Microsoft Windows CE 5.0 Operating System to: F:\WINCE500\	Change
InstallShield -	< <u>B</u> ack <u>N</u> ext >	Cancel

Step 8: The option "Shared Source for Windows CE 5.0" in the following diagram doesn't need to be selected. It is selected here for some reason of screen capture operation; When customizing your system platform, if you are a user of S3C2440, please select "ARMV4I" and continue:

🙀 Microsoft Window	s CE 5.0 - Custo	m Setup		×
Custom Setup Select the program f	eatures you want	installed.		
	m Builder 1 Source for Windo ws CE 5.0 Test Kit ws CE 5.0 Operati Emulator	WS CE 5.0	M 72X, ARMS ARM102X, core: Motorola Dragor NeoMagic, NMS S3C24XX, Texa: OMAP	92X, and s, Intel XScale, nBall i.MX, 72XX, Samsung s Instruments uires 1888MB on
Install to: F:\WINCE500\ InstallShield			3	
<u>H</u> elp	Space	< <u>B</u> ack	<u>N</u> ext >	Cancel

Step 9: confirm the installation. Click "Install" to continue:

_



Step

Í



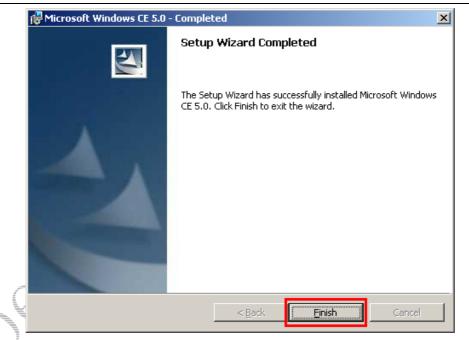
17	Microsoft Windows CE 5.0 - Ready to Install
	Ready to Install the Program The wizard is ready to begin installation.
	Click Install to begin the installation.
	If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.
-	
	< <u>B</u> ack Install Cancel
tallati	ion starts:
	Microsoft Windows CE 5.0 - Installing

Installing Microsoft Windows CE 5.0 - Installing Installing Microsoft Windows CE 5.0 The program features you selected are being installed. Will Please wait while the Setup Wizard installs Microsoft Windows CE 5.0. This may take several minutes to several hours depending on the components selected and whether this is a local or network installation. Status: Validating install InstallShield InstallShield Cancel

Step 11: About 20 minutes later, PB installation complete. Click "Finish" to continue:







The process of PB installation is complete.





Chapter 3 Platform Utilization

Linux OS and Qt graphic interface (the buring files are u-boot_T35.bin, zImage_T35.bin and root_qt_tp.yaffs under the directory of "Images->Linux") has been installed in Development Board by default. We can also change the operating system into WinCE according to your request.

3.1 Introduction of wire connection on platform and PC

Please connect the wire following the steps introduced in this section

> The jumpers: According to the default connection;

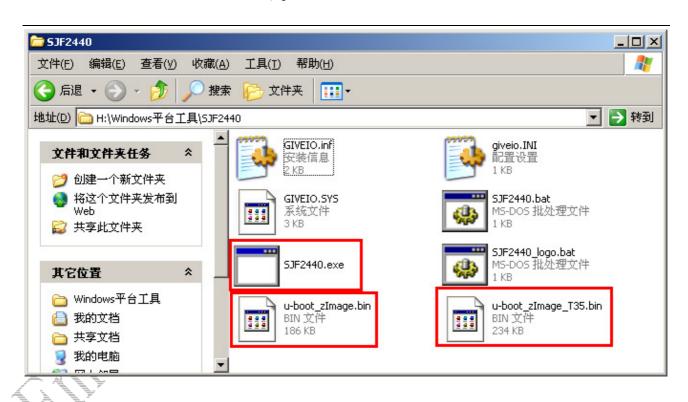
- > Connect 5y power adaptor to power interface in platform (power adaptor output should be less than 7V);
 - > Audio interface: Green interface for output and the pink one for MIC input;
 - Connect platform and PC via USB_Device interface
 - Connect COM1 of platform and PC serial port with direct-connect serial port wire;
 - Connect 100M network card of platform and PC with net line;
 - Connect Jtag interface of platform and PC parallel port with Jtag download board;
 - Connect camer module to camera interface of platform;
 - Connect LCD module to LCD interface of platform with FFC (Flexible Flat Cable);

3.2 Burning u-boot by SJF2440

The software SJF2440 is under the directory "SJF2440" of "Windows 平台工具" in CD-ROM. SJF2440.exe is used for burning software; u-boot_zImage.bin and u-boot_zImage_T35.bin are u-boot image files; SJF2440.bat and SJF2440_logo.bat are batch processing files for burning:

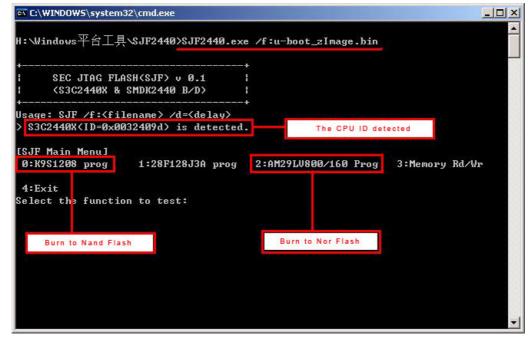






3.2.1 Executing SJF2440 with batch-processing files

Double-click "SJF2440.bat" in the upper diagram:



3.2.2 Running SJF2440

Í

Step1, click "运行" in menu "开始":





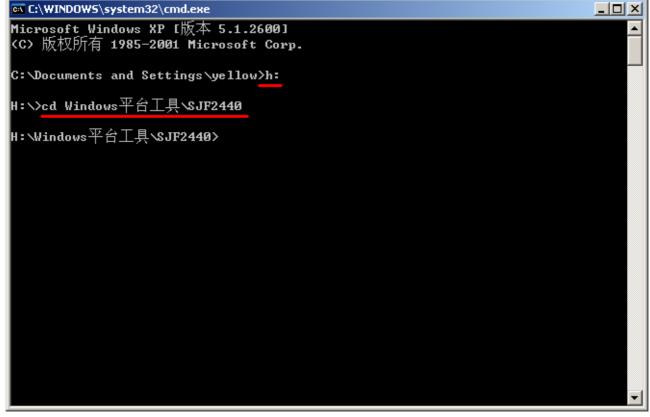
ssion	Þ	搜索(<u>C</u>)
Profe	?	帮助和支持(H)
9	1	运行(<u>R</u>)
swopu	\triangleright	注销 yellow(<u>L</u>)
Wir	0	关闭计 算机(U)
	开始	🕑 💽 🤏 » [
	Windows	Mindows YD

Step2, enter "cmd" and click "确定" to continue:

1

A 7					
	运行				? ×
		请键入程序、 称,Windows	^文 佳来。文礼 1	当或 Internet 它。	资源的名
	打开①	l: emd			•
AAN		2	2		
			确定	取消し	刘览(8)

Step3, execute DOS command to enter into SJF2440 directory shown in the following diagram:



Step4, run SJF2440. execute the following command: SJF2440.exe /f:u-boot_zImage_T35.bin (instruction: /f: dosen't mean disc but file), as shown in the following diagram:如下图所示:





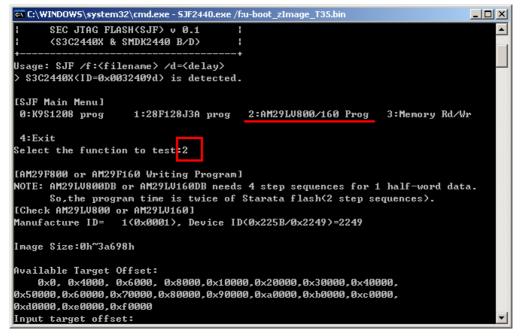
C:\WINDOWS\system32\cmd.exe - SJF2440.exe /f:	u-boot_zImage_T35.bin	<u>_0×</u>
Microsoft Windows XP [版本 5.1.2600] <c>版权所有 1985-2001 Microsoft Corp. C:\Documents and Settings\ye:<mark>the</mark> burning H:\>cd Windows平台工具\SJF24.c</c>	the file name	
H:Windows平台工具\\$JF2440 <mark>}\$JF2440.exe</mark>	/f:u-boot_zImage_T35.]	bin
+		
+	"f.\" means the file to be burned	
[SJF Main Menu] Ø:K9S1208 prog 1:28F128J3A prog	2:AM29LV800/160 Prog	3∶Memory Rd∕Wr
4:Exit Select the function to test:		
		•

3.2.3 Burning u-boot to Nor Flash

The burning process:

Í

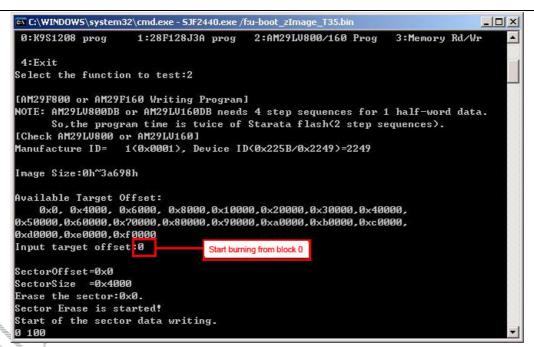
Step1, in previous diagrams of "3.2.1 节", select "2" and press return key to continue:



Step2, enter "0". Start burning from block 0 and press return key to continue:







Step3, when burning process is finished, it exits automatically or the following diagram appears:

```
📾 C:\WINDOWS\system32\cmd.exe
                                                                             - 🗆 ×
0 d400 d500 d600 d700 d800 d900 da00 db00 dc00 dd00 de00 df00 e000 e100 e200 e30 🔺
 e400 e500 e600 e700 e800 e900 ea00 eb00 ec00 ed00 ee00 ef00 f000 f100 f200 f30
0 f400 f500 f600 f700 f800 f900 fa00 fb00 fc00 fd00 fe00 ff00
End of the sector data writing !!!
SectorOffset=0x30000
SectorSize =0x10000
Erase the sector:0x30000.
Sector Erase is started!
Start of the sector data writing.
0 100 200 300 400 500 600 700 800 900 a00 b00 c00 d00 e00 f00 1000 1100 1200 130
  1400 1500 1600 1700 1800 1900 1a00 1b00 1c00 1d00 1e00 1f00 2000 2100 2200 230
 2400 2500 2600 2700 2800 2900 2a00 2b00 2c00 2d00 2e00 2f00
                                                              3000 3100
                                                                        3200 330
  3400 3500 3600 3700 3800 3900 3a00 3b00 3c00 3d00 3e00 3f00 4000 4100 4200 430
  4400 4500 4600 4700 4800
                          4900 4a00 4b00 4c00
                                               4d00 4e00
                                                         4f 00
                                                              5000 5100
                                                                        5200
                                                                             530
  5400 5500 5600 5700 5800 5900 5a00 5b00 5c00 5d00 5e00 5f00 6000 6100 6200 630
  6400 6500 6600 6700 6800 6900 6a00 6b00 6c00 6d00 6e00 6f00 7000 7100 7200 730
  7400 7500 7600 7700 7800 7900 7a00 7b00 7c00
                                               7d00
                                                    7e00 7f00 8000 8100 8200 830
  8400 8500 8600 8700 8800 8900 8a00 8b00 8c00 8d00 8e00 8f00 9000 9100 9200 930
  9400 9500 9600 9700 9800 9900 9a00 9b00 9c00 9d00 9e00 9f00 a000 a100
                                                                        a200 a30
 a400 a500 a600
End of the sector data writing?!?
H:\Windows平台工具\$JF2440>
```

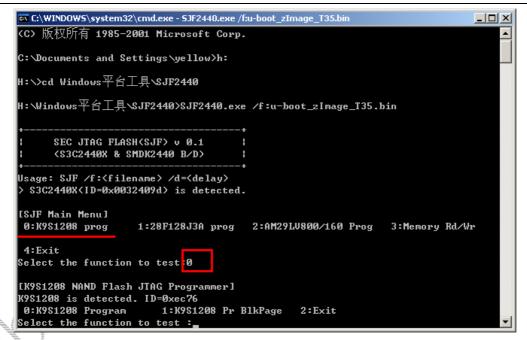
3. 2. 4 Burning u-boot to Nand Flash

The burning process:

Step1、 in previous diagrams of "3.2.1 节", select "0" and press return key to continue:







Step2, enter "0" to select Flash type and press return key to continue:

🔤 C:\WINDOWS\system32\cmd.exe - SJF2440.exe /f:u-boot_zImage_T35.bin	
	-
++ SEC JTAG FLASH(SJF) v 0.1	
(\$3C2440X & SMDK2440 B/D)	
tt	
Usage: SJF /f: <filename> /d=<delay></delay></filename>	
> S3C2440X(ID=0x0032409d) is detected.	
[SJF Main Menu]	
0:K9S1208 prog 1:28F128J3A prog 2:AM29LV800/160 Prog 3:Memory Rd/k	ha
	Ê.
4:Exit	
Select the function to test:0	
[K9S1208 NAND Flash JTAG Programmer] K9S1208 is detected. ID=0xec76	
0:K9S1208 Program 1:K9S1208 Pr BlkPage 2:Exit	
Select the function to test (0	
[SMC(K9S1208V0M) NAND Flash Writing Program]	
Source size:0h~3a697h	
Available target block number: 0~4095	
Input target block number: 0 4075	-

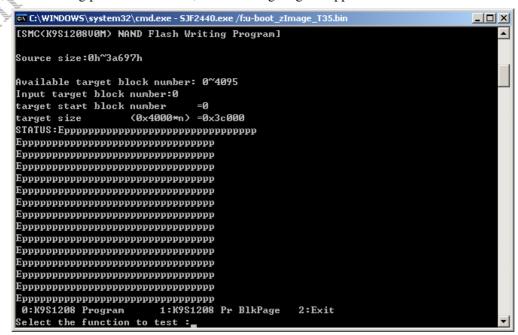
Step3, enter "0". Start burning from block 0 and press return key to continue:





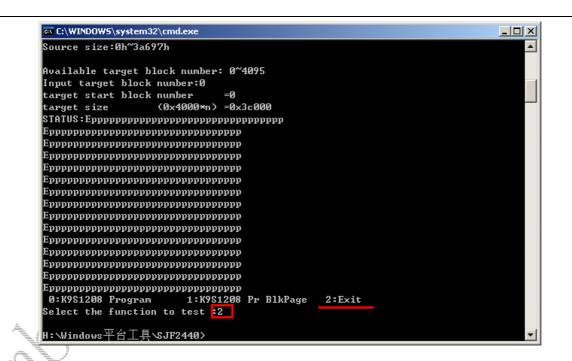
🗠 C:\WINDOWS\system32\cmd.exe - SJF2440.exe /f:u-boot_zImage_T35.bin
(\$3C2440X & \$MDK2440 B/D)
+
/ SJ62440A(1D-0X00J240/u/ 15 uctetteu.
[SJF Main Menu]
0:K9S1208 prog 1:28F128J3A prog 2:AM29LV800/160 Prog 3:Memory Rd/Wr
4:Exit
Select the function to test:0
[K9S1208 NAND Flash JTAG Programmer]
K9S1208 is detected. ID=0xec76
0:K9S1208 Program 1:K9S1208 Pr BlkPage 2:Exit
Select the function to test :0
[SMC(K9S1208V0M) NAND Flash Writing Program]
Source size:0h~3a697h
Available target block number: 0~4095
Input target block number:0 Start burning from block 0
target start block number =0
target size 〈0x4000*n〉=0x3c000 STATUS:Ep

Step4, when burning process is finished, the following diagram appears:



Select "0" to continue or "2" to exit. Here we select "2":





3.3 Burning u-boot with H-Jtag

3. 3. 1 H-JTAG installation

Decompress and install "H-JTAG V0.4.3.zip" under the directory "H-JTAG" of "Windows 平台工具" in CD-ROM; Or download other versions from the website "<u>http://www.hjtag.com/download.html</u>". The following instructions are corresponding to the version "H-JTAG V0.4.3.zip" in CD-ROM.

Step1, double-click "H-JTAG V0.4.3.zip" to decompress it. If you have no decompression tool, please install it first;

Step2, H-JTAG V0.4.3.EXE appears after decompressing, double-click it to enter into installation guide. Keep clicking "next" in the following pop-up interfaces.

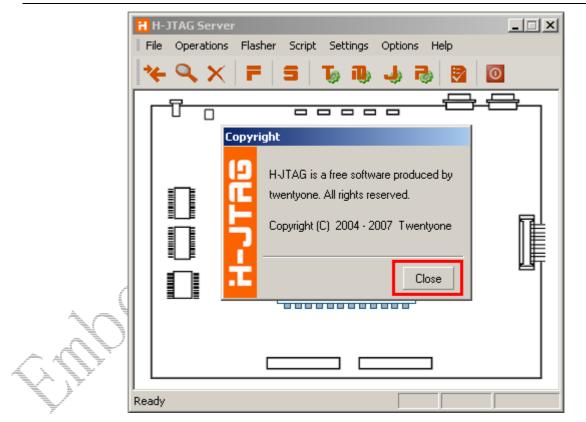
Step3, after installation, the shortcut "H-JTAG" and "H-Flasher" will appear in desktop.

3. 3. 2 H-JTAG configuration

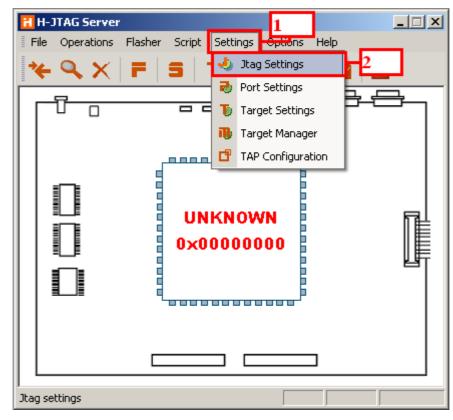
The following diagrams illustrate the configuration steps: Step1, double-click the icon "H-JTAG" as the following diagram:







Step2, click "close". The following interface appearing, click "Jtag Settings" in menu "Settings (设置)":



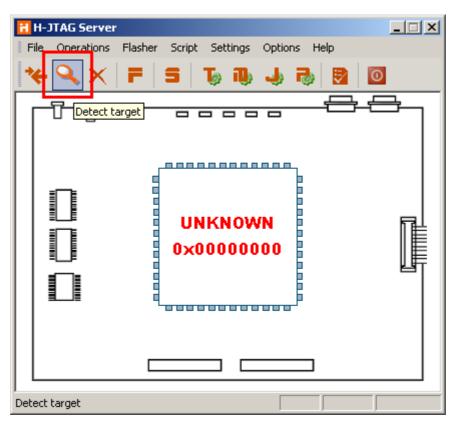
Configure the following interface "Jtag Settings":



Jtag Settings		×
Jtag Selection	User Del	fined Pin Assignment –
C Wiggler (Predefined)	TMS	Pin4 D2 💌
C Sdt Jtag (Predefined)	тск	Pin2 D0 💌
User Defined 1	TDI	Pin3 D1 💌
Reset Signal Output	TDO	Pin11 Busy 💌
nTRST output inver	nTRST	NO TAP RST 💌
nSRST output inverted	nSRST	NO SYS RST 💌
ОК	Cance	el

Step3, click "OK" to finish configuration. Check all the connections in Development Board and turn on the

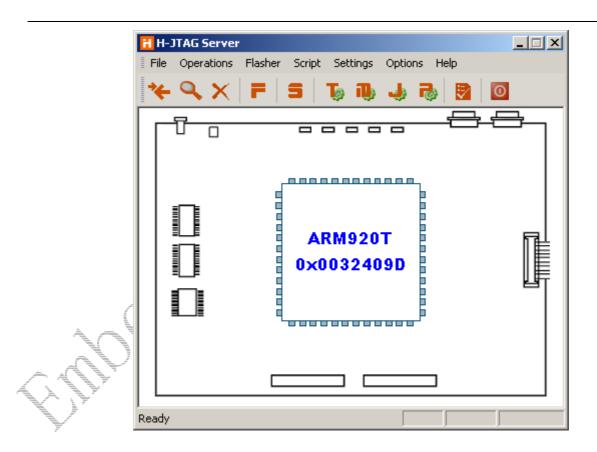
power. Click "Detect target" (_____) or click "Detect target" in menu "Operations":



Step4, if CPU has been detected, the following diagram appears. Or prompting error if failed.







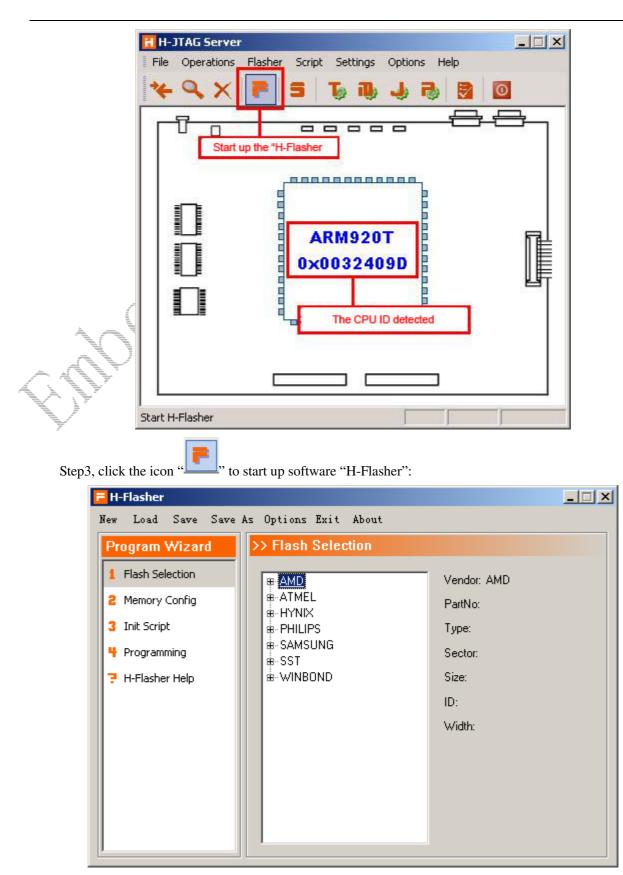
3. 3. 3 H-Flasher configuration

Caution: The software "H-Jtag" can only burn u-boot into Nor Flash.

Step1, connect Jtag wire and remove the jumper on OM0. Then turn on the power; Step2, start up "H-Jtag". The software begins to dectet CPU automatically:







Step4, select "1 Flash Selection" and click "AMD" on the right:

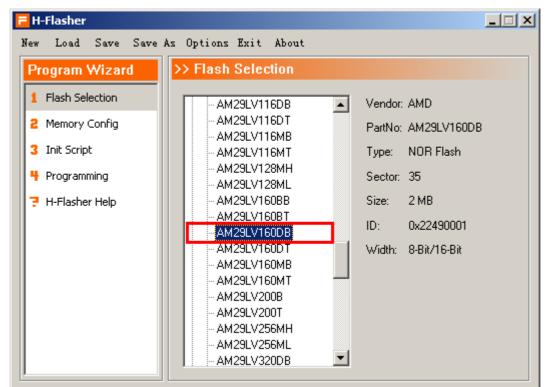




Program Wizard	>> Flash Selection	
1 Flash Selection		
A Hash bolocdon	P AMD 2	Vendor: AMD
2 Memory Config	- AM29F002B	PartNo:
3 Init Script	AM29F004BB	Туре:
4 Programming	AM29F004BT AM29F010	Sector:
P H-Flasher Help	- AM29F010B	Size:
	- AM29F016	ID:
	AM29F016B AM29F016D	Width:
	- AM29F018D	wian:
	- AM29F032	
	- AM29F032B	
	- AM29F040	
	- AM29F040B	
	AM29F080	
	- AM29F080B	-

Find "AM29LV160DB" downwards and click it:

Í



Step5, select "2 Memory Config" and configure the options rightward. As shown in the following diagram:





EH-Flasher New Load Save Save A	As Options Exit About	
Program Wizard	>> Memory Config	
1 Flash Selection		
2 Memory Config	Flash Bit Width: 16-Bit Mode	[
3 Init Script 4 Programming	Flash Start Address: 0x0	2
H-Flasher Help	RAM Start Address: 0x30000000	

Step6, click "3 Init Script" leftward and click the adding button "*" as the following diagram:

Program Wizard	-	t Script				
Flash Selection	Idx	Cmd	Width	Address	Value	
Memory Config	1	Setmem	16-Bit	0x0	0	
Init Script		3	1-			
Programming		<u>~</u>	-		_ Move _ Upward	- •
H-Flasher Help				_	Add	
		2		1	Delete	
	-					
		(Move Downward	
			1			

Step7, click "4 Programming". The following interface appears:





🔁 H-Flasher		
New Load Save Save	As Options Exit About	
Program Wizard	>> Programming - AM29LV160DB	
1 Flash Selection	Flash: Unchecked	Check
2 Memory Config	Target: Unchecked	
3 Init Script		
4 Programming	Type: Auto Flash Download	• Program
P H-Flasher Help	Src File:	
	Dst Addr:	-
	From: Entire Chip	- Erase
	To: Entire Chip	• Blank

Make configuration as the following diagram:

🔁 H-Flasher				
New Load Save Save As Options Exit About				
Program Wizard	>> Programming - AM29LV160DB			
1 Flash Selection	Flash: Unchecked	Check		
2 Memory Config	Target: Unchecked			
3 Init Script				
4 Programming	Type: Plain Binary Format	Program		
P H-Flasher Help	Src File: H:\Images\u-boot_zImage_T35.bin .			
	Dst Add: 0x0			
	From: Entire Chip	Erase		
	To: Entire Chip	Blank		

Click " ":





New Load Save Save	As Options	: Exit About	
Program Wizard	>> Prog	ramming - AM29LV160DB	
1 Flash Selection	Flash:	Unchecked	Check
2 Memory Config	Target:	Unchecked	
3 Init Script			
4 Programming	Туре:	Plain Binary Format	Program
7 H-Flasher Help	Src File:	H:\Images\u-boot_zImage_T35.bin	
	Dst Addr:	0x0	
	From:	Entire Chip	Erase
	To:	Entire Chip	Blank

Locate u-boot:

Í

打开				<u>? ×</u>
查找范围(I): 🥫	🔁 Images	•	← 🗈	r 🖽
DNW PIC SKY2440_Test u-boot_zImage zImage-2.6_C4	_T35.bin	1		2
文件名 (M): 🛛 🖬	-boot_zImage_T35.bin			打开 (0)
文件类型 (重): B	Sinary Files (*.bin)			取消

Click "Check" rightward. You can find the CPU ID and Nor Flash in the window:





New Load Save Save	s Options Exit About	
Program Wizard	>> Programming - AM29LV16	DB
1 Flash Selection 2 Memory Config	Flash: AM29LV160DB 0x2249000 Target: ARM920T Little-Endian	01 Check
3 Init Script		
4 Programming	Type: Plain Binary Format	▼ Program
P-Flasher Help	Src File: H:\Images\u-boot_zImage Dst Addr: 0x0	_T35.bin
	From: Entire Chip	▼ Erase
	To: Entire Chip	▼ Blank

Step8, burning u-boot. Click "Program" to begin burning:

H-Flash	er	
	u-boot_zImage_T35.bin	
	00:01:10 7% 17 KB/s	Size = 233.6 KB
		Stop

Click "Close" to finish burning.

H-Flash	ner	
	Programmed and verified successfully	
	00:13:70 100% 18 KB/s	Size = 233.6 KB
		Close

Step9, click "save" and there is no need to configure H-Flasher again next time.



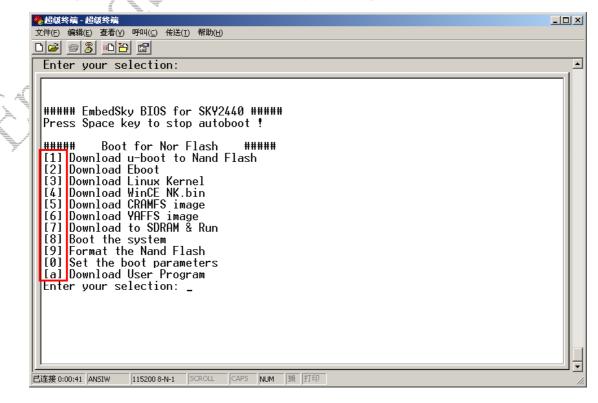
3.4 u-boot application

u-boot in TQ2440 supports 2 start-up modes, starting from Nor Flash or Nand Flash, and also supports 2 file download modes, download Linux and WinCE by using USB or TFTP.

The following contents illustrate how to use one-key operation menu. One-key operation menu adopts USB download mode as default. If you have selected TFTP download mode, please exit one-key operation menu and enter into the default operation interface of u-boot.

3.4.1 Function introduction

The start-up interface is shown in the following diagram. The operation code is located inside the red frame:



3.4.2 burning u-boot

Please consult "2.3 节" and "2.4 节"

3. 4. 3 Configuration and estimation of u-boot start-up state

There are 3 states to select when TQ2440 starts up:

- State 1: Download mode. Update u-boot, Linux and WinCE in this mode;
- State 2: Linux start-up mode. Load and boot Linux in this mode;



State 3: WinCE start-up mode. Load and boot WinCE in this mode.

Entering into state 1: Connect serial port of platform, press and hold space-key of PC keyboard, and then switch on platform power. After that, "Donwload System mode" appears on LCD;

Entering into state 2 and state 3: Switch on power directly. System starts boot Linux or WinCE based on the software installed in platform. The phase "Linux System Loading ..." (state 2) or "WinCE System Loading ..." (state 3) appears on LCD soon.

3. 4. 4 Introduction of u-boot utilization

There are 11 kinds of operations when TQ2440 u-boot starts up.

Caution: The following instroduction about TQ2440 u-boot utilization is arranged in accord with the menu sequence. Please consult "chapter 3.7" when burning Linux or WinCE.

The way entering into u-boot console: Start Hyper Terminal, press and hold space-key of PC keyboard, and switch on platform power.

Caution: eboot and Linux kernel partly overlap. Please burn eboot first if you try to change Linux into WinCE.

The files referred in the following contents are included under the directory of "Images\Linux" or "Images\WinCE" in CD-ROM

Step 1, enter "1": Download u-boot to Nand Flash, as shown in the following diagram:

(caution: This operation is corresponding to reading and writing Nand Flash, if u-boot works well, it is not suggested to re-burn u-boot.)

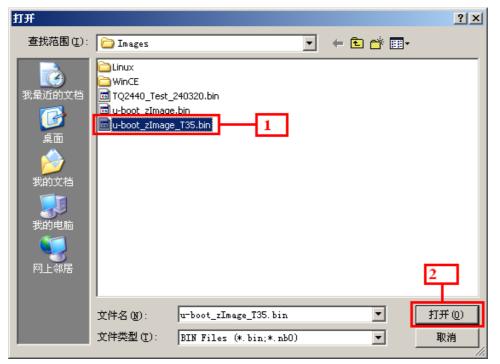
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	- D X
Image: Second State Image: Second State ###### EmbedSky BIOS for SKY2440 ###### Press Space key to stop autoboot ! ###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Linux Kernel [4] Download WinCE NK.bin [5] Download VAFFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [1] Download User Program Enter your selection: 1 USB host is connected. Waiting a download.	
已连接 0:00:13 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印 86	li.



Open the software DNW when the upper sentence "USB hosting is connected, Waiting a download." appears. Click "Transmit" in "USB Port" menu:

Serial Port USB Port Configu	_O×
Transmit 2 Rx Test	*
Status	
	_

Locate u-boot in the following interface. Find "u-boot_zImage_xxxx.bin" under the directory of "Image\Linux" or "Image\WinCE" in CD-ROM (xxxx represents LCD type, T35 is Toshiba 3.5 inch LCD, S35 is Sumsung 3.5 inch LCD and S70 is Sumsung 7 inch LCD). Click "打开" to continue:



u-boot is downloaded and saved automatically:

Í





文件(E)编辑(E)查看(Y)呼叫(C)传送(I)帮助(H) □ 2 2 2 3 ■ 1 2 1 2 1 2 1 2 2 1 2 2 2 2 2 2 2 2 2	
RECEIVED FILE SIZE: 239266 (233KB/S, 1S)	
NAND erase: device 0 offset 0x0, size 0x40000 Erasing at 0x3c000 100% complete. OK	
NAND write: device 0 offset 0x0, size 0x3a698	
Writing data at 0x3a600 100% complete. 239256 bytes written: OK	
##### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image	
 [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters 	
[a] Download User Program Enter your selection:	

Step 2, enter "2": Download eboot, as shown in the following diagram:

後超级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	<u>-0×</u>
NAND erase: device 0 offset 0x0, size 0x40000 Erasing at 0x3c000 100% complete. OK	
NAND write: device 0 offset 0x0, size 0x3a698	
Writing data at 0x3a600 100% complete. 239256 bytes written: OK	
###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [3] Download Linux Kernel	
<pre>[4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system</pre>	
[9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 2 USB host is connected. Waiting a download.	
CI连接 0:04:03 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	

Open the software DNW when the upper sentence "USB hosting is connected, Waiting a download." appears. Click "Transmit" in "USB Port" menu:





Serial Port USB P		
	Test 4	
÷		

Locate "EBOOT_dm9000_u-boot.nb0" under the directory of "Images\WinCE" in CD-ROM and click "打 开" to continue:

打开							<u>? x</u>
查找范围(I):	C WinCE		_	•	(÷ 🗈 🗎	× 💷 -	
表し近的文档 (で) 泉面 (で) 泉面 (で) 泉面 (で) 泉面 (で) 泉面 (で) 泉面 (で) 泉面 (で) (で) 泉面 (で) (で) (で) (で) (で) (で) (で) (で)	EBOOT_dm900	5_5.0.bin bin	1				2
	文件名 (M):	EB00T_dm9000	_u-boot. nb0		-		打开(0)
	文件类型 (<u>T</u>):	BIN Files (*	. bin;*. nb0)		-	3	取消

eboot is downloaded and saved automatically:





New, Downloading IADDRESS:30000000h,TOTAL:901221 RECEIVED FILE SIZE: 90122 (88KB/S, 1S) NAND erase: device 0 offset 0x50000, size 0x20000 Erasing at 0x6c000 100% complete. OK NNND write: device 0 offset 0x50000, size 0x16000 90112 bytes written: 0K ##### 11 Download u-boot to Nand Flash 12 Download Eboot 13 Download CRMMFS image 17 Download to SDRAM & Run 18 Boot the system 19 Format the Nand Flash 101 Set the boot parameters 1a Download User Program Enter your selection: Step 3, enter "3": Download Linux kernel, as shown in the following diagram:		
Now, Downloading IADDRESS: 30000000h, TOTAL: 90122] RECELIVED FILE SIZE: 90122 (88KB/S, 1S) NAND erase: device 0 offset 0x50000, size 0x20000 Erasing at 0x6c000 100% complete. OK NAND write: device 0 offset 0x50000, size 0x16000 90112 bytes written: OK ###### Boot for Nor Flash ##### 11 Download u-boot to Nand Flash 121 Download Linux Kernel 141 Download WinCE NK.bin 131 Download CRAMFS image 161 Download OrFFS image 161 Download VGFFS image 171 Download to SDRAM & Run 181 Boot the system 191 Format the Nand Flash 101 Set the boot parameters 1a1 Download User Program Enter your selection:	文件(E) 編輯(E) 查看(V) 呼叫(C) 传送(I) 帮助(H)	
RECÉIVED FILE SĨZE: 90122 (88KB/S, 1S) NAND erase: device 0 offset 0x50000, size 0x20000 Erasing at 0x6c000 100% complete. OK NAND write: device 0 offset 0x50000, size 0x16000 90112 bytes written: OK ###### Boot for Nor Flash ##### [11 Download u-boot to Nand Flash [21 Download Linux Kernel [31 Download Linux Kernel [41 Download WinCE NK.bin [51 Download CRAMFS image [61 Download CRAMFS image [61 Download CRAMFS image [61 Download CRAMFS image [61 Download SDRAM & Run [81 Boot the system [91 Format the Nand Flash [01 Set the boot parameters [10] Set the boot parameters [11 Download User Program Enter your selection:		
[1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection:	RECÉIVED FILE SĬZE: 90122 (88KB/S, 1S) NAND erase: device 0 offset 0x50000, siz Erasing at 0x6c000 100% complete. OK NAND write: device 0 offset 0x50000, siz	e Øx20000
	<pre>[1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>	
Step 3, enter "3": Download Linux kernel, as shown in the following diagram:	この1:59 ANSIW 115200 8-N-1 SCROLL CAP5 NUM 捕	打印 //
	Step 3, enter "3": Download Linux kernel, as shown	n in the following diagram:
授超终端 - 超级终端 文件(E) 編辑(E) 查看(V) 呼叫(C) 传送(I) 帮助(出)		

文件(F)编辑(E) 查看(y) 呼叫(C) 传送(I) 帮助(H)	
<pre>###### EmbedSky BIOS for SKY2440 ###### Press Space key to stop autoboot ! ###### Boot for Nor Flash ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download WinCE NK.bin [5] Download WAFFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: [3] USB host is connected. Waiting a download.</pre>	
已连接 0:00:09 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	//

Open the software DNW when the upper sentence "USB hosting is connected, Waiting a download." appears. Click "Transmit" in "USB Port" menu:





Serial Port USB Port Configu	
Transmit 2	<u> </u>
Rx Test Status	
	-

Locate "zImage_2.6_xxxx_uboot" under the directory of "Image\Linux" in CD-ROM(xxxx represents LCD type, T35 is Toshiba 3.5 inch LCD, S35 is Sumsung 3.5 inch LCD, W35 is Donghua 3.5 inch LCD and S70 is Sumsung 7 inch LCD). Click "打开" to continue:

打开		? ×
查找范围(I):	: 🔁 Linux 💌 🗢 🗈 📸 🎫 -	
ましい文档 武最近的文档 重面 変か文档 変か文档 変の中間 同 一 一 一 一 一 一 一 一 一 一 一 一 一	readme.txt root_condence.img root_qt_mouse.img root_qt_tp.img u-boot_zImage_t35.bin zImage_2.6_T35_uboot 1	
		开 ([) [2]消

Linux kernel image is downloaded and saved automatically:





※超级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(E) □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	
□ ■ ■ ■ ■ ■ RECEIVED FILE SIZE: 1579298 (771KB/S, 2S) NAND erase: device 0 offset 0x4c000, size 0x200000 Erasing at 0x248000 100% complete. OK NAND write: device 0 offset 0x4c000, size 0x181918	
Writing data at 0x1cd800 100% complete. 1579288 bytes written: OK	
<pre>##### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters</pre>	
[a] Download User Program Enter your selection: 已建接 0:01:20 ANSIW 115200 8-N-1 SCROLL CAPS NUM 浦 打印	
Stand anter "4". Described Wir CE NK his as shown in the fallowing discusses	

Step 4, enter "4": Download WinCE NK.bin, as shown in the following diagram:

2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2	×
文件(E) 編辑(E) 查看(⊻) 呼叫(⊆) 传送(I) 帮助(出)	
###### Boot for Nor Flash ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: [4] Clear the free memory:	1
ok Read eboot image from flash Sector addr on NAND: 0x280 TotalSector: 0x100 LoadAddress: 0x30038000 JumpAddr: 0x30038000 Now, to download the wince image(nk.bin) USB host is connected. Waiting a download. 	

Open the software DNW when the upper window "USB hosting is connected, Waiting a download." appears, and click "Transmit" in the menu "USB Port"





Serial Port USB Port Configu 1	
Transmit 2 Rx Test Status	<u></u>

Locate "NK_xxxx_5.0.bin" under the directory of "Image\WinCE" in CD-ROM (xxxx represents LCD type, 240320_T35 is Toshiba 3.5 inch LCD, 320240_S35 is Sumsung 3.5 inch LCD, 320240_W35 is Donghua 3.5 inch LCD and 800480_S70 is Sumsung 7 inch LCD). Click "打开" to continue:

打开				<u>?</u> ×
查找范围(<u>I</u>):	🗁 WinCE	•	← 🗈 💣 🎟•	
ま 表 よ し の 文 档 で う 点 面 う で う さ さ 一 う 面 う で う む う で う れ し の 文 档 の 文 档 の う で う れ の う で う れ の う で う こ の う で う れ の う で う の う で う の う で う の う で う の う で う の う で う の う で う の う で う の う で う の う つ う の う つ う の う つ う つ う つ う つ う う つ う つ う う つ つ つ つ つ つ つ う つ つ つ つ つ つ つ つ つ つ つ つ つ	ERCOT_dm9000_u-boot.ob0 NK_240320_T35_5.0.bin u-boot_zImage.bin u-boot_zImage_T35.bin	1		2
	文件名 (M): NK_240320_T35_5.0. 文件类型 (I): BIN Files (*.bin;		•	打开 (0) 取消

NK.bin image is being transmitted to memory in the following diagram:



-



文件(E) 編辑(E) 査看(Y) 呼叫(C) 传送(T) 帮助(H) □ 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
Mark Block 13 as unused, has mark 14 blocks Mark Block 14 as unused, has mark 15 blocks Mark Block 15 as unused, has mark 16 blocks Mark Block 16 as unused, has mark 17 blocks Mark Block 17 as unused, has mark 18 blocks Mark Block 19 as unused, has mark 20 blocks Mark Block 20 as unused, has mark 20 blocks Mark Block 21 as unused, has mark 22 blocks Mark Block 22 as unused, has mark 23 blocks Mark Block 23 as unused, has mark 24 blocks Mark Block 25 as unused, has mark 26 blocks Mark Block 27 as unused, has mark 27 blocks Mark Block 27 as unused, has mark 28 blocks Extract wince image throught USB OEMMultiBINNotify: Download BIN file information: 	
OEMVerifyMemory: StartAddr: 0x8c200000, Length:0x172421c *** Downloading UNKNOWN image type *** 	
	•

- 超级终端 - 超级终端 2件(F) 編辑(E) 査看(V) 呼叫(C) 传送(T) 帮助(H)		
Mark Block 20 as unused, has mark 21 bloc Mark Block 21 as unused, has mark 22 bloc Mark Block 22 as unused, has mark 23 bloc Mark Block 23 as unused, has mark 24 bloc Mark Block 24 as unused, has mark 25 bloc Mark Block 25 as unused, has mark 26 bloc Mark Block 26 as unused, has mark 27 bloc Mark Block 27 as unused, has mark 28 bloc Extract wince image throught USB OEMMultiBINNotify: Download BIN file info	sks sks sks sks sks sks sks	
[0]: Base Address=0x8c200000 Length=0x17		
OEMVerifyMemory: StartAddr: 0x8c200000, L *** Downloading UNKNOWN image type *** dwImageStart : 0x8c200000 dwImageLength: 0x172421c LaunchAddr : 0x8c201000 rom_offset=0x0. Run eboot, JumpAddr = 0x30038000	Length: 0x172421c The cause of the failure is that eboot has not been burned. So we suggest burning eboot before burning WinCE, especially a Linux system has been burned in before, when it is not sure if the board has a eboot.	
┃ー 注接 0:14:06 ANSIW 115200 8-N-1 SCROLL CAPS NUM 1捕 1	「印 「印	

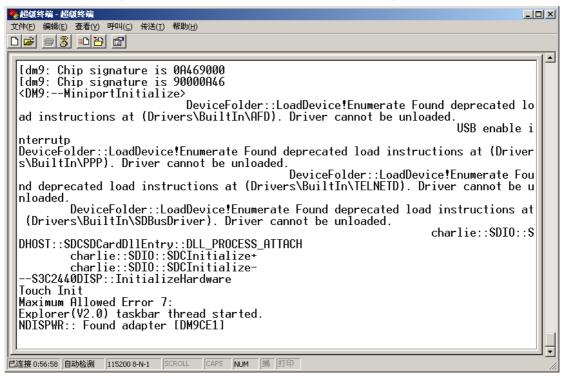
After "NK.bin" has been successfully transmitted, Low-level format begins. Finally it is formatted into BinFS file (Bad blocks might be detected during formatting. Bad blocks is inevitable when Nand Flash is been produced, but our softwares are robust enough to aviod the side-effect brought by bad blocks):





● 品級終端 □ <th></th>	
ID[1] { dwVersion: 0x1 dwSignature: 0x43465349 String: '' dwImageType: 0x6 dwTlSectors: 0x0 dwLoadHddress: 0x0 dwJumpAddress: 0x0 dwStoreOffset: 0x0 chainInfo.dwLoadAddress: 0X00000000 chainInfo.dwLoadAddress: 0X00000000 chainInfo.dwLength: 0X00000000 chainInfo.dwLength: 0X00000000 g_pViviWinceInfo = 0x301D8000, g_pViviWinceInfo->dwViviWinceMagic = 0x12345678 Low-level format nand flash Reserving Blocks [0x0 - 0x1B] reserve complete. Low-level format Blocks [0x1C - 0xFFF] Low-level format Blocks [0x1C - 0xFFF] When a bad block detected. ######### Error Erasing block 3646! erase complete Format nand flash for BinFS, please wait several minutes Misormated into BinFS.	
記注接 0:51:50 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

Burning WinCE is complete about 5 minutes later. Then WinCE starts up automatically:



Step 5, enter "5": Download Cramfs image (Cramfs is read-only which is not as convenient as Yaffs. So Cramfs is not recommended. The download process is the same as Yaffs)

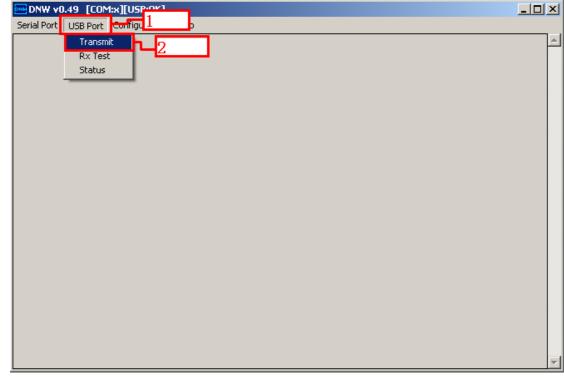
Step 6, enter "6": Download Yaffs image, as shown in the following diagram:





授級终端 - 超級终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H) □ 2 3 3 10 10 11 □ 3 10 10 □ 3 10 10 □ 3 10 10 □ 3 10 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 10 □ 3 □ 3 10 □ 3 □ □ 3 □ 3 □ □ 3 □ □ 3 □ □ 3 □ □ 3 □ □ 3 □ □ 3 □ □ □ 3 □	
NAND erase: device 0 offset 0x4c000, size 0x200000 Erasing at 0x248000 100% complete. OK	
NAND write: device 0 offset 0x4c000, size 0x181918	
Writing data at 0x1cd800 100% complete. 1579288 bytes written: OK	
##### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin	
[5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRHM & Kun	
[8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters	
[0] Set the boot parameters [a] Download User Program Enter your selection: 6	
USB host is connected. Waiting a download.	
	<u> </u>
已连接 0:02:00 ANSIW 115200 8-N-1 SCROLL CAPS NUM 描 打印	li.

Open the software DNW when the upper sentence "USB hosting is connected, Waiting a download." appears. Click "Transmit" in "USB Port" menu:



Locate "root_qt_tp.img" under the directory of "Image\Linux" in CD-ROM (root_condence.img is simplified file system; root_qt_mouse.img supports mouse and contains Qt file system; root_qt_tp.img supports touch-screen and contains Qt file system). Click " $\mathfrak{T}\mathfrak{T}$ " to continue:



1



	E	1 					<u>? ×</u>	
		查找范围(I):	🗀 Li nux		• +	🗈 💣 🎟	,	
		表最近的文档 ぼう 東面 数的文档 変的 支档 変 数的 東面 大都 一 一 一 一 一 一 一 一 一 一 一 一 一	readme.txt root_condence root_qt_mous root_qt_p.im u-boot_zimage u-boot_zimage zimage_2.6_T	e.im e.bin e_T35.bin			2	
			, 文件名 (M):	root_qt_tp.img		-	打开(0)	
	M		文件名 @). 文件类型 (I):	All Files (*.*)			取消	
				AII FILES (#. #)				
Ya	ffs file s	ystem is trans	ferred and sav	ed automatically:				
	卷超级终端 文件(F) 编辑	- 超级终端 昆(E) 查看(Y) 呼叫((_) 传送(T) 帮助(H)					
F								
	RECETA	JED ETLE STZ	E:58497658 (641KB/S, 89S)		1		
	NAND e Erasir OK NAND w	erase: devic ng at Øx3ffc write: devic	e 0 offset 0 000 100% e 0 offset 0 x3868c00	x24c000, size 0x3d				
	###### [1] Da [2] Da [3] Da [4] Da [5] Da [6] Da [6] Da [9] Fa [0] Sa [a] Da Enter	Boot for ownload u-bo ownload Eboo ownload Linu ownload WinC ownload CRAM ownload to S oot the syst ormat the Na et the boot ownload User your select	Nor Flash ot to Nand F t x Kernel E NK.bin FS image S image DRAM & Run em nd Flash parameters Program ion: _					
1	已连接 0:05:4	7 ANSIW 11520	D 8-N-1 SCROLL	CAPS NUM 捕打印				11.

Step 7, enter "7": Download to SDRAM & Run, as shown in the following diagram:





◆ 超级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	<u>_ ×</u>
###### EmbedSky BIOS for SKY2440 ##### Press Space key to stop autoboot ! ###### Boot for Nor Flash ##### 11 Download u-boot to Nand Flash 121 Download Linux Kernel 131 Download CRAMFS image 161 Download CRAMFS image 171 Download CRAMFS image 171 Download CRAMFS image 181 Boot the system 191 Format the Nand Flash 101 Set the boot parameters 11 Download User Program Enter your selection: 7 USB host is connected. Waiting a download.	
	11.

Open the software DNW when the upper sentence "USB hosting is connected, Waiting a download." appears (you need to wait about 5 minutes for the sentence). Click "Transmit" in "USB Port" menu:

Serial Port USB Port Configu	
Transmit 0	<u>^</u>
Rx Test	
Status	
	-

Locate "TQ2440_Test_xxxx.bin" under the directory "Images" in CD-ROM (xxxx represents LCD resolution). Click "打开" to continue:

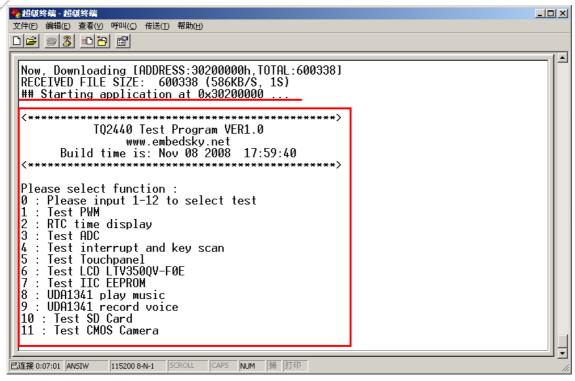




打开		<u>? ×</u>
查找范围(L):	🗁 Images 🔹 🗧 🕂 🎫 🗸	
表最近的文档 定 支 最近 の 文 相 で 。 点 面 、 一 、 一 、 一 、 一 、 一 、 一 、 一 、 一 、 一 、 一 の 、 一 、 一 の 、 一 、 一 の 、 一 、 、 、 、 、 、 、 、 、 、 、 、 、	Linux WinCE dnw.ini h-flash.hfc readme.txt TQ2440_Test_240320.bin U-boot_zImage.bin U-boot_zImage_T35.bin	2
	文件名 (M): TQ2440_Test_240320.bin 🔽	打开 @)
	文件类型 ①:	取消

"TQ2440_Test_xxxx.bin" starts to run automatically when download is complete, as shown in the following

diagram:



Step 8, enter "8": Boot the system, as shown in the following diagram:

The upper diagram displays Linux start-up, which is similar to WinCE start-up:





参超级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H) □ ご ご ご ご	
[a] Download User Program Enter your selection: ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download VYRFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: [8] Boot the System [9] Format the Nand Flash: Sector addr on NAND: 0x720 IotalSector: 0xb922 LoadAddress: 0x30200000 JumpAddr: 0x30201000	

Step 9, enter "9": Format the Nand Flash (be cautious to use this funtion), as shown in the following diagram:

(this option is corresponding to format Nand Flash)

ź

The maxium formating length is 4000000, namely 64MB Nand Flash.

🌺 超级终端 - 超级终端	- 🗆 🗙
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
[9] Format the Nand Flash 10] Set the boot parameters [a] Download User Program Enter your selection: 9 Start address: 0x0 Size(eg. 4000000, 0x4000000, 64m and so on): 64m NAND erase: device 0 whole chip Erasing at 0x3ffc000 100% complete.	
OK ###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection:	
ビ注接 0:08:15 ANSIW 115200 8-N-1 SCROLL CAPS NLM 捕 打印	

If there are bad blocks in Nand Flash, prompts will appear in the interface during formating, as shown in the following diagram:



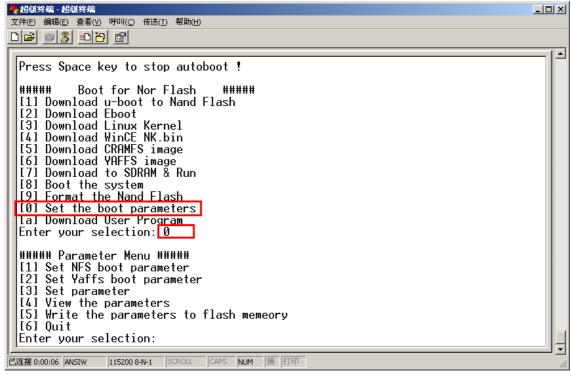


参超级终端 - 超级终端 文件(E) 編掲(E) 査看(Y) 呼叫(C) 传送(T) 帮助(H) □	
[9] Format the Nand Flash [0] Set the boot parameters Enter your selection: 9 Start address: 0x0 Size(eg. 4000000, 0x4000000, 64m and so on): 0x4000000	
NAND erase: device 0 whole chip Skipping bad block at 0x00624000 Skipping bad block at 0x038f8000 Erasing at 0x3ffc000 100% complete. OK	
###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image	
[6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters	
Eliz接1:01:36 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	_

(caution: All information preserved in Nand Flash will be wiped up when formatting. Please backup your data before formatting.

Bad block is always caused by frequent reading, writing or burning Nand Flash. If you often encouter operational errors, it is suggested to format Nand Flash which may recover some bad blocks.)

Step 10, enter "0": Set the boot parameters, as shown in the following diagram:



Enter "1": Set NFS boot parameter, as shown in the following diagram:

(caution: You can also enter "3" to set parameter manually)



, i



参 超级终端 - 超级终端 文件(F) 編録(E) 查看(W) 呼叫(C) 传送(T) 帮助(H)		
######Boot for Nor Flash#####[1] Download u-boot to Nand Flash[2] Download Eboot[3] Download Linux Kernel[4] Download WinCE NK.bin[5] Download CRAMFS image[6] Download VAFFS image[7] Download to SDRAM & Run[8] Boot the system[9] Format the Nand Flash[0] Set the boot parameters[a] Download User ProgramEnter your selection:		
<pre>###### Parameter Menu ##### [1] Set NFS boot parameter [2] Set Yaffs boot parameter [3] Set parameter [4] View the parameters [5] Write the parameters to flash memeory [6] Quit Enter your selection: 1 Enter the PC IP address:(xxx.xxx.xxx)</pre>		
CAPS NUM 描 打印	11.	

Input IP address of PC (we use 192.168.1.10 here) (caution: What you input here is exactly the IP address of Linux on PC. If your Linux is running in virtual machine, input the IP of Linux in virtual machine):

◆ 超级终端 - 超级终端 _ □ □ 文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)	IX
	_
###### Boot for Nor Flash ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0	
<pre>###### Parameter Menu ###### [1] Set NFS boot parameter [2] Set Yaffs boot parameter [3] Set parameter [4] View the parameters [5] Write the parameters to flash memeory [6] Quit Enter your selection: 1 Enter the PC IP address:(xxx.xxx.xxx) 192.168.1.10_</pre>	

Input IP address of platform (we use 192.168.1.6 here.)(caution: The IP address input here should be same as the IP set in NFS , otherwise NFS can not start up. The default IP set in NFS is 192.168.1.6, you can modify the file "etc/init.d/rcS" to reset IP address.):





★超级装装 - 超级装装 文件E 编辑E 查看(2) 呼叫(2) 传送(1) 帮助(1) □ 2 2 3 □ 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
Image:
[2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0 ###### Parameter Menu ##### [1] Set NFS boot parameter
<pre>[3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0 ###### Parameter Menu ##### [1] Set NFS boot parameter</pre>
<pre>[5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0 ###### Parameter Menu ##### [1] Set NFS boot parameter</pre>
<pre>[7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0 ####### Parameter Menu ###### [1] Set NFS boot parameter</pre>
<pre>[9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0 ####### Parameter Menu ###### [1] Set NFS boot parameter</pre>
<pre>[0] Set the boot parameters [a] Download User Program Enter your selection: 0 ###### Parameter Menu ##### [1] Set NFS boot parameter</pre>
Enter your selection: 0 ###### Parameter Menu ##### [1] Set NFS boot parameter
[1] Set NFS boot parameter
[2] Set Yaffs boot parameter
[3] Set parameter
[4] View the parameters [5] Write the parameters to flash memeory
[6] Quit Enter your selection: 1
Enter the PC IP address:(xxx.xxx.xxx) 192.168.1.10
Enter the SKY2440 IP address:(xxx.xxx.xxx)
已连接 0:02:08 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

Enter return-key and enter the Mask IP address (typically 255.255.255.0):

² ●超级终端 - 超级终端	
文件(E) 編辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
	I • [
[4] Download WinCE NK.bin	
[5] Download CRAMFS image	
[6] Download YAFFS image	
[7] Download to SDRAM & Run	
[8] Boot the system	
[9] Format the Nand Flash	
[0] Set the boot parameters	
[[a] Download User Program	
Enter your selection: 0	
##### Parameter Menu #####	
[1] Set NFS boot parameter	
[2] Set Yaffs boot parameter	
[3] Set parameter	
[4] View the parameters	
[5] Write the parameters to flash memeory	
[6] Quit	
Enter your selection: 1	
Enter the PC IP address:(xxx.xxx.xxx)	
192.168.1.10	
Enter the SKY2440 IP address:(xxx.xxx.xxx) 192.168.1.6	
Enter the Mask IP address:(xxx.xxx.xxx)	
255.255.255.0_	
	Ţ
已连接 0:02:34 ANSIW 115200 8-N-1 SCROLL CAP5 NUM 描 打印	1.

Enter return-key to continue:





後超級终端 - 超級终端 文件(E) 編輯(E) 查看(V) 呼叫(C) 传送(I) 帮助(H)
[2] Set Yaffs boot parameter [3] Set parameter
[4] View the parameters
[5] Write the parameters to flash memeory
[6] Quit Enter your selection: 1
Enter the PC IP address:(xxx.xxx.xxx)
192.168.1.10 Enter the SKY2440 IP address:(xxx.xxx.xxx)
192.168.1.6
Enter the Mask IP address:(xxx.xxx.xxx)
bootargs: console=ttySAC0 root=/dev/nfs nfsroot=192.168.1.10:/opt/EmbedSky/root
nfs ip=192.168.1.6:192.168.1.10:192.168.1.6:255.255.255.0:SKY2440.embedsky.net:e
Parameter Menu ##### [1] Set NFS boot parameter
[2] Set Yaffs boot parameter
[3] Set parameter [4] View the parameters
[5] Write the parameters to flash memeory
[6] Quit Enter your selection: _
已连接 0:02:54 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

Enter "2": Set Yaffs boot parameter, as shown in the following diagram:

文件(E) 編辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
[8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0	
<pre>###### Parameter Menu ##### I11 Set NFS boot parameter I21 Set Yaffs boot parameter I33 Set parameter I41 View the parameters I51 Write the parameters to flash memeory I61 Quit Enter your selection: 2 bootargs: noinitrd root=/dev/mtdblock2 init=/linuxrc console=ttySAC0</pre>	
###### Parameter Menu ##### [1] Set NFS boot parameter [2] Set Yaffs boot parameter [3] Set parameter [4] View the parameters [5] Write the parameters to flash memeory [6] Quit Enter your selection:	
CAP5 NUM 捕 打印	

Enter "3": Set parameter, as shown in the following diagram:





#纽茲總維 文件E 編輯E 查看公 呼叫C 传送[] 帮助任) 文件E 編輯E 查看公 呼叫C 传送[] 帮助任) ご ご ご ご ご ご ご ###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download VAFFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0
Image: Section 1 Image: Section 2 ###### Boot for Nor Flash ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download VAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program
Boot for Nor Flash ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download VAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program
Boot for Nor Flash ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download VAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program
<pre>[1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>
<pre>[1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>
<pre>[1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>
<pre>[2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMES image [6] Download CRAMES image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>
<pre>[4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>
<pre>[5] Download CRAMFS image [6] Download VAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>
<pre>[6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program</pre>
[7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program
[8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program
[9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program
[0] Set the boot parameters [a] Download User Program
[a] Download User Program
Parameter Menu
[1] Set NFS boot parameter
[2] Set Yaffs boot parameter
[3] Set parameter
[4] View the parameters
[5] Write the parameters to flash memeory [6] Quit
Enter your selection: 3
Name:
■ Ci车接 0:04:23 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

Step 10.3.1, input the parameter "bootargs" (this parameter is transferred to Linux by u-boot when start-up). Enter "回车" to continue: (we use NFS start-up configuration for example here)

後 超级终端 - 超级终端 文件(E) 編輯(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)	<u>_ </u>
###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash	
[2] Download Eboot [3] Download Linux Kernel	
[4] Download WinCE NK.bin [5] Download CRAMES image	
[6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system	
[9] Format the Nand Flash [0] Set the boot parameters	
Ial Download User Program Enter your selection: 0	
##### Parameter Menu #####	
[1] Set NFS boot parameter [2] Set Yaffs boot parameter	
[3] Set parameter [4] View the parameters	
[5] Write the parameters to flash memeory [6] Quit	
Enter vour selection: 3 Name: bootargs.	
 【注接 0:04:48 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	

Step 10.3.2, input NFS boot parameter

, i

"console=ttySAC0 root=/dev/nfs nfsroot=192.168.1.10:/opt/EmbedSky/root_nfs ip=192.168.1.6:192.168.1.10:192.168.1.6:255.255.0:TQ2440.embedsky.net:eth0:off",

Enter return-key to continue (not include the quotation marks)





後超级终端 - 超级终端 文件(E) 編輯(E) 查看(V) 呼叫(C) 传送(I) 帮助(H)	1
<pre>[2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download VAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0 ###### Parameter Menu ##### [1] Set NFS boot parameter [2] Set Yaffs boot parameter [3] Set parameter [4] View the parameters to flash memeory [6] Quit Enter your selection: 3 Name: bootargs Value: console=ttySAC0 root=/dev/nfs nfsroot=192.168.1.10:/opt/EmbedSky/root_nfs ip=192.168.1.6:192.168.1.10:192.168.1.6:255.255.255.0:TQ2440.embedsky.net:eth0: off</pre>	
已连接 0:09:14 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印 The following diagram briefly illustrates the parameter:	1

超级终端 - 超级终端
:件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)
Parameter Menu
[1] Set NFS boot parameter
[2] Set Yaffs boot parameter
[3] Set parameter
[4] View the parameters
[5] Write the parameters to flash memeory
[6] Quit Content of the Content of the
Enter vour selection: 3 Name of the parameters Content of the parameters
Name: bootargs
/alue: console=ttySHC0 root=/dev/nts ntsroot=192.168.1.10:/opt/EmbedSky/root_nts
ip=192.168.1.6:192.168.1.10:192.168.1.6:255.255.255.0:TQ2440.embedsky.net:eTh0:
setenv bootargs console=ttySAC0 root=/dev/nfs nfsroot=192.168.1.10:/opt/EmbedSky
/root_nfs_ip=192.168.1.6:192.168.1.10:192.168.1.6:255.255.255.0:TQ2440.embedsky.
net:eth0:off
Parameter Menu
[1] Set NFS boot parameter [2] Set Yaffs boot parameter
[3] Set parameter [4] View the parameters
[5] Write the parameters to flash memeory
[6] Ouit
Enter your selection: _
生接 0:10:06 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

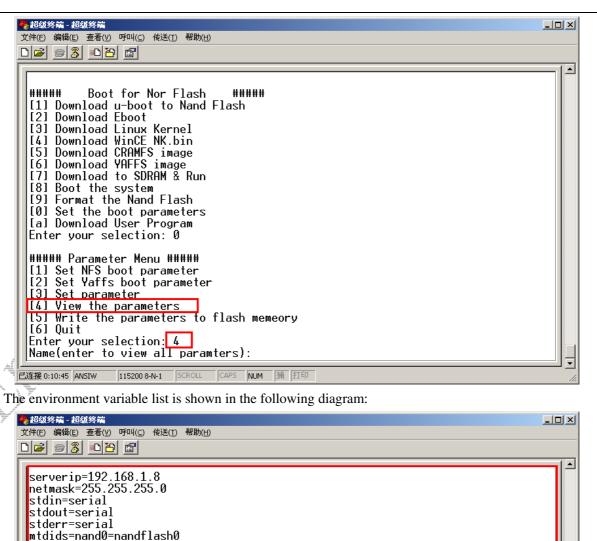
Enter "5" to save the configuration.

1

Enter "4": View the parameters. The parameter list appears in the following diagram:





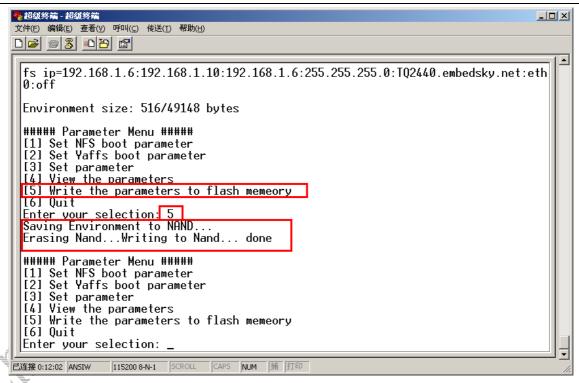


文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
	•
serverip=192.168.1.8	_
netmask=255.255.255.0	
stdin=serial	
stdult-serial	
stderr=serial	
mtdids=nand0=nandflash0	
mtdparts=mtdparts=nandflash0:256k@0(bios),48k(params),2m(kernel),-(root)	
partition=nand0,0	
mtddevnum=0	
mtddevname=bios	
bootargs=console=ttySAC0_root=/dev/nfs_nfsroot=192.168.1.10:/opt/EmbedSky/root_n	
fs ip=192.168.1.6:192.168.1.10:192.168.1.6:255.255.255.0:TQ2440.embedsky.net:eth	
0:off	
Environment size: 516/49148 bytes	
###### Parameter Menu #####	
[1] Set NFS boot parameter	
[2] Set Yaffs boot parameter	
[3] Set parameter	
[4] View the parameters	
[5] Write the parameters to flash memeory	
[6] Quit	
Enter your selection: _	
<u></u>	•
已连接 0:11:04 ANSTW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	1

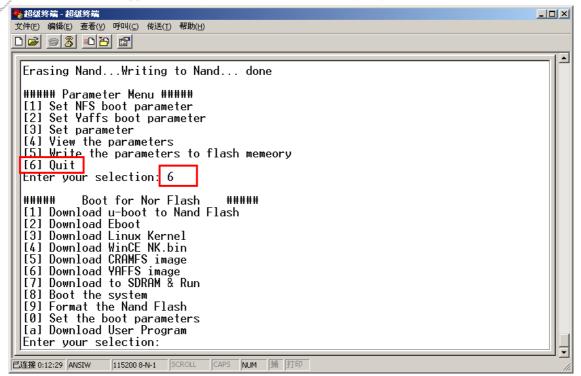
Enter "5": Write the parameters to flash memory, as shown in the following diagram:







Enter "6": Quit to the upper directory, as shown in the following diagram:



Step11, enter "a": Download User Program to Nand Flash, starting from block 0. For example, write non-OS testing program to Nand Flash:



Í



🏀 超级终端 - 超级终端	. <u> </u>
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
<pre>###### EmbedSky BIOS for SKY2440 ###### Press Space key to stop autoboot ! ###### Boot for Nor Flash ###### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download VAFFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: [a] USB host is connected. Waiting a download.</pre>	
已连接 0:00:43 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

Open the software DNW when the upper sentence "USB hosting is connected, Waiting a download." appears (you need to wait about 5 minutes for the sentence). Click "Transmit" in "USB Port" menu:

Serial Port USB Port Configu	
Serial Port USB Port configure 2	A
	×

Locate "TQ2440_Test_xxxx.bin" under the directory "Images" in CD-ROM (xxxx represents LCD resolution). Click "打开" to continue:





打开		<u>?</u> ×
查找范围(I):	🗀 Images 💌 🗲 🖆 🏢 🗸	
ま 我最近的文档 び 泉面 参 教的文档 歌 歌 町 一 歌 一 一 一 一 一 一 一 一 一 一 一 一 一	Unux WinCE dnw.ini Inflash.hfc To2440_Test_240320.bin U-boot_zImage.bin U-boot_zImage_T35.bin	2
	文件名 @): TQ2440_Test_240320.bin 🔽	打开 (0)
	文件类型 Œ): All Files (*.*)	取消

"TQ2440_Test_xxxx.bin" runs automatically when download is complete, as shown in the following diagram:

餐稻级终端 - 超级终端	
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H) □ 😂 📨 🎦 📴 🎦 📴	
RECEIVED FILE SIZE: 600338 (586KB/S, 1S)	
NAND erase: device 0 offset 0x0, size 0x94000 Erasing at 0x90000 100% complete. OK	
NAND write: device 0 offset 0x0, size 0x94000	
Writing data at 0x93e00 100% complete. 606208 bytes written: OK	
<pre>###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection:</pre>	
已连接 0:01:41 ANSIW 115200 8-N-1 SCROLL CAP5 NUM 捕 打印	11.

The platform starts up from Nand Flash, as shown in the following diagram:





◆ 超级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(T) 帮助(H)	
[9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection:	
<pre></pre>	
Please select function : Ø : Please input 1-12 to select test 1 : Test PWM 2 : RTC time display 3 : Test ADC	
4 : Test interrupt and key scan 5 : Test Touchpanel 6 : Test LCD LTV3500V-F0E 7 : Test IIC EEPROM 8 : UDA1341 play music 9 : UDA1341 record voice	
10 : Test SD Card 11 : Test CMOS Camera 已连接0:02:44 ANSIW 115200 8-N-1 SCROLL CAPS NUM 備打印	

The following 2 operations are hidden operations:

Í

Step12, download u-boot to Nor Flash. This operation needs start-up from Nor Flash.

Caution: Please restart the platform after writing u-boot to Nor Flash if you try to burn other programs. Otherwise system failure might happen.

Enter lowercase "o", as shown in the following diagram:

	- O ×
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
###### EmbedSky BIOS for SKY2440 #####	
Press Space key to stop autoboot !	
###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system	
[9] Format the Nand Flash [0] Set the boot parameters	
[a] Download User Program	
Enter your selection:	
USB host is connected. Waiting a download.	
-	

Use software DNW to select the file "u-boot_zImage_xxxx.bin" (xxxx represents LCD type):



Í



打开					<u>? ×</u>	
查找范围(<u>L</u>):	🗀 Images		- + 🗈	-111 *		
ま最近的文档 支最近的文档 運動 東面 教的文档 戦的文档 戦的电脑 変更	Linux WinCE dnw.ini n-flash.hfc readme.txt TQ2440_Test_2 U-boot_zImage	.hin				
网上邻居				2	2	
	文件名(₪):	u-boot_zImage_T35.bin		• I	J开 (0)	
	文件类型(I):	All Files (*.*)		<u> </u>	取消	
u-boot is download and s	saved automati	cally. The following	diagram displa	ays the proc	ess writing N	Vor Flash:
) 传送(<u>I</u>) 帮助(<u>H</u>)				_	
[3] Download Linux [4] Download WinCl [5] Download CRAM [6] Download YAFFS [7] Download to SI [8] Boot the syste [9] Format the Nam [0] Set the boot p [a] Download User Enter your select: USB host is connect	E NK.bin FS image S image DRAM & Run em nd Flash parameters Program	a download.				
RECÉIVED FILE SÎZI Un-Protect Flash I Erasing sector 0 Erasing sector 1 Erasing sector 2 Erasing sector 3 Erasing sector 4 Erasing sector 5 Erasing sector 6 Erased 7 sectors	E: 239266 (2 Bank # 1 ok. ok. ok. ok. ok. ok. ok.	10000h,TOTAL:239266 333KB/S, 1S)	5]			
Erasing sector 6	ok.					

Please reset the platform when writing Nor Flash is finished:

115200 8-N-1 SCROLL CAPS NUM 捕 打印

______ 已连接 0:00:50 ANSIW





參超銀终端 - 超銀终端	_ 🗆 ×
文件(E) 編辑(E) 査看(Y) 呼叫(C) 传送(I) 帮助(H)	
Un-Protect Flash Bank # 1 Erasing sector 0 ok. Erasing sector 1 ok. Erasing sector 2 ok. Erasing sector 3 ok. Erasing sector 4 ok. Erasing sector 5 ok. Erasing sector 6 ok. Erased 7 sectors Copy to Flash done ###### Boot for Nor Flash #####	
[1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection:	
已连接 0:01:26 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.
The following diagram displays boot for Nand Flash:	
▲超级终端 - 超级终端	
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
###### EmbedSky BIOS for SKY2440 ###### Press Space key to stop autoboot ! ###### Boot for Nand Flash I1 Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: 0 USB host is connected. Waiting a download.	

The following diagram displays a failure situation:

. 已连接 0:00:09 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

Í



Í



[0] Set the boot parameters [a] Download User Program Enter your selection: 0	
USB host is connected. Waiting a download.	
Now, Downloading [ADDRESS:30000000h,TOTAL:239266] RECEIVED FILE SIZE: 239266 (233KB/S, 1S) Un-Protect Flash Bank # 1 Erasing sector 0 Erased 7 sectors Copy to Flash Flash not Erased	
###### Boot for Nand Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image	
<pre>[7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: _</pre>	

Step13, back to u-boot operation interface: enter the key "q" to back to u-boot operation interface, as shown in the following diagram:

※ 超级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
	1
##### EmbedSky BIOS for SKY2440 ##### Press Space key to stop autoboot !	
<pre>##### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: [9] EmbedSky></pre>	

Execute the command "menu" to go back to one-key operation interface:





後継 後 後 後 文件(E) 編輯 (E) 查看 (Y) 呼叫(C) 传送(T) 帮助(E) 日 □ □ □ □ □ □ □ □	
<pre>[4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download YAFFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: q EmbedSky> menu</pre>	
<pre>###### Boot for Nor Flash ##### [1] Download u-boot to Nand Flash [2] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [6] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection:</pre>	

The following contents illustrate the download processs in uboot operation interface:

Start TFTP proxy "TFTP Server TFTPDWIN" in Windows:

Í

TFTP Server TFT	PDWIN								
iystem Transmission	n Help								
Þ 💥 👒			۹	() □					
IP	Status	Rate File		File size P	rogress	% Started at	Ended at	Remain	Message
									Þ
Server turned-on									li.
			_						
the icon "		" to stop all	the	operation	is of 'l	FTP. The	icon	turns i	nto "
· · · · ·			.1	C 11 ·	1.				

"System->Setting" to start setting, as shown in the following diagram:



Í



Settings	Help	R.	• 6				
Hide the program Exit program	Rate File	File size	 % Started at	Ended at	Remaining time	Message	
	-						

Set the saving path in page "General". We set "F:\My Documents" here:

2 Settings	×
General Input Output Firewall	
Server port 69	
Maximum number of transmissions 20	
Timeout	
Number of attempts 5 Version blksize	
Attempt duration 10 s 🔽 timeout	
🔽 tsize	
✓ Turn-on logging — 1	-11
F:\My Documents Browse	
Turn-on the server on start	
Save Default 确定 取消	

Set output directory in page "Output" (base on your actual situation), as shown in the following diagram:





General Input Output Tirewall	×
Output directory h:\images Browse Output directory only	
Delete file after transmitting Listed files only Add	
Edit Delete	
Save Default 确定 取消	

Click"save " to save your configuration:

Í

General Input Output Firewall	×
Output directory h:\images Output directory only	Browse
Delete file after transmitting Listed files only	Add Edit Delete
Default 确定	取消

A box "Options have been stored" appears, click "确定" to go back to the upper interface



Click " \mathfrak{m} \mathfrak{z} " to complete the setting:



_



		Settings	
		General Input Output Firewall	
		Output directory	
		h:\images Browse	
		Output directory only	
		Delete file after transmitting	
		Listed files only	
		Add	
		Edit	
		Delete	
	AV		
$\wedge \frown$			
		Save Default 備定 取消	
$\langle X \rangle \langle Y \rangle$			
Click the ic	on "	start TFTP service, and the icon turns into "	
	System Transmission H	VIN	>
	IP Status	Rate File File Progress % Started at Ended at Remaining time Message	
	Server turned-on		

Step13, configure network parameters before using the download function of TFTP (the default IP address set by us probably doesn't fit your need, you can reset it according to the actual situation):

1, set IP address of platform: Execute the command "setenv ipaddr 192.168.1.6", as shown in the following diagram:



Í



🍫 超级终端 - 超级终端 - O X 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H) D 🖻 🚿 🔊 🔁 🖆 ##### EmbedSky BIOS for SKY2440 ##### Press Space key to stop autoboot ! ##### ##### Boot for Nor Flash [1] Download u-boot to Nand Flash [2] Download Eboot [2] Download Eboot
[3] Download Linux Kernel
[4] Download WinCE NK.bin
[5] Download CRAMFS image
[6] Download YAFFS image
[7] Download to SDRAM & Run
[2] Boot the curtar [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: q EmbedSky> <u>setenv ipaddr 192.168.1.6</u> EmbedSky> 115200 8-N-1 SCROLL CAPS NUM 捕 打印 已连接 0:00:18 ANSIW

2, set IP address of TFTP server (this IP should be the same as the IP in Windows OS. We use "192.168.1.8"

here). Execute the command	"setenv serverip	192.168.1.8",	as shown in	the following diagram:
------	------------------------	------------------	---------------	-------------	------------------------

後超線発端 - 超線発端	
文件(E) 编辑(E) 查看(Y) 呼叫(⊆) 传送(T) 帮助(H)	
##### EmbedSky BIOS for SKY2440 #####	
Press Space key to stop autoboot !	
###### Boot for Nor Flash #####	
[1] Download u-boot to Nand Flash [2] Download Eboot	
[3] Download Linux Kernel	
[4] Download WinCE NK.bin	
[5] Download CRAMFS image	
[[6] Download YAFFS image	
[7] Download to SDRAM & Run	
[8] Boot the system [9] Format the Nand Elash	
[0] Set the boot parameters	
Ial Download User Program	
Enter your selection: q	
EmbedSky> setenv ipaddr 192.168.1.6	
EmbedSky> setenv serverip 192.168.1.8	
EmbedSky> _	
	13
Ci连接 0:00:53 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

3, set netmask. Execute the command "setenv netmask 255.255.255.0", as shown in the following diagram:





文件(E) 编辑(E) 查看(Y) 呼叫(⊆) 传送(I) 帮助(H)	
<pre>##### EmbedSky BIOS for SKY2440 ##### Press Space key to stop autoboot ! ###### Boot for Nor Flash ##### 11 Download u-boot to Nand Flash [2] Download Eboot [3] Download Eboot [3] Download Linux Kernel [4] Download WinCE NK.bin [5] Download CRAMFS image [6] Download CRAMFS image [6] Download CRAMFS image [7] Download to SDRAM & Run [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters [a] Download User Program Enter your selection: q EmbedSky> setenv ipaddr 192.168.1.6 EmbedSky> setenv netmask 255.255.0 EmbedSky> _</pre>	
4, set MAC. Execute the command "setenv ethaddr 0a:1b:2c:3d:4e:5f", as shown in the following	diagram:
そ超級终端 - 超級终端 文件(E) 編輯(E) 查看(V) 呼叫(C) 传送(T) 帮助(H) □ ご 回 ③ ③ □ □ □ □ [8] Boot the system [9] Format the Nand Flash [0] Set the boot parameters	

	Laj Download User Program Enter vour selection:
	Linter your selection.
	###### Boot for Nor Flash #####
	[1] Download u-boot to Nand Flash
	[2] Download Eboot
	[3] Download Linux Kernel
	[4] Download WinCE NK.bin
	[5] Download CRAMFS image
	[6] Download YAFFS image [7] Download to SDRAM & Run
	[8] Boot the system
	[9] Format the Nand Flash
	[0] Set the boot parameters
	[a] Download User Program
	Enter your selection: q
	EmbedSky> setenv ipaddr 192.168.1.6
	EmbedSky> setenv serverip 192.168.1.8
	EmbedSky> setenv netmask 255.255.255.0
	EmbedSky> setenv ethaddr 00:11:22:33:44:55 EmbedSky>
- b	

5, check your settings, as shown in the following diagram:





後超级终端 - 超级终端 文件(E) 編輯(E) 查看(W) 呼叫(C) 传送(I) 帮助(H)	
EmbedSky> setenv ipaddr 192.168.1.6 EmbedSky> setenv serverip 192.168.1.8 EmbedSky> setenv netmask 255.255.05 EmbedSky> setenv ethaddr 00:11:22:33:44:55 EmbedSky> printenv bootargs=noinitrd root=/dev/mtdblock2 init=/linuxrc console=ttySAC0 bootad=nboot 0x32000000 kernel; bootm 0x32000000 bootdelay=1 baudrate=115200 stdin=serial stdout=serial stdout=serial stdout=serial stderr=serial mtdids=nand0=nandflash0 mtdparts=mtdparts=nandflash0:256k@0(bios),48k(params),2m(kernel),-(root) partition=nand0,0 mtddevnum=0 mtddevname=bios Ipaddr=192.168.1.8 netmask=255.255.0 Environment size: 418/49148 bytes EmbedSky>	
E连接 0:04:35 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印]J //

Step13, execute TFTP command to download u-boot to Nand Flash:

The TFTP command: tftp 0x30000000 u-boot_zImage_T35.bin; nand erase bios; nand write.jffs2 0x30000000 bios \$(filesize)

Including 3 parts: 1, TFTP downloads "u-boot_zImage_T35.bin" to SDRAM starting from the address 0x30000000; 2, wipe up BIOS partition (from 0x0 to 0x40000) of Nand Flash; 3, save the data begin from the address 0x30000000 to BIOS partition in Nand Flash.

TFTP proxy state is shown in the following diagram:

Í





1010	[FTP Server TF]	IPDWIN								
Sys	tem Transmissio	n Help								
	> > •			🚯 🐘 🍕						
	IP	Status	Rate	File	File size	Progress	%	Started at	Ended at	Remaining
1	192.168.1.6	Finished		u-boot_zimage_t35.bin	233.65 KB	233.65 KB	100%	13:23:34	13:23:34	
—										
•										F
Ø	5erver turned-on									
-										

Step14, execute TFTP command to download Linux kernel to Nand Flash:

The TFTP command: tftp 0x30000000 zImage_2.6_T35_uboot.bin; nand erase kernel; nand write.jffs2 0x30000000 kernel \$(filesize)

The Linux kernel partition: From 0x4C000 to 0x24C000

	超级终端 - 超级终端	l ×
	件(E) 編辑(E) 査看(Y) 呼叫(C) 传送(I) 帮助(H)	
븓		
	<pre>Writing data at 0x3a600 100% complete. 239256 bytes written: 0K EmbedSky> tftp 0x30000000 zImage_2.6_T35_uboot.bin;nand erase kernel;nand write. jffs2 0x30000000 kernel \$(filesize) dm9000 1/o: 0x20000300, 1d: 0x90000a46 MAC: 00:11:22:33:44:55 TFTP from server 192.168.1.8; our IP address is 192.168.1.6 Filename 'zImage_2.6_T35_uboot.bin'. Load address: 0x3000000 Loading: T ###################################</pre>	
11	NAND erase: device 0 offset 0x4c000, size 0x200000 Erasing at 0x248000 100% complete. OK	
	NAND write: device 0 offset 0x4c000, size 0x181918	
11	Writing data at 0x1cd800 100% complete. 1579288 bytes written: OK EmbedSky>	
	连接 0:41:34 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	• <u> </u>

TFTP proxy state is shown in the following diagram:





	TFTP Server TF1									<u>_ ×</u>
oysi	tem Transmissio	in Help	-							
	• <u></u>			1 🐘 🔍						
	IP	Status	Rate	File	File size	Progress	%	Started at	Ended at	Remainin
1	192.168.1.6	Finished		u-boot_zimage_t35.bin	233.65 KB	233.65 KB	100%	13:23:34	13:23:34	
1	192.168.1.6	Finished		zimage_2.6_t35_ubo	1.51 MB	1.51 MB	100%	13:45:07	13:45:09	

Step15, execute TFTP command to download file system to Nand Flash:

The TFTP command: tftp 0x30000000 root_qt_tp.img; nand erase root; nand write.yaffs 0x30000000 root \$(filesize)

s(mesize)

The root partition: From 0x24C000 to 0x4000000

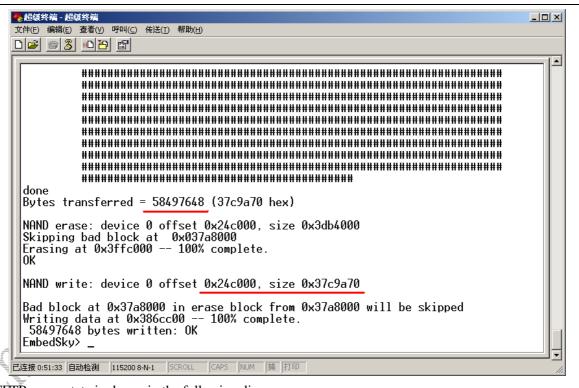
Caution: We use the parameter yaffs here because we are downloading yaffs file system.

经级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(①) 传送(I) 帮助(E)	×
Filename 'zImage_2.6_T35_uboot.bin'. Load address: 0x30000000 Loading: T ###################################	
done Bytes transferred = 1579288 (181918 hex)	
NAND erase: device 0 offset 0x4c000, size 0x200000 Erasing at 0x248000 100% complete. OK	
NAND write: device 0 offset 0x4c000, size 0x181918	
Writing data at 0x1cd800 100% complete. 1579288 bytes written: OK	
EmbedSky> tftp 0x30000000 root_qt_tp.img;nand erase root;nand write.yaffs 0x3000 0000 root \$(filesize)	
dm9000 i/o: 0x20000300, id: 0x90000a46 MAC: 00:11:22:33:44:55 TFTP from server 192.168.1.8; our IP address is 192.168.1.6	
Filename 'root_qt_tp.img'. Load address: 0x30000000 Loading: T ###################################	
	<u>▼</u> 1

The following diagram indicates that download is complete:







THTP proxy state is shown in the following diagram:

4

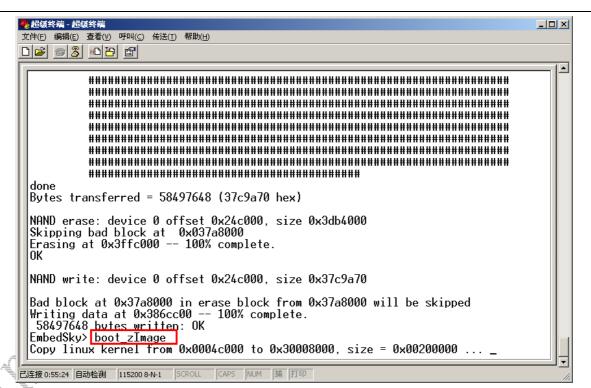
TTTP	FTP Server TF	TPDWIN								- D ×
Syst	tem Transmissio	n Help								
	• ¥ ¶				• 0					
	IP	Status	Rate	File	File size	Progress	%	Started at	Ended at	Remaining
1	192.168.1.6	Finished		u-boot_zimage_t35.bin	233.65 KB	233.65 KB	100%	13:23:34	13:23:34	
1	192.168.1.6	Finished		zimage_2.6_t35_ubo	1.51 MB	1.51 MB	100%	13:45:07	13:45:09	
1	192.168.1.6	Finished		root_qt_tp.img	55.79 MB	55.79 MB	100%	13:47:34	13:48:33	
•										Þ
	5erver turned-on									//.

Step16, execute TFTP command "boot_zImage" to start up Linux:



Á





Step17, execute TFTP command to download u-boot to Nor Flash (start-up from Nor Flash is the prerequisite):

The TFTP command: tftp 0x30000000 u-boot_zImage_T35.bin; protect off all; erase 0 +\$(filesize); cp.b 0x30000000 0 \$(filesize). Be cautious of the change here.

◆ 超级终端 - 超级终端
文件(E) 编辑(E) 查看(V) 呼叫(C) 传送(I) 帮助(H)
[0] Set the heat nemenators
[0] Set the boot parameters [a] Download User Program
Enter your selection: a
EmbedSky> tftp 0x30000000 u-boot_zImage_T35.bin; protect off all; erase 0 +\$(fil
esize); cp.b 0x30000000 0 \$(filesize)
dm9000 1/o: 0x20000300, 1d: 0x90000046
MAC: 0a:1b:2c:3d:4e:5f
TFTP from server 192.168.1.8; our IP address is 192.168.1.6
Filename 'u-boot_zImage_T35.bin'.
Load address: 0x30000000
Loading: T ###################################
done
Bytes transferred = 239256 (3a698 hex)
Un-Protect Flash Bank # 1
Erasing sector 0 ok.
Erasing sector 1 ok.
Erasing sector 2 ok.
Erasing sector 3 ok.
Erasing sector 5 ok.
Erasing sector 6 ok.
Erased 7 sectors
Copy to Flash done
EmbedSky>

TFTP proxy state is shown in the following diagram:

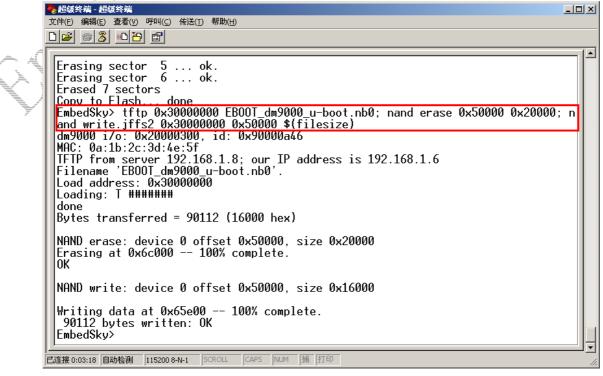




iystei	m Transmission	Help								
	* •		📄 🐘 <	0						
	IP	Status	Rate File	File size	Progress	%	Started at	Ended at	Remain	Me
1	192.168.1.6	Finished	u-boot_zimage_t35.bin	233.65 KB	233.65 KB	10	14:01:38	14:01:38		
									_	

Step18, execute TFTP command to download eboot to Nand Flash:

The TFTP command: tftp 0x30000000 EBOOT_dm9000_u-boot.nb0; nand erase 0x50000 0x20000; nand write.jffs2 0x30000000 0x50000 \$(filesize). 0x50000 is eboot start address in Nand Flash, and 0x20000 is the length of eboot partition.



TFTP proxy state is shown in the following diagram:

	* •			🚯 🐘 <	0 -	1					
	IP	Status	Rate	File	File size	Progress	%	Started at	Ended at	Remain	Me
1	192.168.1.6	Finished		u-boot_zimage_t35.bin	233.65 KB	233.65 KB	10	14:01:38	14:01:38		
1	192.168.1.6	Finished		eboot_dm9000_u-boo	88.00 KB	88.00 KB	10	14:04:06	14:04:06		





Step19, execute TFTP function of eboot to download and burn WinCE image:

Start up eboot in u-boot operation interface. The command is: eboot.

	2843(9) 着き(1) 100000000000000000000000000000000000	<u>_ ×</u>
	Erasing sector 6 ok. Erased 7 sectors Copy to Flash done EmbedSky> tftp 0x30000000 EB00T_dm9000_u-boot.nb0; nand erase 0x50000 0x20000; and write.jffs2 0x30000000 0x50000 \$(filesize) dm9000 i/o: 0x20000300, id: 0x90000a46 MAC: 0a:1b:2c:3d:4e:5f TFTP from server 192.168.1.8; our IP address is 192.168.1.6 Filename 'EB00T_dm9000_u-boot.nb0'. Load address: 0x30000000 Loading: T ######## done Bytes transferred = 90112 (16000 hex)	n
	NAND erase: device 0 offset 0x50000, size 0x20000 Erasing at 0x6c000 100% complete. OK	
	NAND write: device 0 offset 0x50000, size 0x16000	
ALL	Writing data at 0x65e00 100% complete. 90112 bytes written: OK EmbedSky>[eboot] Clear the free memory: [0x30000000~0x33f31f7c]	
7	- [注接後:1:3:44 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	

The following diagram display eboot start-up process:

À

ź

🍓 超级终端 - 超级终端 <u>- 🗆 ×</u> 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(T) 帮助(H) D 🚅 🍙 🔏 🖻 🖻 dwLoadAddress: 0x8C038000 dwJumpAddress: 0x8C038000 dwStoreOffset: 0x0 sgList[0].dwSector: 0x280 sgList[0].dwLength: 0x100 ÍD[1] { dwVersion: 0x1 dwSignature: 0x43465349 String: dwImageType: 0x6 dwTtlSectors: 0x0 dwLoadAddress: 0x0 dwJumpAddress: 0x0 dwStoreOffset: 0x0 chainInfo.dwLoadAddress: 0X0000000 chainInfo.dwFlashAddress: 0X00000000 chainInfo.dwLength: 0X0000000 g pViviWinceInfo = 0x301D8000, g pViviWinceInfo->dwViviWinceMagic = 0x00000000 Press [ENTER] to download image now, or [SPACE] to enter boot monitor. Initiating image download in 15 seconds. 已连接 0:14:57 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印

Enter space-key to enter into eboot download mode:





1	✤超级终端 - 超级终端	
	文件(F) 編辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
ŕ		
	1) IP address: 0.0.0.0	
	Subnet mask: 255.255.255.255	
	2) Boot delay: 15 seconds 3) DHCP: ENABLED	
	4) Reset TOC to default	
	5) Startup image: DOWNLOAD NEW 6) Program RAM image into Boot Media: DISABLED	
	7) MAC address: 00:00:00:00:00	
	8) Kernel Debugger: ENABLED 9) Format Boot Media for BinFS	
	B) Support BinFS: ENABLED	
	D) DOWNLOAD image now(Etherent) F) Low-level FORMAT Boot Media	
	L) LAUNCH existing Boot Media image	
	R) Read Configuration U) DOWNLOAD image now(USB)	
	W) Write Configuration Right Now	
	X) DOWNLOAD image to boot media, then LAUNCH it off the media T) Format a FATFS For NandFlash	
	Enter your selection: _	
1	<u></u>] <u>_</u> _]
	已连接 0:16:03 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.
nt	er "1" to set IP address and Subnet Masks of platform:	
	✤ 超级终端 - 超级终端	
	文件(E)编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
[3) DHCP: ENABLED	
	4) Reset TOC to default	
	5) Startup image: DOWNLOAD NEW	
	6) Program RAM image into Boot Media: DISABLED 7) MAC address: 00:00:00:00:00:00	
	8) Kernel Debugger: ENABLED	
	9) Format Boot Media for BinFS	
	B) Support_BinFS: ENABLED	
	D) DOWNLOAD image now(Etherent) F) Low-level FORMAT Boot Media	
	L) LAUNCH existing Boot Media image	
	R) Read Configuration	
	U) DOWNLOAD image now(USB) W) Write Configuration Right Now	
	X) DOWNLOAD image to boot media, then LAUNCH it off the media	
	T) Format a FATFS For NandFlash	
	Fata was also time 1	
	Enter your selection: 1	
- 1	Enter IP address, or CR for default (0.0.0.0): 192.168.1.6	

Enter "7" to set MAC address:

Enter Subnet Masks, or CR for default (255.255.255.255): 255.255.255.0_





		<u>_ </u>
	文件(E)编辑(E) 查看(W) 呼叫(C) 传送(T) 帮助(H) □ <mark>2 2 3 10 10 11 11 11 11 11 11 11 11 11 11 11 </mark>	
	Subnet mask: 255.255.255.0 2) Boot delay: 15 seconds 3) DHCP: DISABLED 4) Reset TOC to default 5) Startup image: DOWNLOAD NEW 6) Program RAM image into Boot Media: DISABLED 7) MAC address: 00:00:00:00:00 8) Kernel Debugger: ENABLED 9) Format Boot Media for BinFS	
	 B) Support BinFS: ENABLED D) DOWNLOAD image now(Etherent) F) Low-level FORMAT Boot Media L) LAUNCH existing Boot Media image R) Read Configuration U) DOWNLOAD image now(USB) W) Write Configuration Right Now X) DOWNLOAD image to boot media, then LAUNCH it off the media T) Format a FAIFS For NandFlash 	
4	Enter your selection: 7 Enter new MAC address in hexadecimal (hh.hh.hh.hh.hh.): 00.11.22.33.44.55	
15	已连接 0:17:36 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	- li.
/_	Following diagram indicates that the setting is complete:	
	文件(E) 编辑(E) 查看(V) 呼叫(C) 传送(I) 帮助(H)	
Ē		
	 1) IP address: <u>192.168.1.6</u> Subnet mask: <u>255.255.255.0</u> 2) Boot delay: <u>15 seconds</u> 3) DHCP: DISABLED 4) Reset TOC to default 5) Startup image: DOWNLOAD NEW 6) Program RAM image into Boot Media: DISABLED 7) MAC address: <u>00:11:22:33:44:55</u> 8) Kernel Debugger: ENABLED 9) Format Boot Media for BinFS 	
	B) Support BinFS: ENABLED D) DOWNLOAD image now(Etherent) F) Low-level FORMAT Boot Media L) LAUNCH existing Boot Media image R) Read Configuration U) DOWNLOAD image now(USB) W) Write Configuration Right Now X) DOWNLOAD image to boot media, then LAUNCH it off the media T) Format a FAIFS For NandFlash 	

Enter "W" to save all the settings:

Í

╝╤





2 超级终端 - 超级终端	
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(T) 帮助(H) □ 2 2 2 1 1 1 1 1 1 1	
	1
1) IP address: 192.168.1.6 Subnet mask: 255.255.25.0	
2) Boot delay: 15 seconds	
3) DHCP: DISABLED 4) Reset TOC to default	
5) Startup image: DOWNLOAD NEW	
6) Program RAM image into Boot Media: DISABLED 7) MAC address: 00:11:22:33:44:55	
8) Kernel Debugger: ENABLED	
9) Format Boot Media for BinFS	
B) Support BinFS: ENABLED	
D) DOWNLOAD image now(Etherent) F) Low-level FORMAT Boot Media	
L) LAUNCH existing Boot Media image R) Read Configuration	
U) DOWNLOAD image now(USB)	
W) Write Configuration Right Now X) DOWNLOAD image to boot media, then LAUNCH it off the media	
T) Format a FATFS For NandFlash	
Enter your selection: W	
■ C连接 0:18:08 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	_
Enter "F" to low-level format Nand Flash (no case sensitive):	
後超级终端 - 超级终端 文件(E) 編辑(E) 查看(V) 呼叫(C) 传送(I) 帮助(H)	
3) DHCP: DISABLED 4) Reset TOC to default	
5) Startup image: DOWNLOAD NEW	
6) Program RAM image into Boot Media: DISABLED 7) MAC address: 00:11:22:33:44:55	
8) Kernel Debugger: ENABLED	
9) Format Boot Media for BinFS	
B) Support BinFS: ENABLED	
D) DOWNLOAD image now(Etherent) F) Low-level FORMAT Boot Media	
L) LAUNCH existing Boot Media image	
R) Read Configuration U) DOWNLOAD image now(USB)	
W) Write Configuration Right Now	
X) DOWNLOAD image to boot media, then LAUNCH it off the media T) Format a FATFS For NandFlash	
Enter your selection: f	
Reserving Blocks [0x0 - 0x1B]	
Low-level format Blocks [0x1C - 0xFFF]	
	_

Enter "X" (no case sensitive) to download "NK.bin" to SDRAM and then burn it to Nand Flash, as shown in the following diagram:



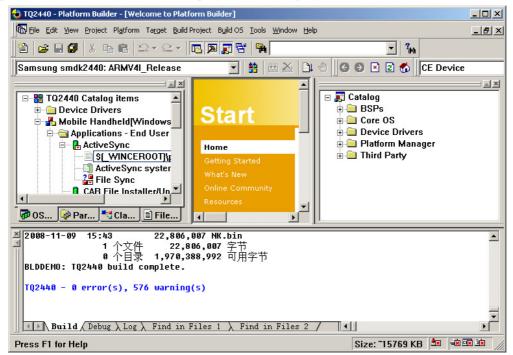
A



	🔗 超级终端 - 超级终端	
	文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
	dwTtlSectors: 0x0 dwLoadAddress: 0x0	
	dwLoadhadress: 0x0 dwJumpAddress: 0x0	
	dwStoreOffset: 0x0	
	chainInfo.dwloadAddress: 0X0000000	
	chainInfo.dwElashAddress: 0X0000000	
	chainInfo.dwLength: 0X0000000	
	-TOC Write	
	DM9000 Mac Address: 00:11:22:33:44:55	
	Found DM9000 ID:90000A46 at address A800000DM9000 work in 16 bus width	
	Not link of ethernet	
	DM9000 Mac : 00	
	DM9000 Mac : 11	
	DM9000 Mac : 22	
	DM9000 Mac : 33	
	DM9000 Mac : 44	
	DM9000 Mac : 55	
	DM9000_Init 0K.	
	System ready!	
	Preparing for download	
	Using device name: 'SMDK244017493'	
A		
	已连接 0:22:48 自动检测 115200 8-№1 SCROLL CAPS NUM 捕 打印	11.

The phrase "SMDK244017493" in the upper diagram represents the device in PB.

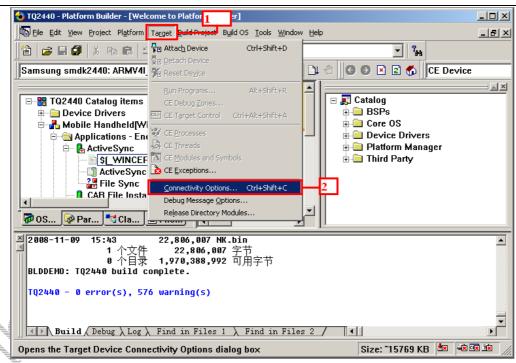
Start up "Platform Builder 5.0", and load project files (the project files loaded here needs to be compiled):



Click "Target->Connectivity Options" to configure PB of TFTP, as shown in the following diagram:







Select "Ethernet" under "Download":

Í

Device Configuration Add Device	<u>T</u> arget Device:		
	CE Device		1
Delete Device	Download:		o choo ction n
Service Configuration	None		406 S
Kernel Service Map			
Core Service Settings	T <u>r</u> ansport:		
Service Status	None	💌 Setti	ngs
	D <u>e</u> bugger:		
	None	Setti	ng <u>s</u>
	Apply	Close Hell	

Click "Settings". Enter the phrase "SMDK244017493" gotten previously and click "Ok":





	Ethernet Download Settings
	Device Boot Name:
	SMDK244017493
	IP Address: 192.168.1.6 Bootloader: 1.2
	Active Devices:
	SMDK244017493
1	
	2
	<u>O</u> K <u>C</u> ancel
	<u>QK</u> <u>Cancel</u>

Download mode configuration is complete, as shown in the following diagram:

Target Device Connectivity Optior	15	
Device Configuration <u>Add Device</u> <u>Delete Device</u>	<u>T</u> arget Device: CE Device Download:	T
Service Configuration	Ethernet	Setti <u>n</u> gs
Kernel Service Map	(SMDK244017493)	
<u>Core Service Settings</u> <u>Service Status</u>	Transport: None Dgbugger: None	Settings Setting <u>s</u>
	<u>Apply</u> <u>Close</u>	Help

Select "Ethernet" under "Transport", as the following diagram:

Í



CateDigi

Device Configuration Add Device	Target Device:	
Delete Device	CE Device	
Delete Device	Download:	
Service Configuration	Ethernet	Settings
Kernel Service Map	(SMDK244017493)	choose the
Core Service Settings	Transport	connection option
Service Status	Ethernet	Settings
	(SMDK244017493)	J Dettings
	D <u>e</u> bugger:	-
	None	Settings
	Apply Close	<u>H</u> elp

The configuration interface appears as the following diagram:

🔵 Target Device Connectivity Optio	ns	_ 🗆 X
Device Configuration <u>Add Device</u> <u>Delete Device</u>	Target Device: CE Device Download:	2
Service Configuration	Ethernet 💌	Setti <u>n</u> gs
Kernel Service Map	(SMDK244017493)	
Core Service Settings	T <u>r</u> ansport:	
Service Status	Ethernet 💌	Settings
	(SMDK244017493) D <u>e</u> bugger:	
	None	Setting <u>s</u>
	4 5 Apply Close	<u>H</u> elp

Click "Apply" and click "Close" to save and exit.

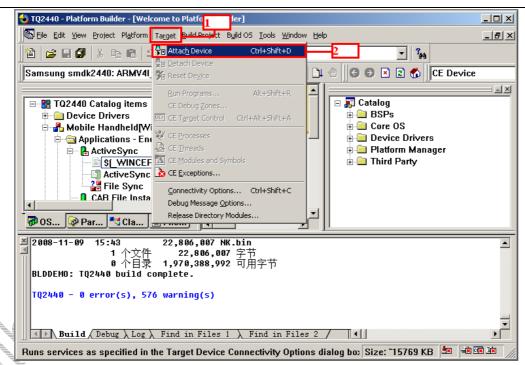
-

Click "Target->Attach Device" to start TFTP transmission:



Í





PB starts up TFTP to commucate with eboot after about 1 second, as shown in the following diagram:

_		
	文件(E) 編辑(E) 査看(V) 呼叫(C) 传送(I) 帮助(E)	
	DM9000 Mac : 22 DM9000 Mac : 33 DM9000 Mac : 44 DM9000 Mac : 55 DM9000_Init 0K. System ready! Preparing for download Using device name: 'SMDK244017493' +EbootSendBootmeAndWaitForTftp Sent B00TME to 255.255.255.255 Sent B00TME to 255.255.255.255 Sent B00TME to 255.255.255.255 Locked Down Link 1 Src IP 192.168.1.6 Port 0400 Dest IP 192.168.1.8 Port 05B2 EthDown::TFTPD_OPEN::boot.bin -EbootSendBootmeAndWaitForTftp 0EMMultiBINNotify: Download BIN file information: 	
	OEMVerifyMemory: StartAddr: 0x8C200000, Length:0x1631AE4 *** Downloading BIN image type *** TFTP: Desktop losing ACK, block number = 3669, Ack again 已连接2:13:58 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	

The transmission interface appears, as shown in the following diagram:





Downloading: F:\\nk.bin Estimated time Download thernet	The dialog box will be closed when download complete.
Transfer rate: 56 KB/sec	download com;

Click "Close" to exit after stransmission is complete:

A N			
	👈 Download Runtime Imag	ge to CE Device	
\sim	Download Complete!		
	Download E	(21.7 MB of 21.7 MB copied) thernet	
AV	Transfer rate: 22	26 KB/sec	
	🗌 Close this <u>d</u> ialog	box when download comp	
		Can Can	cel

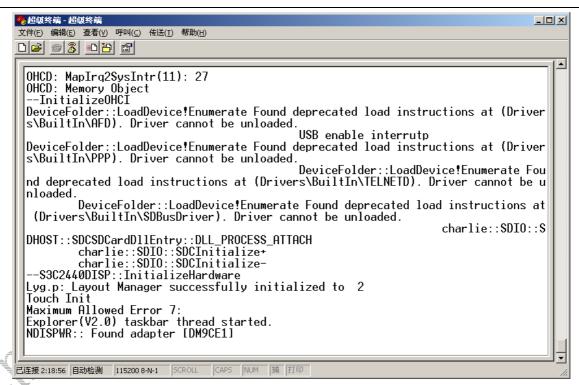
boot begins burning and boot WinCE automatically. This process needs about 4 minutes, as shown in the following diagram:

文件(E) 編辑(E) 査看(Y) 呼叫(C) 传送(I) 帮助(H) 「Diazol and
<pre>*** Downloading BIN image type *** IFTP: Desktop losing ACK, block number = 3669, Ack again IFTP: Desktop losing ACK, block number = 19992, Ack again IFTP: Desktop losing ACK, block number = 23766, Ack again IFTP: Desktop losing ACK, block number = 25512, Ack again IFTP: Desktop losing ACK, block number = 28098, Ack again IFTP: Desktop losing ACK, block number = 35330, Ack again IFTP: Desktop losing ACK, block number = 40675, Ack again IFTP: Desktop losing ACK, block num</pre>
已连接 2:15:36 自动检测 115200 8-N-1 SCROLL CAP5 NUM 捕 打印 //

The following diagram displays the serial port information when WinCE starts boot:







The introduction of u-boot is finished here. You can consult <u>www.embedsky.net/bbs</u> if you have any more questions

3.5 Burning system

Caution: Make sure that u-boot has been burned to Flash and works well before you try to burn operating system. If there is no u-boot in Flash, please consult "3.2 \ddagger " and "3.3 \ddagger " to burn u-boot first.

u-boot has been burned to Flash in factory, so there is no need to burn u-boot again.

3.5.1 Burning Linux OS

Please consult Step 3, Step 5 or Step 6 of "3.4.3 节" when you try to burn Linux OS.

3.5.2 Burning WinCE OS

Please consult Step 2 or Step 4 of "3.4.3 节" when you try to burn WinCE OS.

3. 5. 3 Burning TQ2440-Test_xxxx.bin file

Please consult "3.2 节" when burning. The start address is 0 at Nor Flash.

If you want to burn "TQ2440-Test_xxxx.bin" to Nand Flash, you could choose to enter "a" in u-boot one-key menu, or execute the TFTP command of u-boot. The command is: tftp 0x30000000 TQ2440_Test_xxxx.bin; nand



erase 0x0 0x92000; nand write.jffs2 0x30000000 0x0 \$(filesize)

Caution: the value "0x92000" needs to be modified according to the actual size of "TQ2440-Test_xxxx.bin"

3.6 Linux experiment

3.6.1 Experiment of program termination

2 method to terminate a program:

Method 1: In terminal console, press "Ctrl" and hold, and then press "C".

Method 2: If the program is running background, you can first execute the command "ps" to search the process ID of the program, and then execute the command "kill" to terminal it.

3. 6. 2 Experiment of program auto-run configuration

You can configure the booting script or other system settings to set auto-run. The booting script is under the directory "/etc/init.d/rcS", the settings is shown as the following contents (the following contents might have some tiny differences with the actual ones):

#! /bin/sh

Í

PATH=/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/bi	in: #Set the default execution path
runlevel=S	#User level. Here is: Single
prevlevel=N	
umask 022	
export PATH runlevel prevlevel	
#	
# Trap CTRL-C &c only in this shell so we	can interrupt subprocesses.
#	
trap ":" INT QUIT TSTP	
hwclock –s	#Synchronize Linux clock and RT
mknod /dev/pts/0 c 136 0	
ln -s /dev/v4l/video0 /dev/video0	
ln -s /dev/fb/0 /dev/fb0	#The symbol link of FrameBuffer
ln -s /dev/vc/0 /dev/tty1	
ln -s /dev/sound/dsp /dev/dsp	#The symbol link of sound device
ln -s /dev/sound/mixer /dev/mixer	
ln -s /dev/input/tsraw0 /dev/h3600_tsraw	
#Set the common temporary directory	
mount there have been a	

mount -t proc none /proc





mount -t tmpfs none /tmp mount -t tmpfs none /var

mkdir -p /var/lib mkdir -p /var/run mkdir -p /var/log

/etc/rc.d/init.d/netd start	#Start telnet/ftp service
/etc/rc.d/init.d/httpd start	#Start web server
/etc/rc.d/init.d/leds start	#Start LED
ifconfig lo 127.0.0.1	#IP address of local loop device

route add default gw 192.168.1.2 ifconfig eth0 192.168.1.6 up

#Set and enable the IP address of network card

/bin/qtopia &

#Run Qt/Embeded after start-up

#Set gateway

/bin/hostname -F /etc/sysconfig/HOSTNAME

3. $\overline{6}$. **3** Experiment of setting and saving system real-time clock

Time in Linux is modified by executing the command "date". The command "hwclock" is used to synchronize S3C2440 internal clock and Linux system clock:

≻ The format of "date" command: month date hour minute year. For example 2007-08-28 12:30, the command is "date -s 082812302007".

Execute the command "hwclock -w" to save the time to S3C2440 internal RTC.

Execute the command "hwclock -s" when Linux start-up to recover the RTC time. You can also add the command to the directory "etc/init.d/rcS" to set auto-run when start-up.

Caution: We have added the command "hwclock -s" to the file "rcS" in factory.

3. 6. 4 Experiment of USB mobile storage device test

The device file corresponding to USB mobile storage device in Linux is "/dev/scsi/host(N-1)/bus0/target0/lun0/part1". Caution: the red N indicates the times you insert USB mobile storage device.

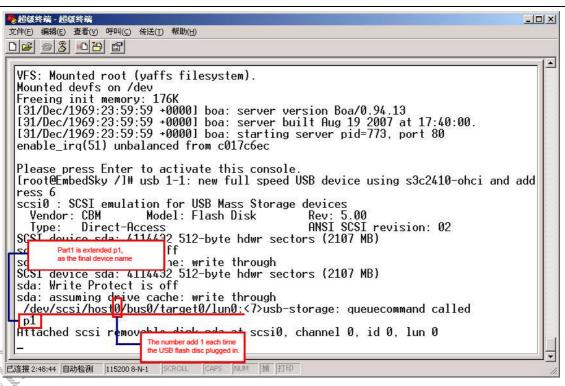
After the USB mobile storage device is inserted, the hyper-terminal appears prompt information, as shown in the following diagram. According to the prompt, you could mount the USB mobile storage device under the directory "/mnt":

(caution: The number in green frame indicates how many times the USB mobile storage device has been inserted. The number will increase by 1 automatically per insertion)

The prompt of U-disk insertion is shown in the following diagram:







Execute the mount command:

Í

#mount /dev/scsi/host(N-1)/bus0/target0/lun0/part1 /mnt

🍫 超级终端 - 超级终端	
文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
sda: assuming drive cache the USB flash disc	
sda: assuming drive cache interest name and the second	
view scarring of busin tai getor funo. (viusb storage, duedecommand carred	
p1 Attached scsi removable disk sda at scsi@ <u>chappel @</u> The first time mount the USB flash disc.	
Attached scsi removable disk sda at scsi0 <mark>, channel 0, the USB flash disc.</mark>	
[root@EmbedSky /]# mount /dev/scsi/host0/bus0/target0/lun0/part1 /mnt/]	
[root@EmbedSky /]# umount /mnt/ [root@EmbedSky /]# usb 1-1: USB_disce UsB Mash disc ddress 6	
I POOLEEMDEdSky / J# USD I-1: USB also usb field usb fie	
usb 1-1: new full speed USB device using s3c2410-ohci and address 7	
scsi1 : SCSI emulation for USB Mass Storage devices	
Type: Direct-Access ANSI SCSI revision: 02	
SCSI device sda: 4114432 512-byte hdwr sectors (2107 MB) sda: Write Protect is off	
sda: assuming drive cache: write through	
SCG: dssuming or ive cache, write through	
SCSI device sda: 4114432 512-bute bdwr sectors (2107 MB) sda: Write Prot ect is The second time plug in the USB flash disc.	
sda: assuming drive cache: write through	
/dev/scsi/hos <mark>11</mark> /bus0/target0/lun0:<7>usb-storage: queuecommand called	
Attached scsi removable disk sda at the use flash disc. 0, id 0, lun 0	
The use flash disc. 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,	
[root@EmbedSky /]#_mount /dev/scsi/host1/bus0/target0/lun0/part1 /mnt/	
[root@EmbedSky /]# _	
Li oo telinbedoky / Jii _	
THE REPORT HEADER IN THE AND A A A A A A A A A A A A A A A A A A	11.

3. 6. 5 Experiment of SD card test

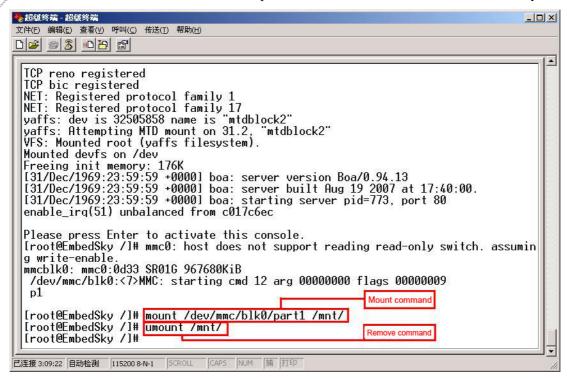
Just like using U-disk, the following information appears after the SD card insertion:





文件(E) 編辑(E) 査看(Y) 呼叫(C) 传送(I) 帮助(E)	
TCP bind hash table entries: 4096 (order: 2, 16384 bytes) TCP: Hash tables configured (established 4096 bind 4096) TCP reno registered TCP bic registered NET: Registered protocol family 1 NET: Registered protocol family 17 yaffs: dev is 32505858 name is "mtdblock2" yaffs: Attempting MTD mount on 31.2, "mtdblock2" VFS: Mounted root (yaffs filesystem). Mounted devfs on /dev Freeing init memory: 176K [31/Dec/1969:23:59:59 +0000] boa: server version Boa/0.94.13 [31/Dec/1969:23:59:59 +0000] boa: server puilt Aug 19 2007 at 17:40:00. [31/Dec/1969:23:59:59 +0000] boa: starting server pid=773, port 80 enable_irg(51) unbalanced from c017c6ec	
Please press Enter to activate this console. Iroot@EmbedSky /]# mmc0: host does not support reading read-only switch. assumin g write-enable. mmcblk0: mmc0:0d33 SR01G 967680KiB /dev/mmc/blk0:<7>MMC: starting cmd 12 arg 00000000 flags 00000009 p1 Part1 is extended p1, as the mounting device name.	
■ 已连接 3:07:10 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

Execute the command "#mount /dev/mmc/blk0/part1 /mnt" to mount SD card under the directory of "/mnt":



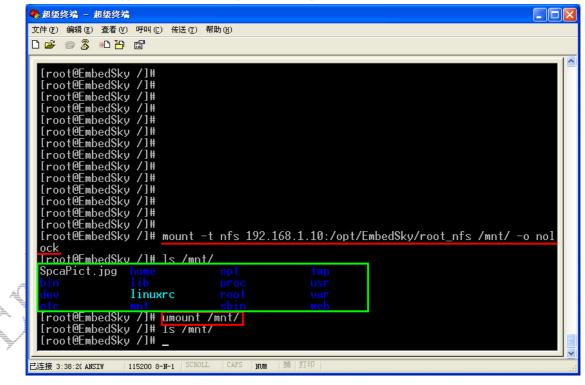
3. 6. 6 Experiment of mounting NFS

Build NFS server before the start this experiment. Set NFS server IP address: 192.168.1.10, and execute the following command to mount:



"mount -t nfs 192.168.1.10:/opt/EmbedSky/root_nfs /mnt -o nolock"

Select the NFS file under the directory "/opt/EmbedSky/root_nfs" in PC "192.168.1.10" as root file system. The following diagram appears after mount operation completes:



3. 6. 7 Experiment of USB camera capturing screen test

TQ2440 Development Board supports most USB cameras sold in market, for example the USB camera produced by Smics. After the camera is inserted to USB interface, the following information appears, and you can find the device name under the directory "/dev/v4l/":

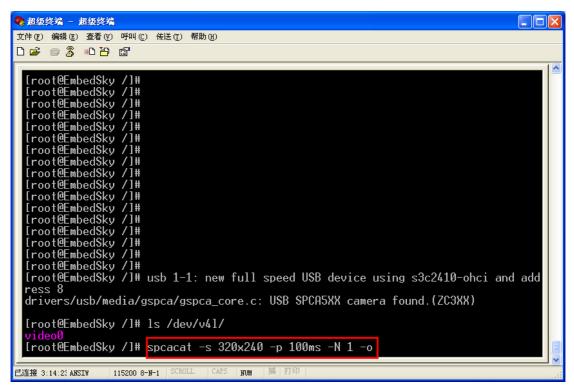


-N 3



超级终端 - 超级终端	
件で迎編器で回道者で回呼叫で回信送で回報助任日	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
Iroot@EmbedSky_/1#	
[root@EmbedSky /]# usb 1-1: new full speed USB device using s3c2410-ohci and	add
ress 8	
drivers/usb/media/gspca/gspca_core.c: USB SPCA5XX camera found.(ZC3XX)	
froot@E mbedSky /]# ls /dev/v41/	
froot@EmbedSky /]# _	
车接 3:13:2(ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	

Use the software sapcacat you can get images captured by camera. Execute the command: spcacat -p 100ms



Command illustration: "-s" represents solution; "-p" represents the time interval between two captures; "-N" represents how many images to capture; "-o" represents over writing the former images named "SpcaPict.jpg" and saving the new one.





28. 超级终端 - 超级终端 - 超级终端 - 回X
文件(2)編輯(2) 査看(2) 呼叫(2) 传送(2) 帮助(2) □ 22 - 22 - 32
Available Resolutions width 192 heigth 144 Available Resolutions width 176 heigth 144 Format asked 15 check 8 VIDIOCSPICT brightnes=32768 hue=0 color=0 contrast=32768 whiteness=0depth=12 pal ette=15 VIDIOCGPICT brightnes=32768 hue=0 color=0 contrast=32768 whiteness=0depth=12 pal ette=15 grabbing method default MMAP asked VIDIOCGMBUF size 2457616 frames 2 offets[0]=0 offsets[1]=1228808 Waiting for Incoming Events. CTrl_c to stop !!!! picture jpeg SpcaPict.jpg GRABBER going out !!!!! unmapping frame buffer close video_device freeing output buffer 0 freeing output buffer 2 freeing output buffer 3 Ircont@EmbedSky /]# 1s SpcaPict.jpg Lin proc usr he picture generated Differ bedSky /]# _
cute the command "spcacat –h" to get more information of the parameters:
🗞 超级终端 — 超级终端
文件 (2) 编辑 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助 (3)
GRABBER going out !!!!! unmapping frame buffer close video_device freeing output buffer 0 freeing output buffer 1 freeing output buffer 3 Iroot@EmbedSky /l# 1s SpcaPict.jpg home opt tmp bin lib proc usr dev linuxrc root var etc mnt spcacat -h usage: cdse [-h -d -g -s -P -p -N -o] -h print this message -d /dev/videoX use videoX device -g use read method for grab instead mmap -f video format default yuv others options are r16 r24 r32 yuv jpg -s widthxheight use specified input size -P /dev/partportX use partportX device -p x ms take a picture every x ms minimum is set to 50ms -N take a N pictures every p ms and stop -o overwrite picture, each picture come with the same name SpacPict.jpg Iroot@EmbedSky /l#

3. 6. 8 Experiment of sound card test

115200 8-N-1 SCROLL

已连接 3:17:48 ANSIW

1

madplay is a MP3 player running on console, with various control modes. Execute the command "madplay -h" to get more help information:

NMM 捕 打印



ź



) 超级终端 - 超级终端 (件(F) 编辑(E) 查看(V)		BHH (V)		
1 🖙 💮 🔏 💷 🤂		and (T)		
	Ež		 	
	1.2			🗠
[root@EmbedSky	/]#			
[root@EmbedSky	/]#			
[root@EmbedSky [root@EmbedSky	/]#			
[root@EmbedSky	/]#			
lroot@EmbedSky	/]#			
[root@EmbedSky	/]#			
[root@EmbedSky	/]#			
[root@EmbedSky [root@EmbedSky	/]#			
[root@EmbedSky	/1#			
[root@EmbedSky	/1#			
[root@EmbedSku	/1#			
[root@EmbedSky [root@EmbedSky	/1#			
[root@EmbedSky	/1#			
[root@EmbedSky	/1#			
	/1#			
[root@EmbedSky [root@EmbedSky	/1#			
[root@EmbedSky	/]#			
[root@EmbedSky	/]#			
I FOUL@Embed3Ky	/]#			
[root@EmbedSky [root@EmbedSky	/]#			
LFOOL@EMDedSKy	/]#			
[root@EmbedSky				
[root@EmbedSky	/]#			
[root@EmbedSky [root@EmbedSky				
lroot@EmbedSky	/I# madplay	-h		

You could execute the command "madplay xxxx.mp3" to play music in default mode. We provide a test music "madplay /root/Documents/Test.mp3" under the directory "/root/Documents/"

🍣 超级终端 - 超级终端	
文件 (2) 编辑 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助 (1)	
다 🛥 📨 🕈 🗈 🎦	
no-tty-control	disable keyboard controls
Miscellaneous: -V,version license -h,help	display version number and exit show copyright/license message and exit display this help and exit
aiff Audio IFF, [16-bit] wave Microsoft RIFF/WAVE, snd Sun/NeXT audio, 8-bi raw binary [16-bit] host hex ASCII hexadecimal [2 null no output (decode on [root@EmbedSky /]# madplay /ro	[16-bit] PCM (*.wav) t ISDN mu-law (*.au, *.snd) -endian linear PCM 4-bit] linear PCM lv) ot/Documents/Test.mp3
MPEG Hudio Decoder 0.15.0 (bet Title: EYES ON ME Artist: 王菲 Album: 天籁村 Year: 2000 Genre: Pop Comment: http://tdk.126.com	a) - Copyright (C) 2000-2003 Robert Leslie et al.
	PS NM 補 打印

3. 6. 9 Experiment of files transmission with PC via serial port

After log on OS via serial port interface, you could execute the command "rz" and "sz" to transmit files with

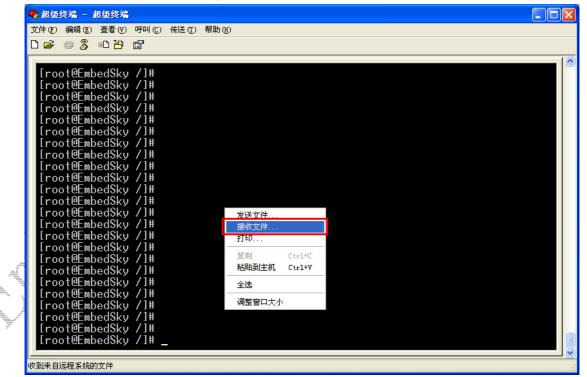


PC via serial port. ≝

The operation is introduced as follows 操作如下:

1), send files to PC:

Step1, click mouse right button in hyper-terminal interface, and select "接收文件":



Step2, the interface "接收文件" pops up. Configure the interface as the following diagram and click "接收" to continue:

■ 接收文件		? 🛛
在下列文件夹中的	故置收到的文件 (₽)∶	
D:\SKY2440		浏览(B)
使用接收协议(U)) <u>:</u>	
Zmodem 与崩溃的	反复	~
	接收(23) 关闭(2) 取消

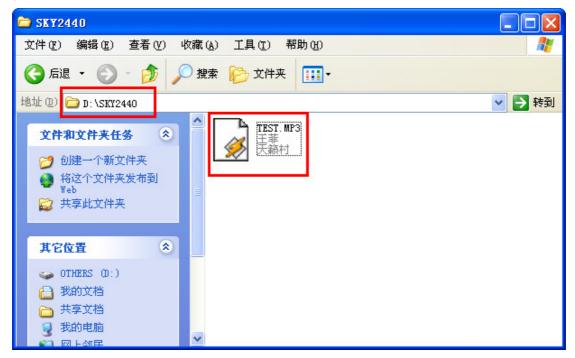
Stpe3, enter the command "sz /root/Documents/Test.mp3" in hyper-terminal to start transmitting "Test.mp3" under the directory "/root/Documents/" to PC:





		超级终端	P¥UU (C)	传送 (T)		
				18/2 (1)	(1) (1) (1)	
	<i>₩</i> ₽					
[ma	a + QE al	a delas	/1#			
		oedSky oedSky	/1#			
		edSky				
Iro	ot@Emik ot@Emik	odSku	/1#	为 超级约	S端 接收 Zmodem 与崩溃恢复 文件	
Iro	ot@Emik	oedSký oedSky	71#			
Iro	ot@Emb	edSky	/1#	正在接收:	TEST. MP3	
		edSky				
Iro	ot@Emb	edSkv	/1#	存储为:	D:\SKY2440\TEST.MP3 文件数: 1 / 1	
Iro	ot@Emb	oedSky oedSky	/]#	F)を事件・	正在接收 重试次数:	
Iro	ot@Emk	bedSky	/]#			
Iro	ot@Fmŀ	bedSku	/1#	状态:	正在接收	
[ro	ot@Emb	oedSky oedSky	/]#			
[ro	ot@Emk	bedSky	/]#	文件:	304K / 5376K	
[ro	ot@Emk	bedSkv	/]#	JUIT.	I JOAK / JOIN	
Iro	ot@Emb	oedSky oedSky	/]#	已用:	00:00:27 剩余: 00:07:37 吞吐量: 11361 cps	
lro	ot@Emb	bedSky	/]#			
Iro	oteemt	edSky	/]#		取消 跳过文件 (S) cps/bps (C)	
Iro	oteemt	edSky	/]#			
Iro	ot@Emb	oedSky oedSky	/]#			
Iro	OTUELMI A + GE - L	oedSky oedSky	/]# /]#			
F		101				
LT U	ot@Emb	odsku	/1#	7 /200	at/Documents/Test_mp3	
2 0	DADDDD	ANNANA	1112	52 7100	ot/Documents/Test.mp3	
. 0	000000	00000				
1)***	:25:42 ANS		15200 8-1	I-1 SCROI	LL CAPS MMM 捕 打印	

Step4, the file is automatically saved to the directory that you have just set after the transmission is over:



2), transmit files to platform:

1

Step1, enter the command "rz" in hyper-terminal to start receiving files from PC:



Í



超级终端 - 超级终端	
沈件(12) 编辑(12) 査看(12) 呼叫(12) 传送(12) 帮助(♪ ~~? ~~~ ?? ?? ~~?!	Ð
) 🖨 💮 🍒 🗈 🎦 😭	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSkv /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# rz	0000000-50
? waiting to receive.**1B000	0000230630
遊接 0:16:0f ANSIW 115200 8-N-1 SCROLL 2, click mouse right button in hyp	CAPS MM
	er terminal and select "发送文件":
2, click mouse right button in hyp 加级终端 - 超级终端	er terminal and select "发送文件":
2, click mouse right button in hyp 起级终端 - 超级终端 这件论 编辑 © 查看 ⑦ 呼叫 © 传送 ⑦ 帮助] 译 會 多 = □ 语 音	er terminal and select "发送文件":
2, click mouse right button in hyp ##级终端 - 超级终端 2件 또 編輯 또 출품 안 呼叫 또 传送 또 帮助] 译 當 중 == 관 협 [root@EmbedSky /]#	er terminal and select "发送文件":
2, click mouse right button in hyp <u>思级终端 - 超级终端</u> 文件で 編輯で 查看で 呼叫で 传送で 帮助 こ	er terminal and select "发送文件":
2, click mouse right button in hyp ² 2 2 2 2 2 2 2 2 2 2 2 2 2 2	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 文件で 編輯で 查看 ① 呼叫 © 传送 ① 帮助 こ	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 文件で 編録で 查看 ① 呼叫で 传送で 帮助 こ	er terminal and select "发送文件":
2, click mouse right button in hyp 2, click mouse right button	er terminal and select "发送文件":
2, click mouse right button in hyp a 超级终端 - 超级终端 文件で 編録で 查看 の 呼叫で 传送で 帮助 ご 学 会 意 中 音 留 [root@EmbedSky /]# [root@EmbedSky /]#	er terminal and select "发送文件":
2, click mouse right button in hyp #板袋袋 - 超级终端 文件で 編録で 查看 の 呼叫で 传送で 帮助 ご 学 会 意 中 子 留 「root@EmbedSky /]# [root@EmbedSky /]#	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 文件 (2) 編輯 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助) (2) 字 ② 章 印 召 雷 [root@EmbedSky /]# [root@EmbedSky /]#	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 文件 (2) 編輯 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助)) 译 ② ③ 电 召 昏 [root@EmbedSky /]# [root@EmbedSky /]#	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 文件 (2) 编辑 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助)) 译 ② ③ 电 召 留 [root@EmbedSky /]# [root@EmbedSky /]#	er terminal and select "发送文件":
2, click mouse right button in hyp ##\$9\$ ### ##\$10 #### ##\$20 ### ## ## ## ## ## ## ## ## ##	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 定性 ① 编辑 ② 查看 ② 呼叫 ② 传送 ① 帮助 ② 當 ② ■ ○ 沪 當 「root@EmbedSky /]# [root@EmbedSky /]#	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 文件で 編程で 查看 の 呼叫で 传送で 帮助 こ	er terminal and select "发送文件":
2, click mouse right button in hyp 超级终端 - 超级终端 文件 (2) 編輯 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助) (2) 字 (2) 意 (1) 合 (1) 字 (2) 意 (1) 合 (1) 合 (1) (1) (1) (1) (1) (1) (1) (1)	er terminal and select "发送文件":
2, click mouse right button in hyp 2. click mouse right button in hyp 2. adgkg - 超级终端 本住 (2) 編輯 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助) 2. add (2) 和 (2) \Lambda	er terminal and select "发送文件":
2, click mouse right button in hyp ■超级终端 - 超级终端 CHTE 編輯 2 查看 9 呼叫 2 传送 1 帮助 CHTE 編輯 2 查看 9 呼叫 2 传送 1 帮助 CHTE 編輯 2 查看 9 呼叫 2 作送 1 帮助 CHTE 編輯 2 查看 9 呼叫 2 作送 1 帮助 CHTE 編輯 2 查 2 = □ 2 章 I root 0EmbedSky / 1 # I root 0EmbedSky / 1 #	er terminal and select "发送文件":
2, click mouse right button in hyp 2. click mouse right button in hyp 2. adgkg - 超级终端 本住 (2) 編輯 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助) 2. add (2) 和 (2) \Lambda	er terminal and select "发送文件":

Step3, click the button "浏览" in the pop-up interface "发送文件" and locate the file for transmission, as shown in the following diagram:

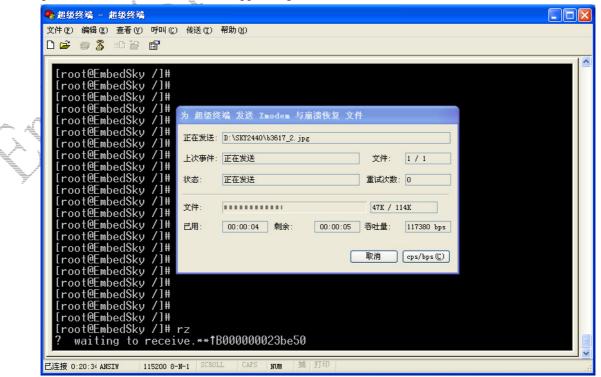


_





Step4, click the button "发送" in the upper diagram to start transmission:



Step5, the name of the file transmitted is in the following red frame:





文件 (E) 编辑 (E) 查看 (Y) 呼叫 (C) 传送 (T)	帮助(H)		
D 🚔 📨 🍒 🗈 🎦 😭			
-			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]# [root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]# [root@EmbedSky /]#			
[root@EmbedSkv /]#			
[root@EmbedSky /]# [root@EmbedSky /]#			
lroot@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]# [root@EmbedSky /]# rz			
2 [root@EmbodSky /]# r2	RAAAAAAAAAAA	50	
[root@EmbedSky /]# .ssi	D000000020D6	:00	
? [root@EmbedSky /]# .**1 [root@EmbedSky /]# 1s SpcaPict.ing_etc			
b3617_2.jpg home			
bin lib			
dev linuxrc			
[root@EmbedSky /]#			
LFOOT@EmbedSky 71#			

3. 6. 10 Experiment of screen capture

Execute the command "snapshot" to capture screen and save the image into the png format.

The command: snapshot PIC.png

文件 (E) 編辑 (E) 查看 (V) 呼叫 (E) 传送 (E) 帮助 (E) □ ☞ ◎ ⑤ ==□ 凸 留
<pre>[root@EmbedSky /]# [root@EmbedSky /]#</pre>

The captured image is under the root directory:





文件 (2) 编辑 (2) 查看 (2) 呼叫 (2) 传送 (2)	帮助(H)		
D 🛩 🍘 🍒 🗈 🎦 😭			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]# [root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSkv /]#			
[root@EmbedSky /]# [root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]# [root@EmbedSky /]#			
[root@EmbedSky /]#			
[root@EmbedSky /]# snapsh	ot PTC.png		
Converting image from 16 Now writing PNG file [root@EmbedSky /]# ls			
Now writing PNG file			
[root@EmbedSky /]# 1s			
PIC.png dev	linuxrc		
SpcaPict.jpg etc b3617_2.jpg home			
bip lib			
bin lib [root@EmbedSky /]# _			
LI OUTCEMBEGOKY 7 Im _			

3. 6. 11 Experiment of user LED test

1), LED server

Í

After system start-up, the user LED service application "/etc/rc.d/init.d/leds" executes automatically. The application calls a script of led-player to create a pipe file under the directory "/tmp". The user LED flash mode varies according to the parameters send to this pipe:

 \geq #echo 0 0.5 > /tmp/led-control

The 4 user LEDs run marquee at 0.5-second interval after the command running:





文件(2) 編輯(2) 查看(2) 呼叫(2) 保険(2) 帮助(3) □ ☞ ● ③ □ 凸 □ 「root@EmbedSky /1# 「root@EmbedSky /1# 「root@EmbedS	🍣 超级终端 - 超级终端	
<pre>[root@EmbedSky /]# [root@EmbedSky /]#</pre>		传送 (I) 帮助 (H)
	[root@EmbedSk; [root@EmbedSk;]]	ype is 0, period is 0.500000

Í

The 4 user LEDs run accumulator at 0.5-second interval after the command running:

超级终端 一 超级终端
(件 (2) 編辑 (2) 査看 (2) 呼叫 (2) 传送 (2) 帮助 (4)
<pre>[root@EmbedSky /]# [root@EmbedSky /]# [root@Em</pre>

#/etc/rc.d/init.d/leds stop

The 4 user LEDs stop flashing after the command running:



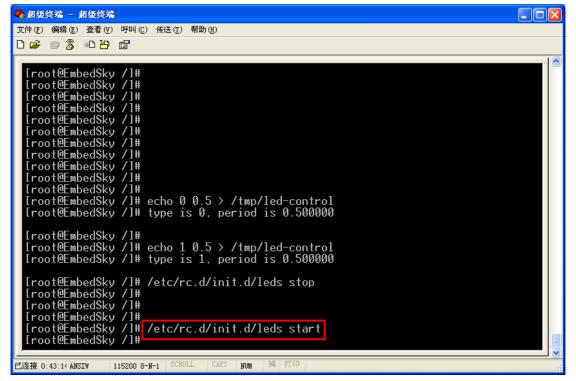


2014 @ 编辑 @ 查看 @ 呼叫 @ 传送 ① 帮助 创] 🕝 🥥 🔉 💷 🎦 📴	
) 🖆 🍘 🌋 🗈 🎦 🗃	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[rootGembedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# echo 0 0.5 > /tmp/led-control	
[root@EmbedSky /]# type is 0, period is 0.500000	
[root@EmbedSky /]#	
[root@EmbedSky /]# echo 1 0.5 > /tmp/led-control	
[root@EmbedSky /]# type is 1, period is 0.500000	
[root@EmbedSky /]# /etc/rc.d/init.d/leds stop [root@EmbedSky /]#	
连接 0:42:42 ANSIN 115200 8-N-1 SCROLL CAPS NOM 描 打印	

► #/etc/rc.d/init.d/leds start

A

The 4 user LEDs re-flash after the command running:



2), control LED separately

The application "/sbin/leds" can control LED separately. You must stop led-player before run this application. Execute the command "/etc/rc.d/init.d/leds stop" to stop led-player.

"leds" usage method:





文件(E) 編辑(E) 査者(V)) 呼叫 @) 传送 @) 帮助 @)	
ነጅ 🛯 🗞 🕛 🗃	6	
	/ /]# / /]#	
	y/]# y/]# echo 1 0.5 > /tmp/led-control y/]# type is 1, period is 0.500000	
[root@EmbedSky [root@EmbedSky [root@EmbedSky [root@EmbedSky [root@EmbedSky	y /]# y /]# y /]# /etc/rc.d/init.d/leds start y /]# /etc/rc.d/init.d/leds stop	
[root@EmbedSky [root@EmbedSky [root@EmbedSky [root@EmbedSky [root@EmbedSky Usage: leds le [root@EmbedSky	//]# //]# //]# leds ed no 011	

"led_no" is the LED sequence number (0, 1, 2, 3); The value "0" and "1" represent operation switching off and switching on.

For example "leds 3 1" means switch on LED3.

3. 6. 12 Experiment of user keyboard test

Mount user keyboard driver first:



Enter "buttons" to start keyboard test. The following diagram displays the responses when pressing the 4 buttons separately:





▶ 超级终端 → 超级终端 文件 使) 編輯 使) 查看 使) 呼叫 使) 传送 ① 帮助 ℓℓ)	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]# [root@EmbedSky /]# buttons	
<u>[root@EmbedSky_</u>]# buttons	
Type: 1 Code: 1 Type: 0 Code: 0	
Tune: 1 Code: 1	
Type: 0 Code: 0	
Type: 1 Code: 2	
Type: 0 Code: 0	
Type: 1 Code: 2 Type: 0 Code: 0	
Type: 1 Code: 3	
Type: 0 Code: 0	
Type: 1 Code: 3 Key 3	
Type: 0 Code: 0 Type: 1 Code: 4	
Type: 0 Code: 0	
Type: 1 Code: 4 Key 4	
Type: 0 Code: 0	
[root@EmbedSky /]# _	
法连接 1:16:01 ANSIN 115200 8-N-1 SCROLL CAPS NUM 捕 打印	

3. 6. 13 Experiment of log in BBS in telnet

telnet is a kind of commonly used remote login tool. We can use telnet to log in other telnet servers from the platform. If the platform is connected to internet, you could use telnet command to log in external BBS.

Use telnet to log in BBS "水木清华":

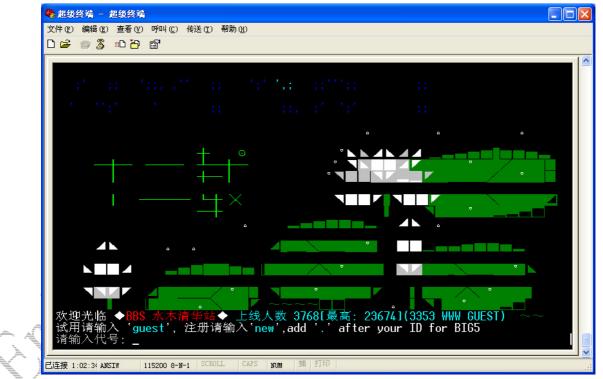
-

🗞 超级终端 - 超级终端	
文件 (2) 编辑 (2) 查看 (Y) 呼叫 (C) 传送 (I) 帮助 (H)	
요 🚔 🐵 🐉 📫 🎦	
[root@EmbedSky /]#	
[root@EmbedSky /]# [root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]#	
[root@EmbedSky /]# telnet 166.111.8_238	



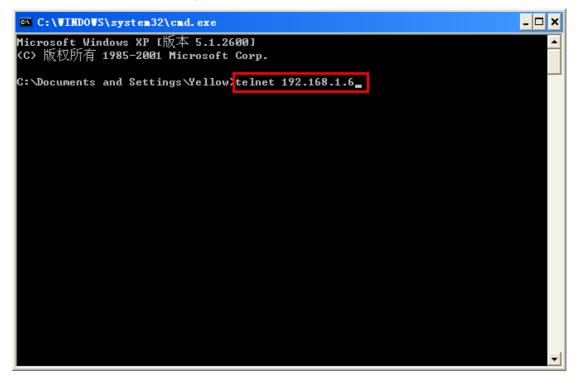


Log in successfully:



3. 6. 14 Experiment of remote login platform by telnet

The user can directly log in platform after system starts up. Enter the command "telnet 192.168.1.6" in command window in Windows OS, and press return-key to continue:





Enter "root" in the following log-in interface to enter into the system:

EmbedSky logi				
[root@EmbedSk PIC.png	(y / J# IS dev	linuxrc		
SpcaPict.jpg		mnt		
b3617_2.jpg				
bin				
[root@EmbedS}				

3. 6. 15 Experiment of FTP remote file transmission

We can use the ftp application contained in Linux or Windows to log in remote host and transmit file, if the remote host support ftp service and the authority is available. TQ2440 provides ftp application and ftp service. Here we make a test to log in platform from PC command window and send files to the platform.

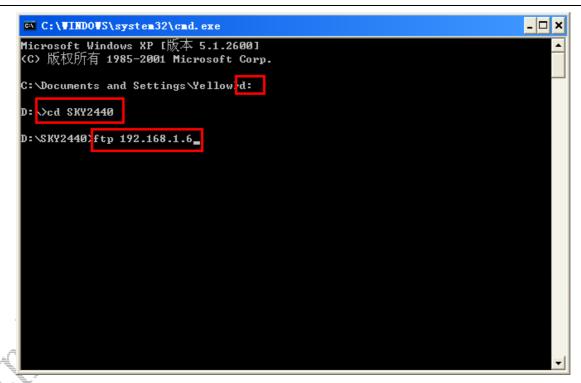
Caution: Make sure the transmission file is under the same directory with ftp and it is available. Here we use the file "PPMM.jpg" for transmission.

After the transmission is over, the file "PPMM.jpg" is added to the directory "/home/sky/".

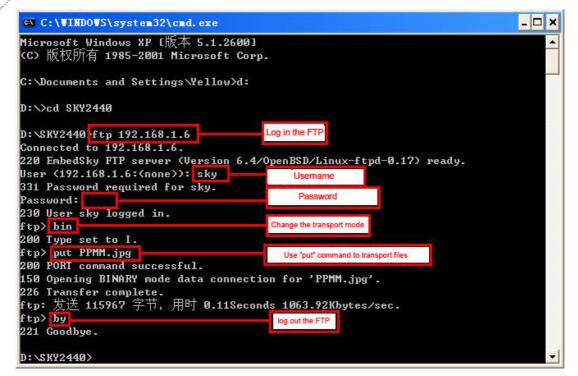
Enter "ftp 192.168.1.6" in command window and press return-key to continue:







Operating as the following diagram:



Enter the command "ls /home/sky/", you can find the file "PPMM.jpg"





3. 6. 16 Experiment of Web server test

Í

The Web server runs automatically after system start-up. The user can use webpage explorer to access the webpage of Web server based on platform. Enter "192.168.1.6" in address table and press return-key. Then you can access the following page:





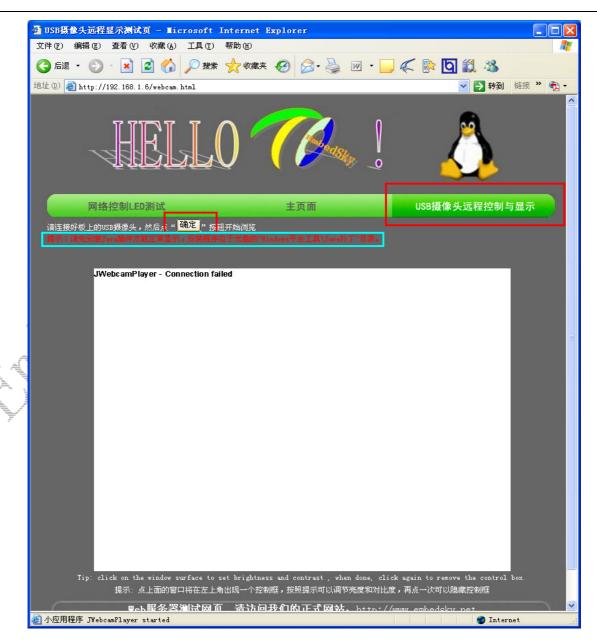
3. 6. 17 Experiment of USB camera remote control

Click the option "USB 摄像头远程控制与显示" in the upper webpage to access USB camera test page:

(caution: Make sure the Jave patch has been installed successfully which is under the directory "Windows 平 台工具/Java 补丁". And it is suggested to use IE explorer provided by Windows, otherwise the experiment will probably fail)







Click "确定" in the following page:

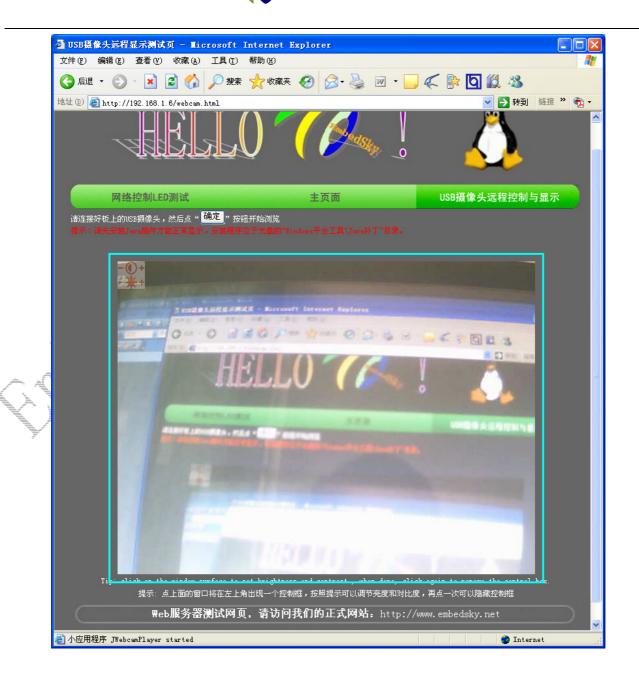
1

🗿 设置wemcam结果 - Microsoft Internet Explorer		
文件 (E) 编辑 (E) 查看 (Y) 收藏 (A) 工具 (E) 帮助 (H)		-
③ 后退 • ③ · ▲ 🛃 🏠 🔎 搜索 🌟 收藏夹 𝚱 🙆 • 🎍 🔟 • 🗍 🌾 除 🔯 🎎 烙		
地址 (1) 🕘 http://192.168.1.6/webcam.cgi?submit=%C8%B7%B6%A8 💙 🔁 转到	链接 » 🤅	1 -
WEBCAM设置已经提交 返回上一页		~
🥘 🛛 🔮 Intern	iet	:

Click "返回" to go back to the former interface. The dynamic images appear in webpage now as shown in the following diagram:







3. 6. 18 Experiment of user LED remote control

Click "网络控制 LED 控制" to access LED control page:





Select the options "类型" and "速率" and click "确定(OK)". Then the user LEDs on platform start flashing.

3.7 WinCE experiment

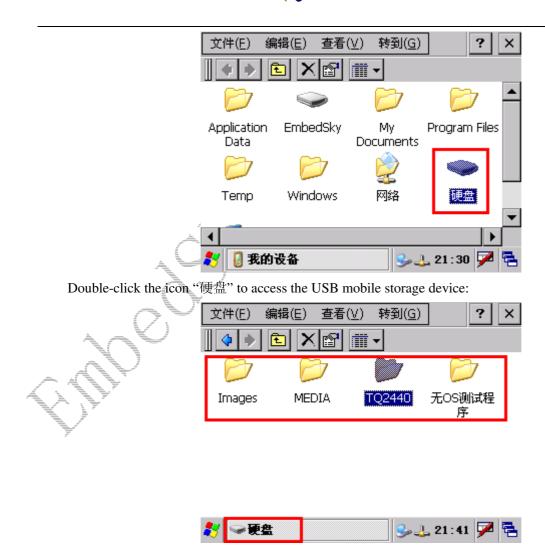
Caution: The IP address of 100M network card is "192.168.1.6".

3.7.1 Experiment of USB mobile storage device test

Like using USB mobile storage device in Windows XP system, insert the USB device into USB Host interface in WinCE system, and the system will load the device automatically after a few seconds. Double-click the icon "我的电脑" on desktop, and open resource manager. You can find the USB mobile storage device disk "硬盘", as shown in the following diagram:



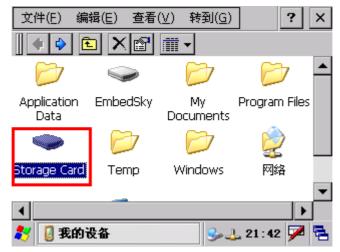




3.7.2 Experiment of SD card test

4

After SD card has been inserted into SD interface on platform, the SD card disk "Storage Card" could be found in resource manager of platform:







Double-click the icon to access the SD card:

[文件(E)	编辑(<u>E</u>) 査	種(⊻) 奪	考到(<u>G</u>)	?	×
	4		9 📺 -			
				A		
	(TQ2440)] 《猴子戏 虎》 巨拍		 訂广告 <u>4</u>	 +仔短裤的 新做法	
		1元// 三折	0,		动印以云	
À		-al				
	<u> </u> 中シ°,					—
	🟹 🥣 St	orage Card		j 🎐 👍 i	21:46 🏓	5

3. 7. 3 Experiment of Flash power failure protection

The free space in Flash could be used as disk which is supported by BSP package in WinCE. This part of Flash can be used for power failure protection, and it exists in WinCE in the form of the directory "EmbedSky".

Test the function of "EmbedSky":

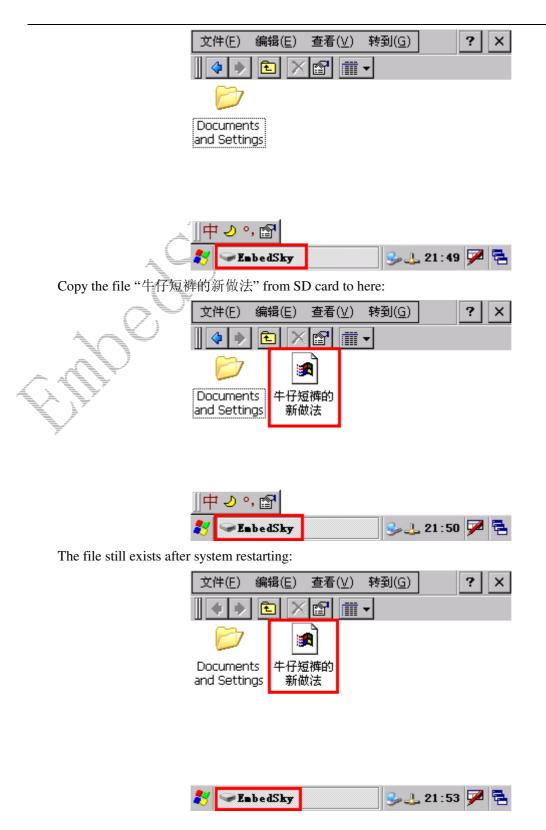
Double-click the icon "我的电脑" and find "EmbedSky":

文件(E) 4	辑(E) 査看()	<u>V</u>) 转到(<u>G</u>)	?	×		
\triangleright		\triangleright	\triangleright			
Application Data	EmbedSky	My Documents	Program Files			
	\triangleright	\triangleright	2			
Storage Card	Temp	Windows	网络			
中						
🐉 🔋 我的	设备	3	L 21 : 47 🏓			

Double-click to access it.







3.7.4 Experiment of player utilization

Í

WinCE supports two kinds of player: The one is "Media Player", and the other is "超级播放器".



1) play MP3 music by Media Player

Double-click the icon "Media Player" on the desktop. Windows Media Player starts to run, as shown in the following diagram:



Start playing:

Í







Media Player also supports the file of WMV format.

2) play MPEG4 movie by "超级播放器"

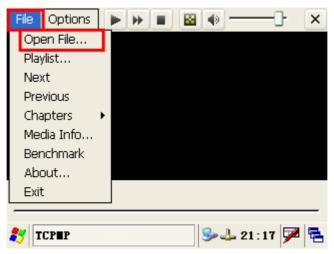
"超级播放器" is a commonly used player in Windows Mobile, similar to "暴风影音". This player supports the format of mpeg2, mov, avi and so on.

Double-click the icon "超级播放器" on the desktop:



Click "File->Open File":

Í







Select the movie file:

Filter	Selecting	All Dir		ок 🗙
\Sto:	rage Card			🔽 Go
Name		Type	Size	Date
*		Up		
6 08		JPG	26 KB	2004-4-10
6]19		JPG	28 KB	2004-4-10
🖆 met	eor shower:	MPG	1373 KB	2001-11-19
🖆 PAG	EANT	MPG	1598 KB	2005-4-24
創愛就	计个字	MP3	5478 KB	2004-7-22
87 TC	P∎P		\$-4	21 : 18 🏓 🖷

The screen capture image is shown in the following diagram:下图是播放中的截图:



3.7.5 Experiment of 100M network card test

Click "开始->设置->网络和拨号连接":

4







The following interface appears:

连接 登 新建连接 上 上 上 上 上 上 上 上 上 上 上 上 上	K? ×
Č	
▲ 【】 【】 【】 【】 【】 】 】 】 】 】 】 】 】 】 】 】 】 】	Se 🕹 19:00 🏴 🖶

Double-click "DM9CE1" to open the configuration interface. We use the default parameters here as the following diagram shows. The user can set the parameters according to the actual situation:

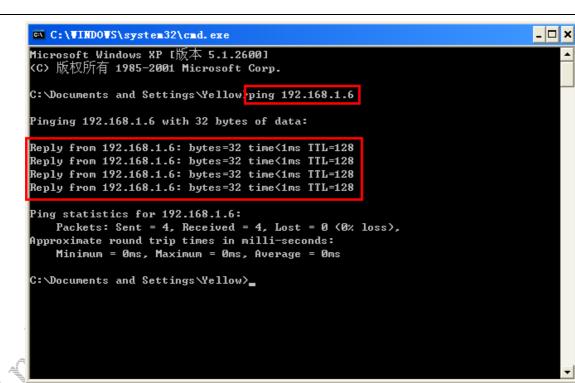
The screen capture image of 100M network card is shown in the following diagram:

连接	*DM9000 ISA Fasi	t Ethern OK 🔀	?	×
3	IP 地址 名称服务器	ł		
新建)	■ 可以为此计算机自动	b分配 IP 地址。		
	○ 通过 DHCP 获得	IP 地址		
	● 指定一个 IP 地址	Ŀ		
	IP 地址:	192.168.1.6		
	子网掩码:	255.255.255.0		
	默认网关:	192.168. 1 . 2		
8	连接	🦻 🕹 19:02	7	4
连接	"DM9000 ISA Fasi	: Ethem OK 🔀	?	×
连接	*DM9000 ISA Fasi IP 地址 名称服务器		?	×
	IP 地址 名称服务器		?	×
9	IP 地址 名称服务器		?	×
9	IP 地址 名称服务器 名称服务器地址		?	×
9	IP 地址 名称服务器 名称服务器地址 主控 DNS:		?	×
9	IP 地址 名称服务器 名称服务器地址 主控 DNS: 辅助 DNS:		?	×
9	IP 地址 名称服务器 名称服务器地址 主控 DNS: 辅助 DNS: 主控 WINS:		?	×

Execute the command "ping" on PC to test the connection state of network:





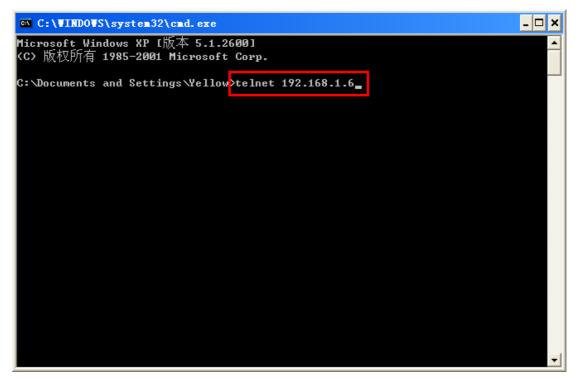


3. 7. 6 Experiment of telnet remote log-in

, i

The telnet service runs automatically after WinCE start-up in TQ2440. Connect the net wire, and the user can log in platform remotely by telnet.

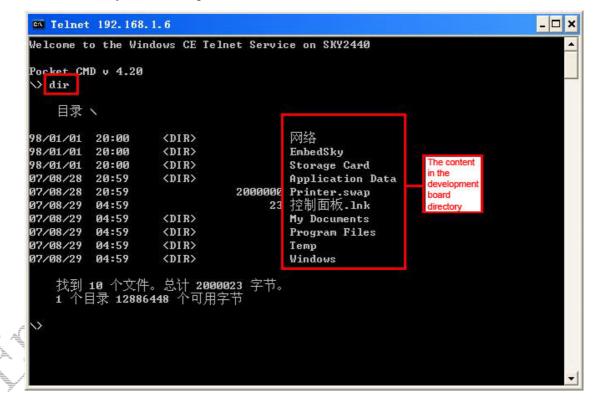
Enter the command "telnet 192.168.1.6" in command window:







Press the return-key to enter into platform console:



Caution: The default IP address in WinCE is "192.168.1.6". No user name and password are needed when log-in.

3.7.7 Experiment of FTP remote file transmission

The FTP service runs automatically after WinCE start-up in TQ2440. Connect the net wire, and the user can log in platform remotely by FTP.

Enter the command "ftp 192.168.1.6" in command window:

Í

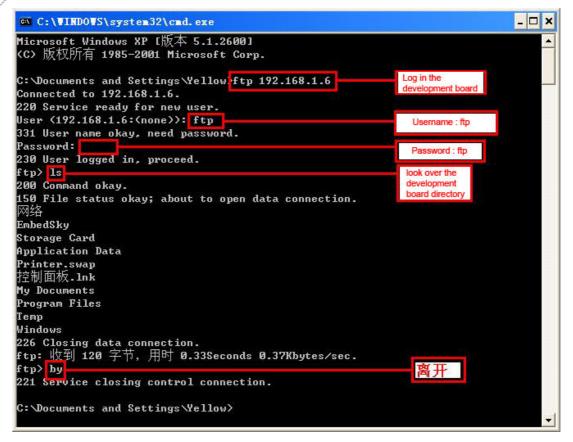


_



C: \TIEDOTS\system32\cmd.exe
Microsoft Windows %P [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.
C: \Documents and Settings\Yellow}ftp 192.168.1.6_

Press the return-key to log in platform. Input user name and password (ftp):



Caution: The default IP address in WinCE is "192.168.1.6". User name and password are both: ftp.



3.7.8 Experiment of Web server test

The http service, namely Web service runs automatically after WinCE start-up in TQ2440. Connect the net wire, and the user can access webpages provided by platform.

Enter "192.168.1.6" in address table and press return-key.



3.7.9 Experiment of touch-screen correction

The correction is needed when touch-screen doesn't work correctly. Connect the USB mouse and click "开始->设置->控制面板":



Find the icon "笔针" and double-click it:





The interface "笔针属性" appears. Click the tag "校准" to enter into correction interface:



Click "再校准" to start correction:

Í

You could also choose to double-click the shortcut "校正触摸", as shown in the following diagram:

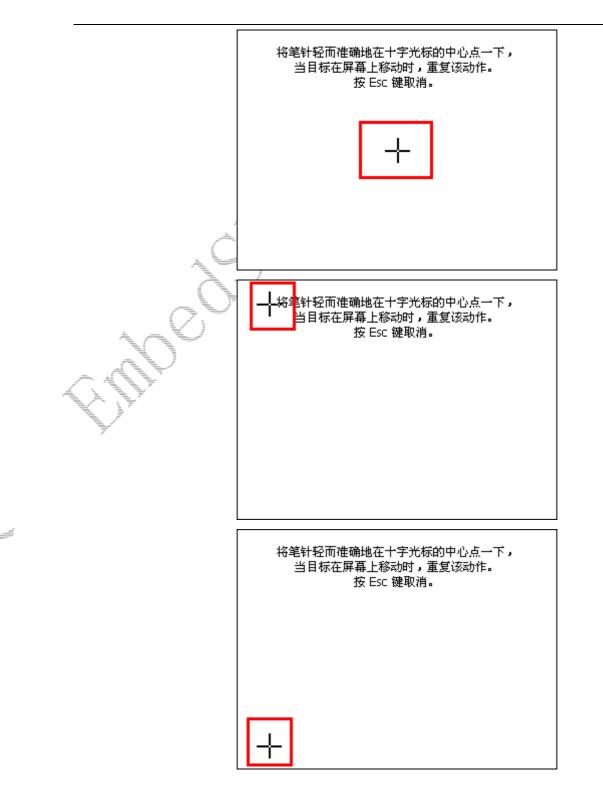


Correction starts:

(caution: There are totally 5 points need to corrected. The center of the cross is the correction point. Touch the 5 correction points precisely for about 1 second each)

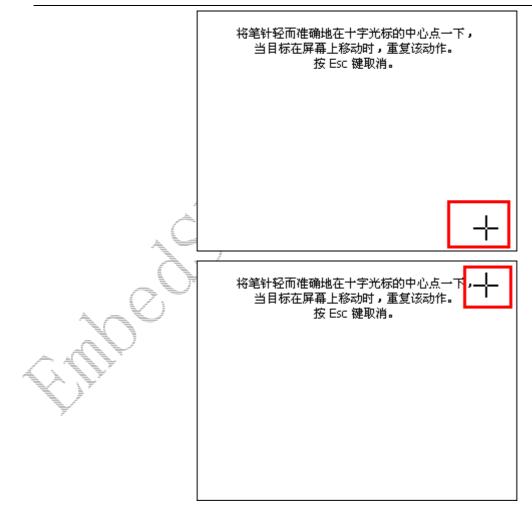




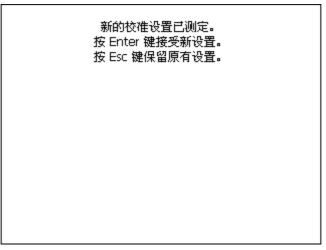








The following diagram appears after correction. Click any place of the touch-screen to continue:



Get back to the following interface. Click "OK" to finish the correction:







(caution: The touch-screen test value contained in BSP package is derived from plenty of actual tests. The user can also consult "section 4.8" to find a better value, and edit it into OS kernel. After the kernel re-compilation, there is no need to do correction any more.)

3.7.10 Experiment of USB camera test

The OS kernel contains USB camera driver which supports Z301P camera chip. Insert the USB camera into the USB interface on platform and start up the platform. After running the test application, the image captured by camera will appear on LED.

The test application has been compiled into OS kernel. It is under the hidden directory "Windows".



Open the directory "Windows" in WinCE:







Deselect the options "不显示隐藏文件和文件夹" and "隐藏受保护的操作系统文件(推荐)" and click "OK" to continue:

文件(E	文件夹选项 2 K	× .	×
地址	高级设置		~
stdsr	🔄 不显示隐藏文件和文件夹	h	^
\$	□ 鹅藏受保护的操作系统文件(推 荐)	1	
240	✔ 不显示文件扩展名		
tlces	浏览文件夹	- m	
	💿 在同一窗口中打开每个文件夹		
viewsr	○ 在不同窗口中打开不同的文件夹	ck	~
🐉 文作	牛夹选項 🈏 🕹 3:2	21 🔛	1 🖷

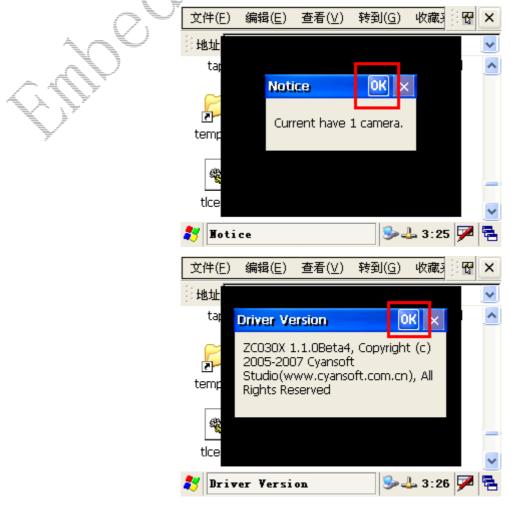
The USB camera test application "Testzc030x" can be found under the directory "Windows":







Double-click the test application. Click "OK" in the following 2 pop-up interfaces:

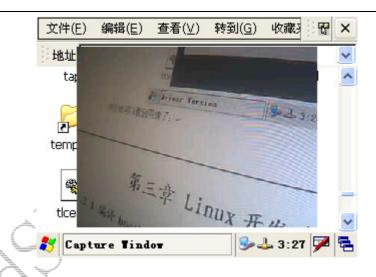


Í

The captured image appears (the manual is in the image):







3. 7. 11 Experiment of image rotation in LCD

Click the shortcut "LCD 旋转测试" on the desktop, as shown in the following diagram:



The following interface appears:

Í

3		、、	
回收:	LCD旋转测试程序 天嵌科	호 OK 코	
C	旋转0°	_	
我的话	旋转270° 旋转90°		
Z Embec	旋转180°	退出員	
🏞 📱	LCD旋转测试程序 😏	🕹 21 : 59 🏓	4

Click the option "旋转 90°". The desktop image rotates by 90 degree, as shown in the following diagram:







270 degree rotation and 180 degree rotation are similar to the upper case.

3.7.12 Experiment of 3 serial ports test



Click the shortcut " ${}^{\oplus}\Box$ 测 试" on the desktop, as shown in the following diagram:

The serial port test application starts to run:

Í





	串口测试程	序 天嵌科技	ок 🗙	*
E	「串口设置-		1 接收区	
	串口号:	COM1 🔽	 ▼	
	波特率:	115200 🔻		
我	数据位:	8 🔻	发送区	
	停止位:	1 🔻	<u>^</u>	
Em	校 验:	无 💌	发送	
	打开串口	1 关闭串口		
- 🐉 🛛	こ 串口測试	【程序 ー 天	😼 🔔 22:38 📝	6

Select COM1 and click "打开串口". The prompt interface appears in the following diagram:

	串口测试程	序 天嵌科技	ок 🗙	
E	「串口设置-		接收区	
	串口号:			
	波特率:	SerialPort OK	· × · · · · · · · · · · · · · · · · · ·	
我	数据位:	打开COM1:成)	5)I X	
	停止位:			
Em	校验:	无 💌	发送	
	打开串口	关闭串口	清除发送区	ST.
*	📮 串口测证	【程序 天	🎭 🔔 22 : 39 🗭	2 🔁

Use serial port tool in PC to send data to platform. We enter "TQ2440" in PC and send the string. Then we find that the platform has received the string:

	串口测试程	序 天嵌科技	ок ×	*
E	「串口设置一		1 接收区	
	串口号:	COM1 🔻	TQ2440	
	波特率:	115200 🔽	清除接收区	
我	数据位:	8 🔻	发送区	
	停止位:	1 💌	×	
Em	校验:	无 💌	发送	
	打开串口	关闭串口		1
87	📮 串口测试	程序 天	. 🎐 🔔 22 : 40 💆	Ę

The information entered in platform can also be send to PC. We enter "<u>www.embedsky.net</u>" in platform and send the string. The string is received by PC, as shown in the following diagram:

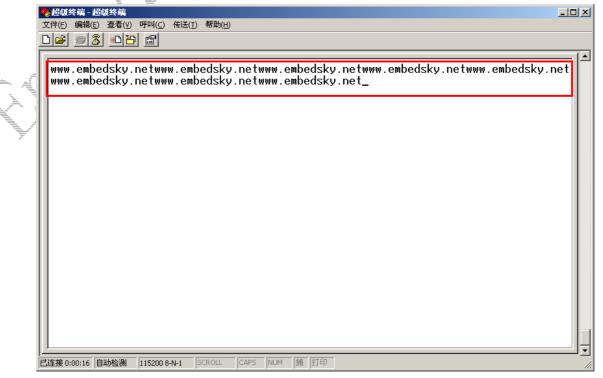
In the platform:





	串口测试程)	亨 天嵌科技	ок 🗙	*
E	「串口设置一		接收区	1
	串口号:	COM1 🔻	TQ2440 🔺	
	波特率:	115200 🔻	 清除接收区	T
我	数据位:	8 🔻	发送区	M
	停止位:	1 💌	www.embe 🔺 dsky.net 🚽	
Em	校 验:	无 💌	发送	
	打开串口	关闭串口	清除发送区	
27	こ 串口測试	程序 天	🌭 🔔 22 : 41 📝	1

In PC (receive the data more than once):



The process of testing COM2 and COM3 is similar to COM1 test. The following diagrams show that COM2 and COM3 are open successfully:

COM2 is open successfully:





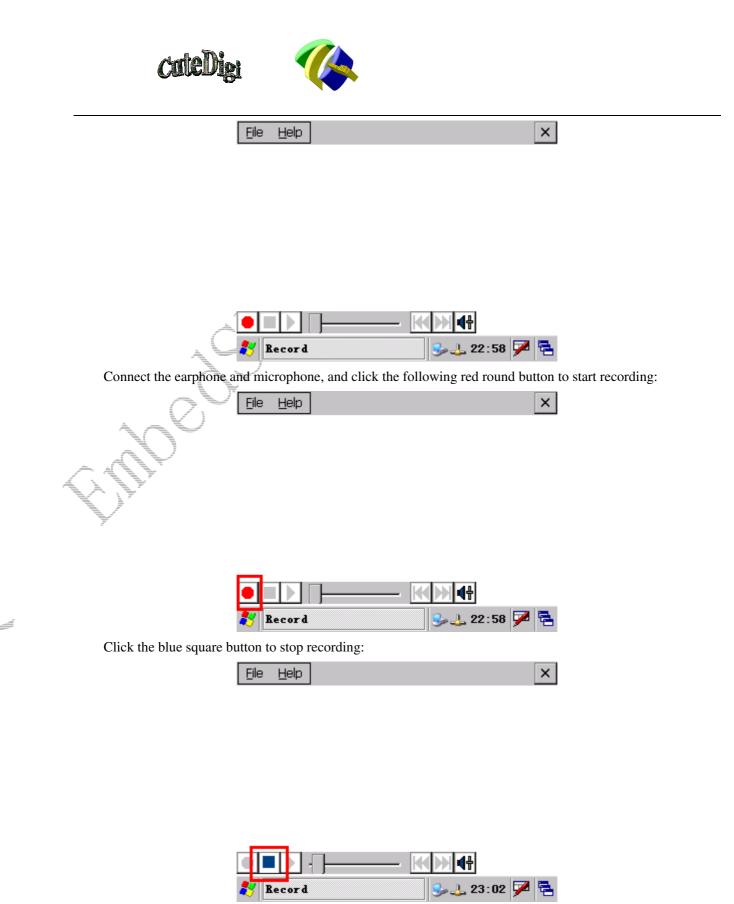
3.7.13 Experiment of recording test

_

Find the shortcut "录音机" on the desktop, as shown in the following diagram:



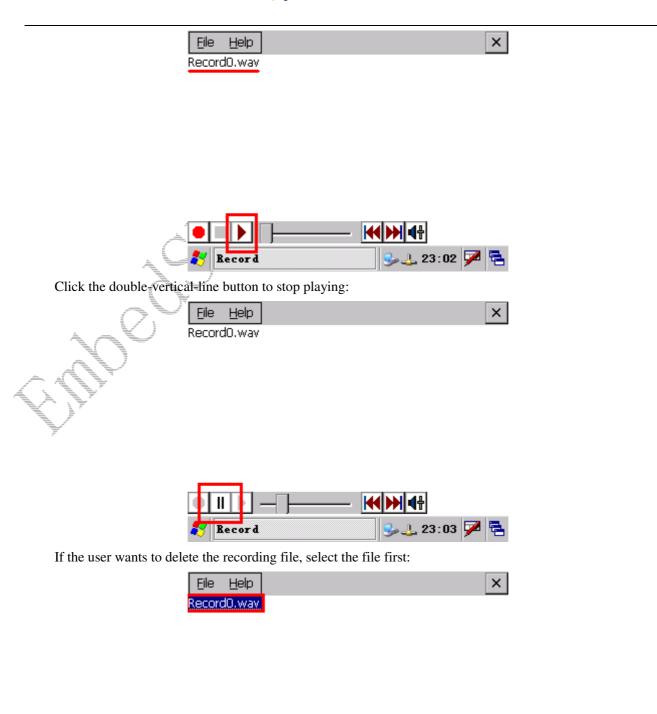
Double-click it to start recording:



Click the brown triangle button to replay the recording:





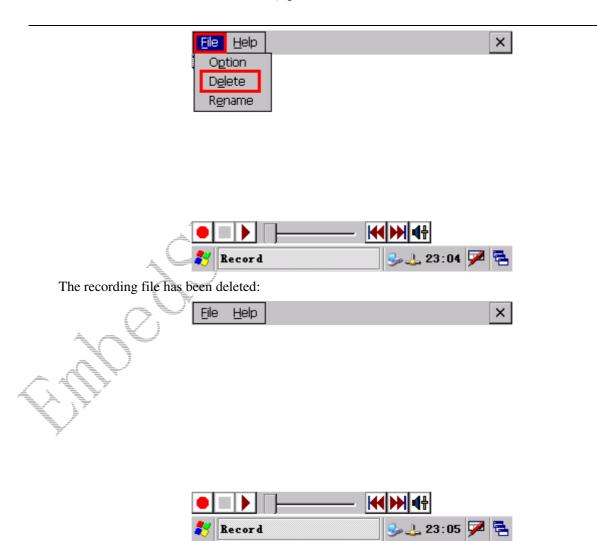




Click "File->Delete" to delete the file:







3. 7. 14 Experiment of surfing by IE explorer

Find the shortcut "Internet Explorer" on the desktop:

1



Start the IE explorer and open the website <u>www.embedsky.net</u>:



Í









3. 7. 15 Experiment of USB synchronization by ActiveSync

Please consult "5.5 节".

3. 7. 16 Experiment of WinCE self-carried game test

Click "开始->程序->纸	牌":		
<u>À</u>	3		
	🗾 🖬 ԱԾ Նե 🔤 Բուր	el Viewer 🗖 Microsoft 💋 Pov	werPoint
	💼 程序(<u>P</u>) 🔹 🕨	🗁 Microsoft File Viewers 🕨	
	☆ 收藏(<u>A</u>) 🔷 🕨	🗁 通信 🔹 🕨	
	🕒 文档(D) 🔷 🕨	🥰 Internet Explorer	ialPort
		😰 Microsoft WordPad	at many
		🐉 Windows 资源管理器	A
	一运行(<u>R</u>)	🙀 空当接龙	440
	圓 挂起(U)	🕑 纸牌	To and the second
	20	چ 🕹 0	:00 🏴 🗟

The playing card game:

Í



Click "开始->程序->当空接龙":







3. 7. 17 Experiment of Zuma game test

The running interface of Zuma game:

CateDigi















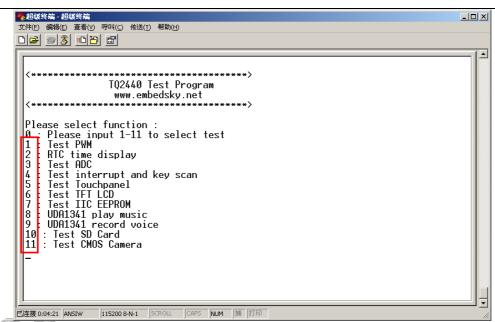
3. 8 Experiment of non-OS testing Demo

Download the non-OS testing Demo to SDRAM, or burn it to Nor Flash or Nand Flash. Please consult "3.3 节"(use H-Jtag to burn Nor Flash), or "Step7" of "3.4.3 节"(download the application to SDRAM), or "Step11" of "3.4.3 节"(burn application to Nor Flash).

Execute the non-OS testing Demo:







Enter the upper number according to prompt to start testing:

Caution: it is not suggested to select the option 10 (Test SD Card). Because when SD Card is being tested, a group of data would be written to SD card which probably destroy the original data. If the user tries to use the SD Card in PC or other device after doing this test, please format the SD Card first.

3.8.1 Experiment of PWM function test

-

This experiment needs a buzzer. The strength of sound made by buzzer indicates the output power of PWM.

Use the key "+" and "-" on keyboard to increase and decrease the number of PWM pulses. Or press the key "ESC" to exit:

文件() 編編() 査看() 呼叫() 代送() 帮助() □ 2 2 3 10 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
7 : Test IIC EEPROM 8 : UDA1341 play music	
9 : UDA1341 record voice	
10 : Test SD Card 11 Test CMOS Camera	
1 BUZZER TEST (PWM Control)	
Press +/- to increase/reduce the frequency of BUZZER !	
Press <u>'ESC'</u> key to Exit this program !	
Freq = 790 Freq = 780	
Freq = 770	
Freq = 760 Freq = 770	
Freq = 780 Freq = 790	
Freq = 800	
Freq = 810 Freq = 820	
Freq = 830	
Freq = 840 Freq = 850	
已连接 0:22:30 ANSTW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	1.

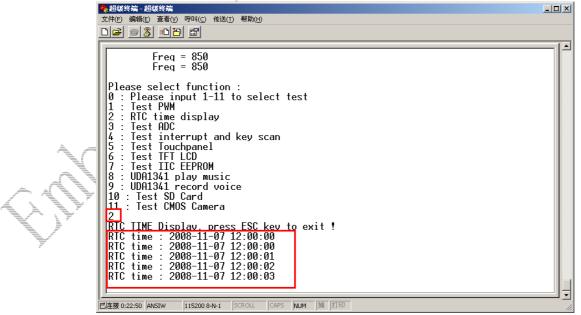


3.8.2 Experiment of real-time clock test

Write the test time to the relevant register of real-time clock, and then CPU will read it and print it out by serial port.

(be cautious to use this test. Because the test can probably modify the register of real-time clock. Thus you need to reset the real-time clock everytime when system starts up.)

Press "ESC" on keyboard to exit the test:



3.8.3 Experiment of ADC conversion test

Rotate the screw of adjustable resistance can correspondingly change the strength of output from serial port. (caution: The ripple exists in reference voltage. Therefore, there will be tiny difference between print-out values even if the value of resistance doesn't change.)

Press "ESC" on keyboard to exit:





《超敏终端 - 超敏终端	
文件(E) 編辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
3 : Test ADC	
4 : Test interrupt and key scan	
5 : Test Touchpanel 6 : Test TFT LCD	
6 : Test TFT LCD 7 : Test IIC EEPROM	
8 : UDA1341 play music	
9 : UDA1341 record voice	
10 : Test SD Card	
11 : Test CMOS Camera	
HUC INPUT Test, press ESC key to exit !	
ADC conv. freq. = 2500000Hz PCLK/ADC FREQ - 1 = 19	
AIN2: 0274	
AIN2: 0284	
AIN2: 0306	
AIN2: 0312	
AIN2: 0318	
AIN2: 0312 AIN2: 0322	
AIN2: 0322	
AIN2: 0342	
] _
已连接 0:49:50 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

3.8.4 Experiment of external interrupt test

The user can press external button on TQ2440 to create interrupt signal. The corresponding information appears in the following diagram:

Press "ESC" on keyboard to exit:

-

授 超级终端 - 超级终端 文件(F) 編撮(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)	<u>- </u>
8 : UDA1341 play music	
9. UDA1341_record voice	
10 : Test SD Card	
11 : Test CMOS Camera	
Key Scan Test, press ESC key to exit !	
Interrupt occur K1 is pressed!	
Interrupt occur K1 is pressed!	
Interrupt occur Key is released!	
Interrupt occur Key is released!	
Interrupt occur K2 is pressed	
Interrupt occur K2 is pressed!	
Interrupt occur Key is released! Interrupt occur Key is released!	
Interrupt occur Key is released!	
Interrupt occur K3 is pressed!	
Interrupt occur K3 is pressed!	
Interrupt occur Key is released!	
Interrupt occurKey is released!	
Interrupt occur K4 is pressed	
Interrupt occurK4 is pressed! Interrupt occur Key is released!	
Interrupt occur Key is released!	
Therefore the second se	
1]]
E连接 0:57:07 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

3.8.5 Experiment of touch-screen test

This test is similar to "3: Test ADC". Touch the touch-screen by using a pen to get the test values. Press any key on the keyboard to exit:





※超级终端 - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H) □ (C) (公) (公) (C) (公) (C) (C) (C) (C) (C) (C) (C) (C) (C) (C	
⑦ : Test IIC EEPROM 8 : UDA1341 play music 9 : UDA1341 record voice 10 : Test SD Card 11 : Test CMOS Camera 5 Touch Screen test Press any key to quit! Stylus Down, please. count=007 XP=0564, YP=0212 count=008 XP=0611, YP=0226 count=010 Count=010 XP=0625, YP=0599 count=011 XP=0658, YP=0655 count=015 XP=0636, YP=0468 count=015 XP=0626, YP=0530 count=018 XP=0618, YP=0530 count=018 XP=0618, YP=0590	

3.8.6 Experiment of LCD test

1

Press any key on the keyboard to start LCD test. The white color, blue color, green color, multi-color, flower and "EmbedSky" logo are displayed in sequence. Press any key to exit. The "EmbedSky" logo still appears on LCD after exit.

Press "ESC" on the keyboard to exit the test:

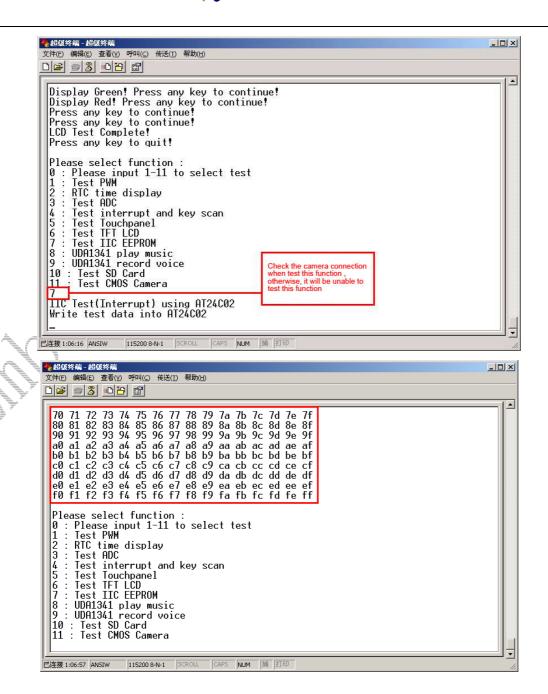
◆超級终端 - 超級终端	
文件(E) 编辑(E) 查看(V) 呼叫(C) 传送(I) 帮助(H)	
	1 •
1 : Test PWM	
2 : RTC time display	
3 : Test ADC	
4 : Test interrupt and key scan	
5 : Test Touchpanel 6 : Test TFT LCD	
7 : Test IIC EEPROM	
8 : UDA1341 play music	
9 : UDA1341 record voice	
10 : Test SD Card	
11 - Test CMOS Camera	
Test TFT LCD!	
Test IFI LUP	
Display Black! Press any key to continue!	
Display White! Press any key to continue!	
Display Blue! Press any key to continue!	
Display Green! Press any key to continue!	
Display Red! Press any key to continue!	
Press any key to continue!	
Press any key to continue! LCD Test Complete!	
Press any key to guit!	
	· _
已连接 1:06:00 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

3.8.7 Experiment of IIC interface test

Use the chip AT24C02 to make the test. Write some data into the chip and then read it. Make sure to remove the camera before the test, otherwise the test may fail:







3.8.8 Experiment of audio output test

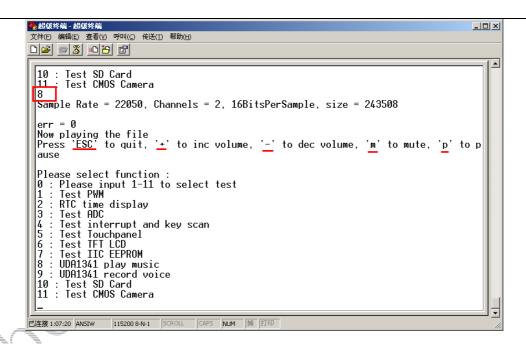
Í

Insert the earphone to hear the start-up music of Windows.

Press "ESC" to exit, "+/-" to increase and decrease the volume, "p" to pause and continue and "m" to mute:







3.8.9 Experiment of audio input test

Insert microphone and earphone. You can hear the voice from earphone when speak.

Press "ESC" on the keyboard to exit:

Í

参超级终端 - 超级终端 文件(C) 編載(C) 查看(V) 呼叫(C) 传送(T) 帮助(H)	- 🗆 🗵
10 : Test SD Card 11 : Test CMOS Camera 9 The Frequency of record is 48KHz err = 0 Added 1024 buffer for record Press any to Record Press any to Record Now begin recording, Press 'ESC' to quit Please select function : 0 : Please input 1-11 to select test 1 : Test PWM 2 : RTC time display 3 : Test ADC 4 : Test interrupt and key scan 5 : Test Interrupt and key scan 5 : Test IIC EEPROM 8 : UDA1341 play music 9 : UDA1341 record voice 10 : Test SD Card 11 : Test CMOS Camera	
二 	

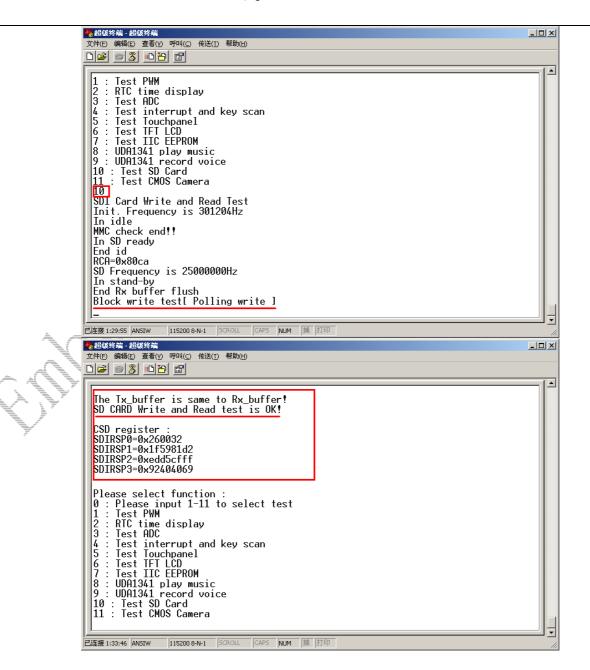
3.8.10 Experiment of SD card test

Write some data to SD card and then read it:

(be cautious to make this test. Some data would be written to the SD card during the test which can break the file system. After the test, you need to format the SD card in Windows OS for the further usage.)







3.8.11 Experiment of camera test

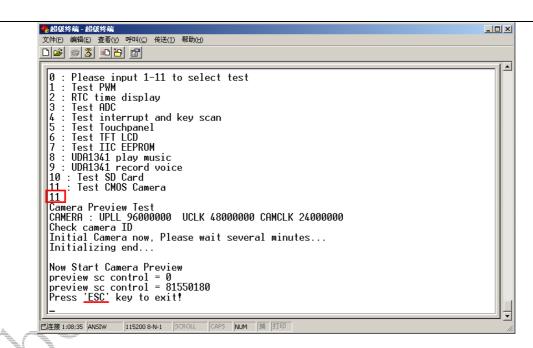
Í

Connect the camera and initialize it. Then the images captured by the camera will appear on LCD. Press "ESC" on the keyboard to exit:



Í









Chapter 4 Linux Development Manual

4.1 Compiling bootloader

_

Caution: The cross-compiler of version 2.95.3 is needed when compiling bootloader. Make sure to use the correct version before the compiling:

▼ root@HJ:/opt/EmbedSky	/// - - ×
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)	
[root@HJ EnbedSky]# /usr/local/arm/2.95.3/bin/arm-linux-gcc -v	*
Reading specs from /usr/local/arm/2.95.3/lib/gcc-lib/arm-linux/2.95.3/sp	pecs
gcc version 2.95.3 20010315 (release)	
[TOOTGED EIDEUSKY]#	
	\

If the compiler version 2.95.3 has been successfully installed, the upper information above the red line will appear; Otherwise please consult "2.5.1" to install the compiler.

bootloader source code is under the directory "Linux/bootloader.tar.bz2/" in CD-ROM. Copy "bootloader.tar.bz2" to the directory "/opt/EmbedSky/" and decompress it:

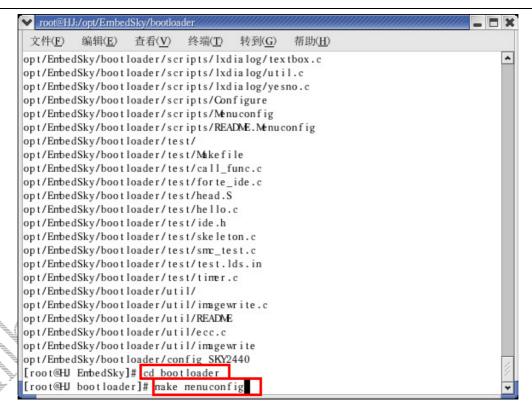
✓ root@H.	l:/opt/Embe	dSky					
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)	_	
[root@HJ	EnbedSky]# tar xvi	fj bootlo	ader.tar.	bz2 -C /		٠
opt/Enbed	dSky/boot	loader/				-	
opt/Enbed	dSky/boot	loader/ar	ch/				
opt/Enbed	dSky/boot	loader/ar	ch/de f-co	nfigs/			
opt/Enbed	dSky/boot	loader/ar	ch/de f-co	nfigs/sm	lk2440		
opt/Enbed	dSky/boot	loader/ar	ch/Makefi	le			
opt/Enbed	dSky/boot	loader/ar	ch/config	. in			
opt/Enbed	dSky/boot	loader/ar	ch/vivi.l	ds.in			
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/			
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/Makefil	le		
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/head.S			
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/nmu.c			
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/nand_re	ad.c		
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/proc.c			
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/smdk.c			
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/smdk244	0_test.c		
opt/Enbed	dSky/boot	loader/ar	ch/s3c244	0/test.c			
opt/Enbed	lSky/boot	loader/in	it/				
opt/Enbed	dSky/boot	loader/in	it/versio	n.c			
opt/Enbed	dSky/boot	loader/in	it/main.c				
opt/Enbed	iSky/boot	loader/.C	nangeLog.	swp			
opt/Enbed	dSky/boot	loader/00	PYING				
opt/Enbed	dSky/boot	loader/Cha	angeLog				2
		loader/Mal					
opt/Embed	dSky/boot	loader/Ru	les.make				¥

The following steps illustrate how to compile bootloader.

Step1, configure bootloader. Get into the directory of bootloader and input the command "make menuconfig":







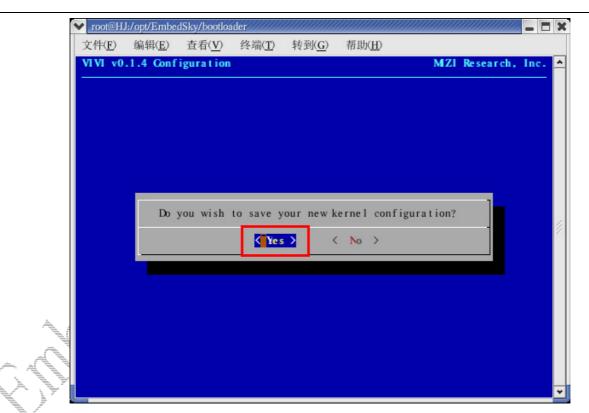
Step2, get into bootloader configuration interface and use the default parameter. Select "<Exit>" to exit:

	J:/opt/Embe 编辑(<u>E</u>)	dSky/bootlo: 査看(<u>V</u>)		转到(<u>G</u>)	帮助(<u>H</u>)	
VIVI v0	.1.4 Conf	iguration				MZI Research, Inc. 🔺
Hig Pre	hlighted ss <esc><</esc>		he menu. re hotkey xit,	s. Press for Help	selects su sing <y> in</y>	ubmenus>. ncludes, <№ excludes.
		Ceneral Private Serial Monory Add Bui		> > gy Devices mmands	s (MID)	>
		Load an	ng messag Alterna	ges> te Configu	iration Fi Alternate	
		< <mark>S</mark> e	lect≻	< Exit >	< He Ip	>

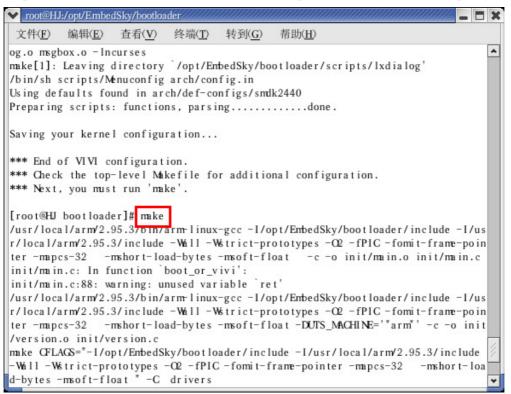
Select "<Yes>" to save the configuration and continue:







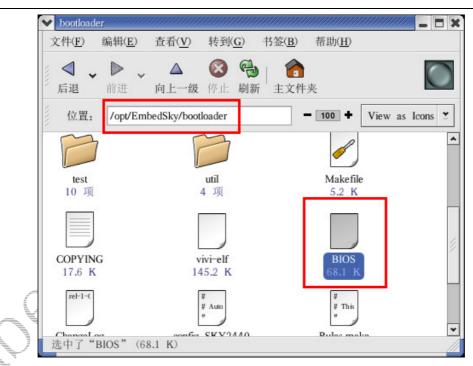
Step3, compile bootloader. Enter the command "make" and press return-key to start compiling:



Step4, the file "BIOS" will appear under the directory "/opt/EmbedSky/bootloader/" after compiling:







Burn the file "BIOS" to platform and then it can be used to boot Linux.

4. 2 Compiling Linux-2. 6 kernel

Caution: The cross-compiler of version 3.4.1_softfloat is needed when compiling Linux kernel of version 2.6. Make sure the compiler of correct version has been installed successfully.

v root@En	nbedSky:~							
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)			
	oedSky roo							*
					_3.4.1_soft	float/arm-linux/	gcc-3.4.	1-
-	3.3/lib/gc			-				
0						rm-linux/gcc-3.4	0	
						6-host_pc-linux-		
ix=/opt/H	EnbedSky/c	rosstools	s_3.4.1_s	oftfloat/	/arm-linux/	gcc-3.4.1-glibc-	2.3.3	wi
th-float=	softwi	th-header	s=/opt/E	mbedSky/o	crosstools_	3.4.1_softfloat/	arm-linu	x/ /
gcc-3.4.1	-glibc-2.	3.3/arm-	linux/inc	ludewi	i th-loca l-p	refix=/opt/Embed	Sky/cros	st 🖉
ools_3.4	.1_softflo	at/arm-li	inux/gcc-	3.4.1-gli	ibc-2.3.3/a	rm-linuxdisab	le-nls -	-e
nable-thi	eads=posi	xenab	le-synver	s=gnue	enablecx	a_atexitenabl	e-langua	ge
s=c, c++ -	-enable-s	harede	nable-c9	9enabl	le-long-lon	g		
Thread m	odel: posi	x						
gcc vers	ion 3.4.1	1000 C						
l root@Ent	oedSky roo	ot]#						+

If the compiler version 3.4.1_softfloat has been successfully installed, the upper information above the red line will appear; Otherwise consult "2.5.1 \ddagger " to install the compiler.

4.2.1 Use configuration file of EmbedSky to compile Linux kernel

The Linux kernel source code is under the directory "Linux/ kernel-2.6.13.tar.bz2". Execute the command "#tar xvfj kernel-2.6.13.tar.bz2 –C /" to decompress it to the directory "/opt/EmbedSky/kernel-2.6.13":

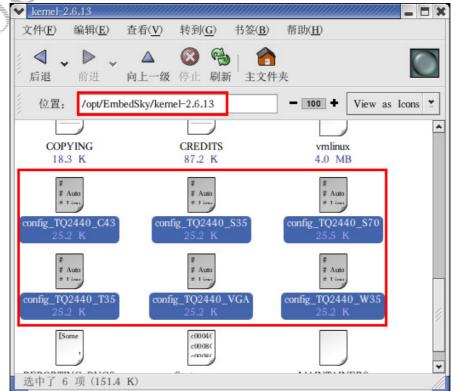




🚩 root@HJ:/opt/EmbedSky	1 ×
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
[root@HJ EnbedSky]# tar xvfj kernel-2.6.13.tar.bz2 -C / opt/EnbedSky/kernel-2.6.13/	*
op t/EnbedSky/kernel-2.6.13/fs/	
op t/EnbedSky/kernel-2.6.13/fs/fifo.c op t/EnbedSky/kernel-2.6.13/fs/aio.c	
op t/EnbedSky/kerne l-2.6.13/fs/xfs/ op t/EnbedSky/kerne l-2.6.13/fs/xfs/suppor t/	
op t/EnbedSky/kernel-2.6.13/fs/xfs/support/nove.h	
op t/EnbedSky/kerne l-2.6.13/fs/xfs/suppor t/ktrace.c op t/EnbedSky/kerne l-2.6.13/fs/xfs/suppor t/qsor t.h	8
op t/EnbedSky/kerne l=2.6.13/fs/xfs/suppor t/ktrace.h op t/EnbedSky/kerne l=2.6.13/fs/xfs/suppor t/uu id.c	

The default files are under the decompression directory:

- config_TQ2440_\$35 is the Samsung 3.5 inch LCD default configuration file.
- config_TQ2440_S70 is the Samsung 7 inch LCD default configuration file.
- config_TQ2440_T35 is the Toshiba 3.5 inch LCD default configuration file.
- ______Config_TQ2440_W35 is the Donghua 3.5 inch LCD default configuration file.



The following steps illustrate the process compiling Linux kernel: Step1, input "make menuconfig" to start configuring Linux kernel:

✓ root@EmbedSky:/opt/EmbedSky/kernel-2.6.13		×
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)		
[root@EnbedSky root]# cd /opt/EnbedSky/kernel-2.6.13/ [root@EnbedSky kernel-2.6.13]# make menuconfig		•
		•



1



Step2, select "Load an Alternate Configuration File":

文件(F) Linux K	编辑(<u>E</u>) ernel v2.(查看(<u>V</u>) 6.13 Conf	终端(<u>T)</u> iguration		帮助(<u>H</u>)		
Hig ≺M≱	hlighted modulari	avigate t letters a zes featu Legend: ode ener oada yste us s	he menu. re hotkey res. Pre [*] buil maturity al setup	<pre><enter> /s. Press /s. Press /ss <esc>/ t-in [] level op> le suppor>></esc></enter></pre>	sing <y> in (Esc> to ex excluded tions></y>	bmenus>. cludes, <№ it, for <№ module	excludes, Help,
· · · · · · · · · · · · · · · · · · ·		Float Sers Ower Notwo Fevic File Fofi Ferne Secur Crypt	pace bina manageme orking	t emulation ent option > > port> g> ons> options	$ts \longrightarrow ts \longrightarrow ts$		
		Load	an Alteri	nate Conf	<mark>iguration F</mark> an Alternat		

Step3, input the configuration file name according to the LCD type you are using. Here we use the configuration file of Toshiba 3.5 inch LCD for example. Click "OK" after input complete.



Í



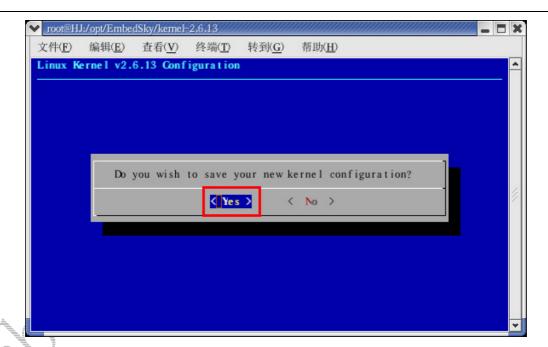
n die von de zuwanden die konste gehouwer nie bezeit die houde in der	
Enter the name of the configuration f to load. Accept the name shown to re- configuration you last r abort. Different name configuration fil to different LCD config_TC2440_T35	store the of the le correspond
Ck > < He 1p >	

Step4, go back to main menu. Select "<Exit>" to save the configuration and exit:

Y	root@HJ:/opt/EmbedSky/kemel-2.6.13	. 🗆 🗙
3	文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
I	Linux Kernel v2.6.13 Configuration	*
	Linux Rernel Configuration Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m <esc="" features.="" modularizes="" press=""><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m> module <> Cryptographic options> Library routines> Load an Alternate Configuration File Save Configuration to an Alternate File </m></esc></m></n></y></enter>	







Step6, input "make zImage" and press return-key to start compiling:

▼ root@EmbedSky:/opt/EmbedSky/kernel-2.6.13	//	×
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)		
<pre>[root@EmbedSky root]# cd /opt/EmbedSky/kernel-2.6.13/ [root@EmbedSky kernel-2.6.13]# make menuconfig scripts/kconfig/mconf arch/arm/Kconfig # # using defaults found in .config #</pre>		•
*** End of Linux kernel configuration. *** Execute 'make' to build the kernel or try 'make help'. [root@EmbedSky kernel-2.6.13]# make zlmage		*

Step7, the compiling is complete:

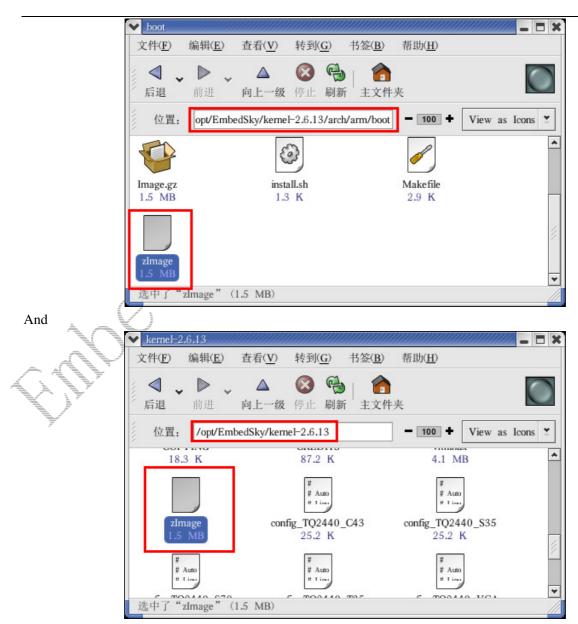
4



The kernel image file "zImage" is automatically created under the directory "/opt/EmbedSky/kernel-2.6.13/arch/arm/boot/" or "/opt/EmbedSky/kernel-2.6.13/" after compilation:







Burn the kernel image file to platform.

Í

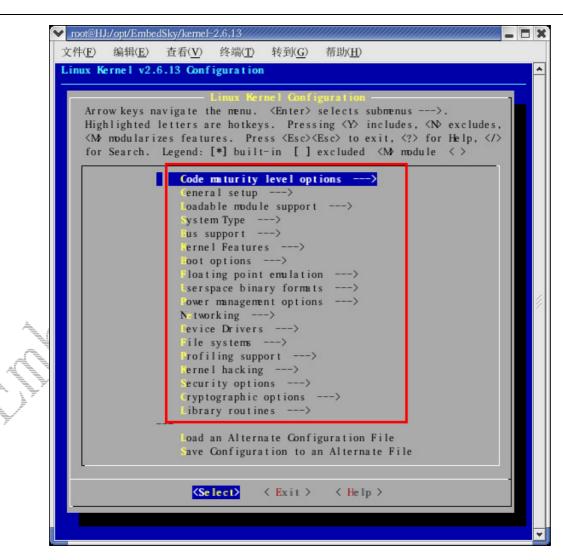
4.2.2 Customizing Linux kernel

In previous introductions, we use the default files to configure the Linux kernel. Here we introduce more other knowledge about configuring Linux kernel to enhance your understanding.

Execute "make menuconfig" to enter into Linux kernel configuration main menu:







➤ Configure CPU:

Í

Select "System Type" in main menu and press return-key:





		iguration			
1	ystem Type				
ubmenus>. ncludes, <n> excludes,</n>	<pre></pre>		•		100000000
xit, for Help,		· · · · · · · · · · · · · · · · · · ·			
	-in [] excluded <m< td=""><td></td><td></td><td></td><td></td></m<>				
			Lo genia	Jour en l	.01
>	(Samsung S3C24X0)	tem type	ARM s		
	tations>				
			S3C24		
		Se tup	S3C24		
	ock 12Mnz	Input Cl	*] \$3C244	[*]	
	ort	DMA supp	*] 3C24	[*]	
				- 14	
	ock 12Mnz	Boot Setup Input Cl	S3C24 S3C24 *] \$3C24	[*]	

The upper diagram displays the options of S3C2410, S3C2440, S3C24X0 and S3C24XX. The ARM chips of S3C24X0 serial always have the same register address and setting patterns. Therefore we do not do any configuration to these 2 kinds of CPU in version 2.6 Linux kernel any more.

In the menu "ARM system type (Samsung S3C24X0)", select the CPU type "Sansung S3C24X0" and press return-key to continue:

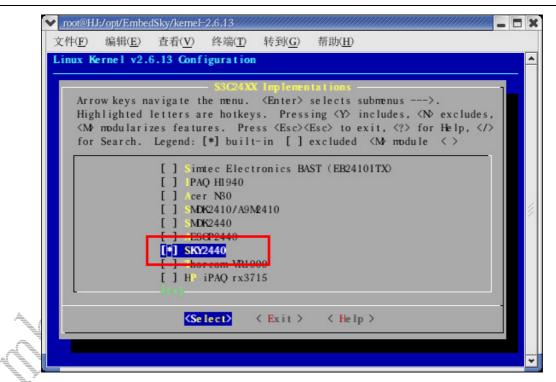
▼ root@HJ:/opt/EmbedSky/kernel=2.6.13	_ = ×
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
Linux Kernel v2.6.13 Configuration	A
ARM system type Use the arrow keys to navigate this window or press the hotkey the item you wish to select followed by the <space bar="">. Press <? > for additional information about this option. () BiscPC () TAI100 based () Samsung S3C24XD () Sharp LH7A40X () TI OMAP () Select></space>	
	✓

Select the platform type "SKY2440" in menu "S3C24XX Implementations". After that, select "Exit" and press return-key to continue:



Í



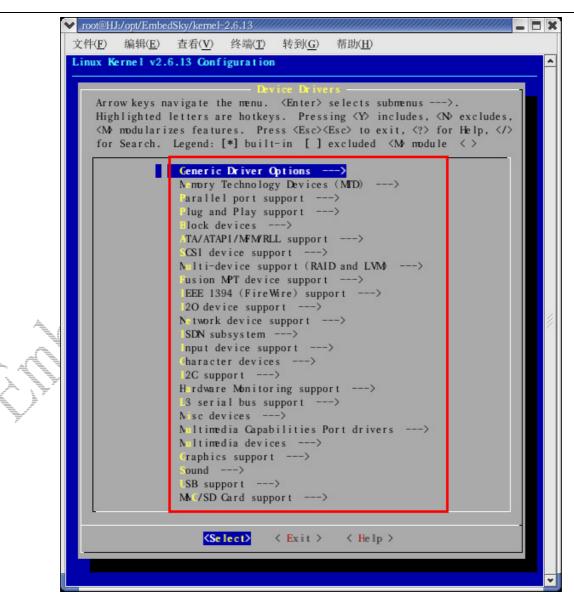


The type of our Development Board is SKY2440, which is corresponding to the file "sky2440.c" under the directory "arch/arm/mach-s3c2410/".

All the device configuration sub-menus are under the main menu "Device Drivers".







Configure LCD:

Í

Select "Graphics support" in the menu "Device Drivers", and press return-key to continue:





		evice Drivers		
	ys navigate the menu	. <enter> se</enter>	elects submenus>. ng ≺Y> includes, ≺N> exclud	es.
<m≯ nodu<="" td=""><td>larizes features. P</td><td>ress <esc><es< td=""><td>sc> to exit, <? > for Help, scluded <m> module < ></m></td><td></td></es<></esc></td></m≯>	larizes features. P	ress <esc><es< td=""><td>sc> to exit, <? > for Help, scluded <m> module < ></m></td><td></td></es<></esc>	sc> to exit, for Help, scluded <m> module < ></m>	
	H rdware Monit			
	3 serial bus Nisc devices		•	
			t drivers>	
	Craphics suppo			
	Sound >			
	SB support -			
	MN€/SD Card su	pport>		

Select "Support for frame buffer devices", "S3C24X0 LCD framebuffer support" and "LCD select" in sequence:

Linux Ke	扁辑(E) 查看(V) 终端(T) 转到(G) 帮助(H) nel v2.6.13 Configuration	-
High ≺M≱ :	• Graphics support keys navigate the menu. <enter> selects submenus>. ighted letters are hotkeys. Pressing <y> includes, <n> excludes, odularizes features. Press <esc> <esc> to exit, <? > for Help, earch. Legend: [*] built-in [] excluded <m> module < ></m></esc></esc></n></y></enter>	
	<pre><*> Support for frame buffer devices [*] Inable Video Mode Handling Helpers [] Inable Tile Blitting Support <> Ipson S1D13XXX framebuffer support <*> S3C24X0 LCD framebuffer support LCD select (3.5 inch 240X320 Toshiba LCD)> [] S3C24X0 LCD debug messages</pre>	
	<pre>< > Virtual Frame Buffer support (ONLY FOR TESTING!) Console display driver support> Logo configuration> [] Backlight & LCD device support></pre>	

Select the LCD driver in "LCD select" menu. Here we select "3.5 inch 240*320 Toshiba LCD":





Kernel v2.	6.13 Conf	iguration				
the item	you wish	to navig to select	followed	window or pr	ess the hotkey CEBARO.Press	
) 3.5 inc 3.5 inc 4.3 inc 3.4 inc	ch 320x240 ch 240X320 ch 480X272 ch 640x480	Samsung LCD WanXin LCD Toshiba LCD CHIMEI LCD TFT LCD Gamsung LCD		
		<se c<="" le="" td=""><td>1) <</td><td>Help ></td><td></td><td></td></se>	1) <	Help >		

After selection, press return-key to go back to the upper menu. And select "<Exit>" to go back to the menu "Device Drivers".

Configure touch-screen:

Í

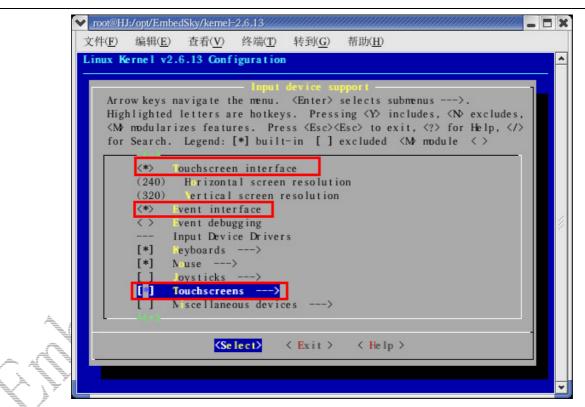
Select "Input device support" in the menu "Device Drivers" and press return-key to continue:

▼ root@HJ:/opt/EmbedSky/kernel-2,6.13	////// _ \ ×
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
Linux Kernel v2.6.13 Configuration	<u>^</u>
Arrow keys navigate the menu. (Enter) selects submenus>. Highlighted letters are hotkeys. Pressing (Y) includes, (N) ex (M modularizes features. Press (Esc) (Esc) to exit, (?) for He for Search. Legend: [*] built-in [] excluded (M module (SDN subsystem> inaracter devices 2C support> H rdware Monitoring support> 3 serial bus support> M Itimedia Capabilities Port drivers> M Itimedia devices> M Itimedia devices>	lp,
	¥

Select "Touchscreen interface", "Event interface" and "Touchscreens" in sequence to continue:







Select "Samsung S3C24X0 touchscreen input driver" in the sub-menu "Touchscreens" and continue:

文件④ 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H) Linux Kernel v2.6.13 Configuration Touchscreens Arrow keys navigate the menu. 〈Enter〉 selects submenus>. Highlighted letters are hotkeys. Pressing 〈Y〉 includes, 〈N excludes, 〈M modularizes features. Press 〈Esc〉 to exit, 〈?〉 for Help, 〈/〉 for Search. Legend: [*] built-in [] excluded 〈M module 〈 〉 Touchscreens Samsung S3C24X0 touchscreen input driver [*] ansung S3C24X0 touchscreen 〈 > Cunze AHL-51S touchscreen 〈 > Lo serial touchscreens 〈 > N croTouch serial touchscreens 〈 > N croTouch serial touchscreens 〈 > Keit > 〈 Help >	root@HJ:/	opt/Ember	dSky/kernel	-2.6.13				2 -	
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m <esc="" features.="" modularizes="" press=""><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m> module <> </m></esc></m></n></y></enter>	文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)			
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m <esc="" features.="" modularizes="" press=""><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m <="" module=""> Touchscreens Touchscreens </m></esc></m></n></y></enter>	Linux Ker	rnel v2.	6.13 Conf	iguration	n -				
	Highl ≺M≱r	lighted todulari: Search.	letters a zes featu Legend: Touchsere Samsung Samsu Cunze A Elo ser NicroTo	he menu. re hotkey res. Pre [*] built ens S3C24X0 ng S3C24X HL-51S to ial touch uch seria	<pre> <enter> ys. Press ess <esc> t-in [] touchscreen screens touchscreen </esc></enter></pre>	selects su sing <y> in (Esc> to en excluded een input of creen debug t</y>	ncludes, <n> exclude xit, <? > for Help, < <m> module < > driver g messages</m></n>		

Select "<Exit>" to go back to the upper menu, and select "<Exit>" to go back to the menu "Device Drivers".

➤ Configure USB mouse & keyboard:

Í

Select "USB support" in the menu "Device Drivers" and press return-key to continue:



Í





Find and select "Support for Host-side USB" and "OHCI HCD support":

Y	root@H	J:/opt/Embe	dSky/kernel	-2.6.13				- 0	×
	文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)			
1	Linux B	ernel v2.	6.13 Conf	iguration	1				-
1					ISB suppor				
	Ar r	ow keys n	avigate t				submenus>.		
	Hig	hlighted	letters a	re hotkey	s. Press	ing <y></y>	includes, <№ excludes,		
							exit, for Help,		
	for	Search.	Legend:	[*] built	-in []	excluded	≺M≯ module <>		
		Suppor t	for Host-	side USB					
	L J		rbose deb		e s				
			laneous U		1 S				
			vice file				D. T. L.		
			e USB ban c USB min						2
	li i		spend/res				L/		
			st Contro						
	< >	SP116	X HCD sup	port					
	<*>		CD suppor						
	$\langle \rangle$	SL811H	S HCD sup	por t					
			<se< td=""><td>lect></td><td>< Exit ></td><td>< He li</td><td>p ></td><td></td><td></td></se<>	lect>	< Exit >	< He li	p >		
	L				100000000				
L									-

Select the USB mouse and keyboard option "USB Human Interface Device (full HID) support" and "HID input layer support":





Jinux Ke	rnel v2.6.13 Configuration
High ≺M∳	USB support w keys navigate the menu. <enter> selects submenus>. lighted letters are hotkeys. Pressing <y> includes, <n> excludes, modularizes features. Press <esc> <esc> to exit, <? > for Help, Search. Legend: [*] built-in [] excluded <m> module <> USB Input Devices</m></esc></esc></n></y></enter>
$\langle \bullet \rangle$	USB Human Interface Device (full HID) support
[*]	H D input layer support Force feedback support (EXPERIMENTAL)
ίj	/dev/hiddev raw HID device support
< > 	Aiptek 6000U/8000U tablet support

Select "< Exit>" to go back to the menu "Device Drivers".

Configure USB memory:

The instruction set SCSI is needed.

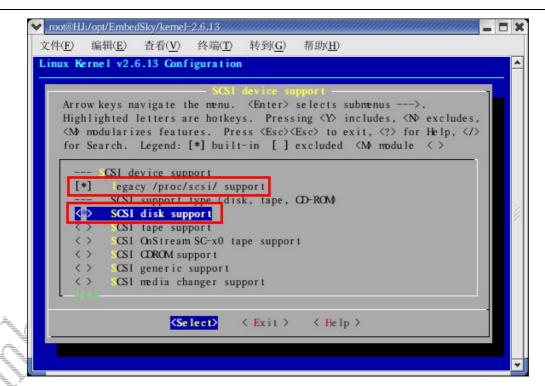
Select "SCSI device support" in the menu "Device Drivers" and press return-key to continue:

▼ root@HJ:/opt/EmbedSky/kernel=2.6.13	- 0	×
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)		
Linux Kernel v2.6.13 Configuration		*
Arrow keys navigate the menu. (Enter) selects submenus>. Highlighted letters are hotkeys. Pressing (Y) includes, (N) excludes (M) modularizes features. Press (Esc) (D exit, (?) for Help, (/) for Search. Legend: [*] built-in [] excluded (M) module (>) Parallel port support> Plug and Play support> Plock devices> TA/ATAPL/MEMRLL support> Niti-device support (RAID and LVM)> rusion MPT device support> EEE 1394 (FireWire) support> 20 device support>		

Select "legacy /proc/scsi support" and "SCSI disk support":







Select "<Exit>" to go back to the menu "Device Drivers".

Select "USB support" in the menu "Device Drivers" and press return-key to enter into "USB support" menu. And then select "USB Mass Storage support":

文件(E) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H) Linux Kernel v2.6.13 Configuration 	~	root@H	J:/opt/Embe	dSky/kernel-	-2.6.13					 ×
USB support Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing For modularizes features. Press <esc><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <</esc></esc></enter>	文	件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)			
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <n> modularizes features. Press <esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m> module <> <pre> </pre> <!--</td--><td>Li</td><td>nux I</td><td>ærnel v2.</td><td>6.13 Conf</td><td>iguration</td><td>1</td><td></td><td></td><td></td><td>*</td></m></esc></n></n></y></enter>	Li	nux I	ærnel v2.	6.13 Conf	iguration	1				*
		Hig <m for < > < ></m 	hlighted modulari Search. SB Mo SB Pr NOTE: USB M SB ata ree SD-	letters a zes featu Legend: inter sup USB STORA ss Storag Mass Stor fab Conpa com USB/A 200 USB/A otech/ZiO	he menu. re hotkey res. Pre [*] built ACM supp port GE enable e support age verbo ct Flash TAPI Brid TA Bridge ! Compact	<pre> <enter> /s. Press /ss <esc> /ss <esc> /sort /ss SCS1, a /sse debug Reader su /ge support /support /sflash/Sm </esc></esc></enter></pre>	selects sing <y> <esc> to excluded and 'SCSI upport (E rt artMedia</esc></y>	includes, <n> exit, <? > for I <m≱ module<br="">disk support EXPERIMENTAL) support</m≱></n>	excludes, Help, < >	100

Select "<Exit>" to go back to the menu "Device Drivers".

➤ Configure USB camera:

The USB camera needs the support of V4L.

Select "Multimedia devices" in the menu "Device Drivers" and press return-key to continue:





inux N	ernel v2.6.13 Configuration	
Hig ≺M≱	-Device Drivers ow keys navigate the menu. <enter> selects submenus>. hlighted letters are hotkeys. Pressing <y> includes, <n> excludes, modularizes features. Press <esc><esc> to exit, <? > for Help, Search. Legend: [*] built-in [] excluded <m> module <></m></esc></esc></n></y></enter>	
	Hardware Monitoring support> 13 serial bus support> Misc devices> Multimedia Capabilities Port drivers> Multimedia devices>	
	Craphics support> Sound> SB support> MMC/SD Card support>	
	<mark>⟨Select⟩</mark> ⟨Exit⟩ ⟨Help⟩	

Select "Video For Linux" and press return-key to continue:

Í

Y	root@H	U:/opt/Embe	dSky/kernel	-2.6.13				×
文	:件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)		
L	inux H	Kernel v2.	6.13 Conf	iguration	1			*
	Hig <m< th=""><th>hlighted modulari Search.</th><th>letters a zes featu Legend: *> \ideo Vide adi Digita</th><th>he menu. re hotkey res. Pre [*] built For Linux o For Lin o Adapter l Video F</th><th>vs. Press ess <esc> - in [] nux></esc></th><th>selects sing <y> (Esc> to of excluded</y></th><th></th><th>10</th></m<>	hlighted modulari Search.	letters a zes featu Legend: *> \ideo Vide adi Digita	he menu. re hotkey res. Pre [*] built For Linux o For Lin o Adapter l Video F	vs. Press ess <esc> - in [] nux></esc>	selects sing <y> (Esc> to of excluded</y>		10
								2

Select "OmniVision Camera Chip support" in the menu "Video For Linux":



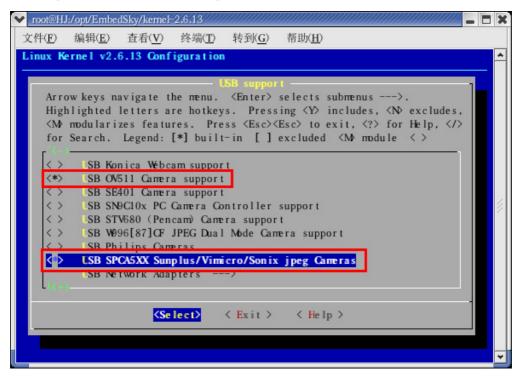


				转到(<u>G</u>)	111/00/11		
LINUX N	ernel v2.6	5.13 Conf	iguratio	n			
Hig ≺M≱	hlighted modularia	letters an zes featur	he menu. re hotkeg res. Pro	ys. Press ess <esc><</esc>	selects su ing ⟨Y⟩ in Esc> to ex	bmenus>. cludes, <№ it, for <№ module	excludes, Help,
		CPiA Vide SAA5246A SAA5249 ' SAB3036	ion Prol eo For L , SAA528 Teletext tuner	inux	_	or Linux	
		<mark>≺Se</mark>	lect>	<exit></exit>	< Help	>	

Select "<Exit>" to go back to the menu "Multimedia devices". And then select "<Exit>" again to go back to the menu "Device Drivers"

Select "USB support" in the menu "Device Drivers" and get into the sub-menu "USB support". Find and select "USB OV511 Camera support" and "USB SPCA5XX Sunplux/Vimicro/Sonix jpeg Cameras":

(OV511 supports the cameras based on OV511 chip, and SPCA5XX supports the cameras based on Smics micro 301 serials chips which own more than 70 percent market share.)



Select "<Exit>" to go back to the menu "Device Drivers".



Configure SD card:

Select "MMC/SD Card support" in the menu "Device Drivers", and press return-key to get in:

	✓ root@H	U:/opt/Embec	iSky/kernel	-2.6.13				 ×
	文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)		
	Linux H	Kernel v2.6	6.13 Conf	iguratio	1			*
All the second s	Hig ≺M	hlighted modularia	letters a zes featu Legend: H rdwar 13 seri N sc de N ltime Craphic Sound SB sup MC/SD	he menu. re hotkey res. Pro [*] buil e Monitor al bus su vices dia Capal dia devia s suppor > port Card supp	rs. Press ess <esc> in [] ing support poport> pilities l ces> t></esc>	selects su sing <y> in <esc> to ex excluded ort></esc></y>	excludes, Help,	111
								۲

Select "MMC support", "MMC block device driver" and "TQ2440 SD card support (no MMC)" in the sub-menu "MMC/SD Card support":

v ro	ot@H.	l:/opt/Embe	dSky/kernel-	-2.6.13				 ×
文件	⊧(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)		
Lin	ux K	ernel v2.	6.13 Conf	iguration	1			*
	High ≺M≱	i ighted modulari Search. (*> M [*] (*> (*) (>)	letters a zes featu Legend: C support M debug M block inbond W ansung S SKY2440 S	he menu. re hotkey res. Pre [*] built ging device d 83L51xD S 3C2410 M D card su	s. Pres: ss <esc> -in [] lriver D/MC Cat</esc>	selects sing <y> <esc> to excluded rd Interf Card Int NC</esc></y>	submenus>. includes, <№ excludes exit, for Help, 2<br I <№ module < > ace support erface support	
								¥

Caution: the upper configuration only contains SD card driver, and no MMC card support is included. Select "<Exit>" to go back to the menu "Device Drivers".

> Configure network card:

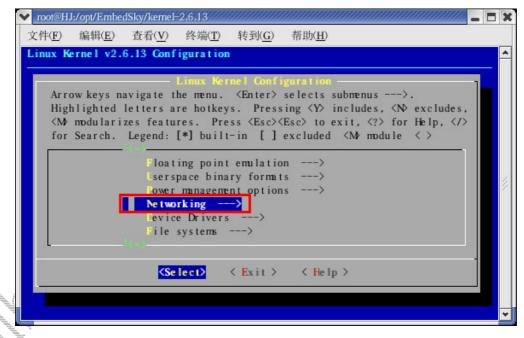
Ż



Í



Select "Networking" in the main menu to configure the network protocol support:



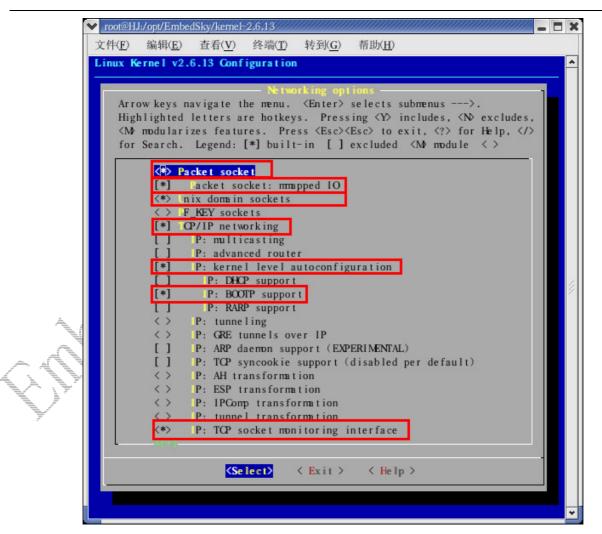
Select "Networking support" in the sub-menu "Networking" and get in:

▼ root@HJ:/opt/EmbedSky/kernel-2.6.13	- 🗆 X
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)	
Linux Kernel v2.6.13 Configuration	^
Networking – Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes <m> modularizes features. Press <esc> <esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m> module <></m></esc></esc></m></n></y></enter>	
[*] Networking support Networking options [] mateur Kadio support <> IrDA (infrared) subsystem support <> Eluetooth subsystem support	1
<pre></pre>	

Select the following options in the sub-menu "Networking options":







After the upper operation, go back to the main menu. And select "Network device support" in the menu "Device Drivers":



Á



	- Device Drivers -
Highlig ≺M≱ mod	teys navigate the menu. <enter> selects submenus>. whted letters are hotkeys. Pressing <y> includes, <n> excludes, hularizes features. Press <esc> <esc> to exit, <? > for Help, to the low of the low o</esc></esc></n></y></enter>
101 364	(=)
	N lti-device support (RAID and LVM)>
	EEE 1394 (FireWire) support>
	20 device support>
	Network device support>
	SLN subsystem>
	Input device support>
L	2C support>

Select "Network device support" and "Dummy net driver support" in the sub-menu "Network device support", and enter into the sub-menu "Ethernet (10 or 100Mbit)":

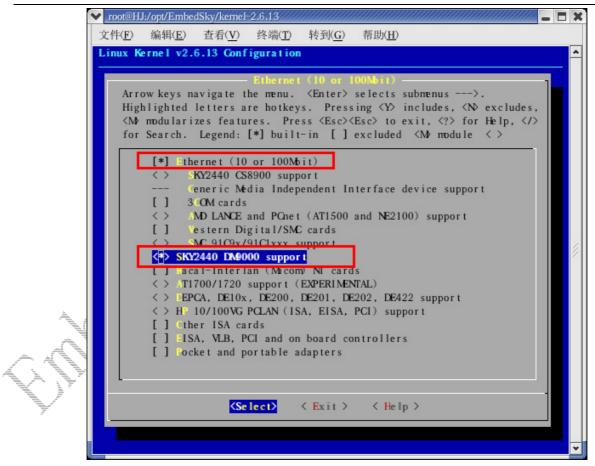
▼ root@HJ:/opt/EmbedSky/kernel=2.6.13	>
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
Linux Kernel v2.6.13 Configuration	<u>^</u>
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes <n# <esc="" features.="" modularizes="" press=""> to exit, <? > for Help, <!--<br-->for Search. Legend: [*] built-in [] excluded <m> module <> [*] N twork device support <> ionding driver support <> ionding driver support <> iQL (serial line load balancing) support <> iniversal TUN/TAP device driver support <> RCnet devices> HY device support> Ethernet (10 or 100Nbit)> i thernet (1000 Mbit)> </m></n#></n></y></enter>	
	¥

Select "Ethernet (10 or 100Mbit)" in the sub-menu "Ethernet (10 or 100Mbit)":

- Select "DM9000 support" to activate 100M network card:







Select "<Exit>" twice and go back to the menu "Device Drivers".

> Configure sound card:

Select "L3 serial bus support" in the menu "Device Drivers" before configuring the sound card:

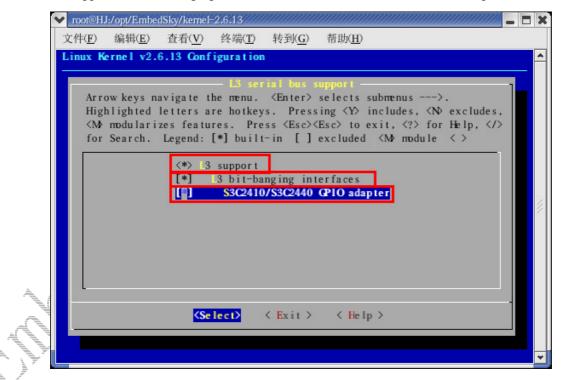
▼ root@HJ:/opt/EmbedSky/kernel-2.6.13	///// 🗕 🗖	X
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)		
Linux Kernel v2.6.13 Configuration		*
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> exclu <n <esc="" features.="" modularizes="" press=""> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m <="" module="">> H rdware Monitoring support> N sc devices> N sc devices> N ltimedia Gapabilities Port drivers> N ltimedia devices> Craphics support> SB support> M </m></n></n></y></enter>	10000001100000000	111
		7



_



Select "L3 support", "L3 bit-banging interfaces" and "S3C2410/S3C2440 GPIO adapter":



Select "<Exit>" to go back to the menu "Device Drivers", and select "Sound":

▼ root@HJ:/opt/EmbedSky/kernel-2.6.13		-
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)		
Linux Kernel v2.6.13 Configuration	-	
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m <esc="" features.="" modularizes="" press=""><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m <="" module=""> H rdware Monitoring support> A serial bus support> N isc devices> N itimedia Capabilities Port drivers> N Itimedia devices> Sound> SB support> M /SD Card support> M /SD Card support></m></esc></m></n></y></enter>		
		ļ

Select "Sound card support" and get into the sub-menu "Open Sound System":





Linux K	ernel v2.	6.13 Confi	iguration	1			
Hig ≺M∕	hlighted modulari	letters ar zes featur	e hotkey es. Pre	/s. Press ess <esc><</esc>	ing ≺Y> in Æsc> to ex	lbmenus>. acludes, <n> excludes it, <? > for Help, <!--<br-->≺M≥ module < ></n>	
	[dyana		oport Sound Ar	chitecture	>	
		(Se	le c t >	< Rvit >	< Help	>	

Select "Open Sound System (DEPRECATED)", "TQ2440 UDA1341 Driver" and "UDA1341 Stereo Codec" in the menu "Open Sound System":

▼ root@HJ:/opt/EmbedSky/kernel=2.6.13	//// _ 🗖 🗙
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)	
Linux Kernel v2.6.13 Configuration	<u>^</u>
Open Sound System Arrow keys navigate the menu. 〈Enter〉 selects submenus>. Highlighted letters are hotkeys. Pressing ⟨Y⟩ includes, ⟨N⟩ exclud ⟨M⟩ modularizes features. Press ⟨Esc⟩ ⟨Esc⟩ to exit, ⟨?⟩ for Help, for Search. Legend: [*] built-in [] excluded ⟨M⟩ module <>> (*> (pen Sound System (DEPRECATED)) ⟨*> (pen Sound System (DEPRECATED)) (*> SKY2440 UAI341 Driver (*> TDAI341 Stereo Codec (>> support for Turtle Beach MultiSound Classic, Tahiti, Monterey (>> Support for Turtle Beach MultiSound Pinnacle, Fiji (>> SS sound modules (>> TV card (bt848) mixer support (>> D1980 front/back switch plugin	.
	×

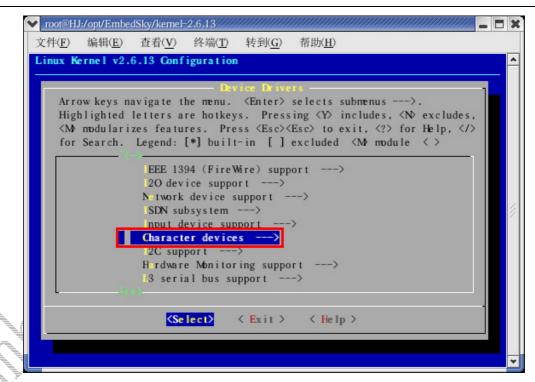
Select "<Exit>" twice to go back to the menu "Device Drivers".

➤ Configure user LED:

ź







Select "TQ2440 LEDs Driver":

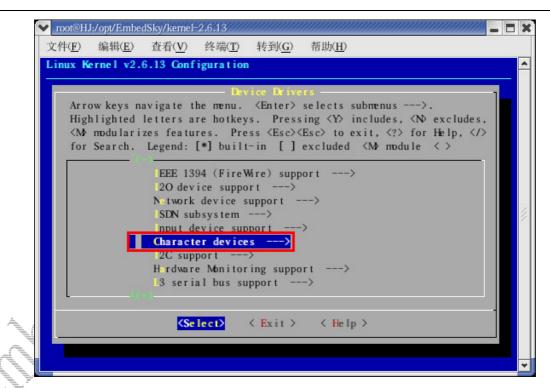
Í

▼ root@HJ:/opt/EmbedSky/kerneF2.6.13	X
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
Linux Kernel v2.6.13 Configuration	-
Arrow keys navigate the menu. (Enter) selects submenus>. Highlighted letters are hotkeys. Pressing (Y) includes, (N) excludes, (N) modularizes features. Press (Esc) to exit, (?) for Help, (/) for Search. Legend: [*] built-in [] excluded (N) module <> (>) Enhanced Real Time Clock Support [*] \$3C24X0 RTC Driver [N] SKY2440 LEDS Driver (N) SKY2440 HELLO Driver (>) Fouble Talk PC internal speech card support (>) Siemens R3964 line discipline Ftape, the floppy tape device driver> (>) FAW driver (/dev/raw/rawN) (OBSOLETE) TPM devices> (Select) (Exit) (Help)	

> Configure real-time clock:







Select "S3C24X0 RTC Driver":

Í

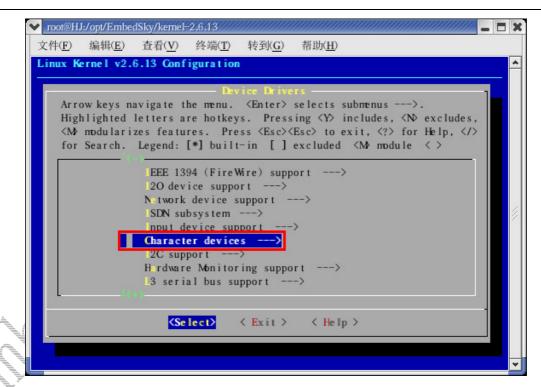
Caution: Don't select "Enhanced Real Time Clock Support".

文件① 编辑② 查看(Y) 终端(T) 转到(G) 帮助(H) Linux Kernel v2.6.13 Configuration Character devices Arrow keys navigate the menu. 〈Enter〉 selects submenus>. Highlighted letters are hotkeys. Pressing (Y) includes, 〈N excludes, 〈M modularizes features. Press 〈Esc〉(Esc〉 to exit, 〈?〉 for Help, 〈/〉 for Search. Legend: [*] built-in [] excluded 〈M module 〈 〉	v root@HJ:/opt/EmbedSky/kernel-2.6.13	- [X
Character devices Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, modularizes features. Press <esc><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <n> module <> inbanced Real Time Clock Support SSC24XD RTC Driver [*] KY2440 LEDs Driver Siemens R3964 line discipline itape, the floppy tape device driver> AW driver (/dev/raw/rawN) (OBSOLETE) PM devices></n></esc></esc></n></y></enter>	文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)		
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m <esc="" features.="" modularizes="" press=""> <to <?="" exit,=""> for Help, for Search. Legend: [*] built-in [] excluded <m <="" module=""> <pre></pre></m></to></m></n></y></enter>	Linux Kernel v2.6.13 Configuration		*
	Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <n> modularizes features. Press <esc><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <n> module <> <pre></pre></n></esc></esc></n></n></y></enter>		

> Configure TQ2440_Hello driver:







Set "M" in the option "TQ2440 HELLO Driver":

▼ root@HJ:/opt/EmbedSky/kernel-2.6.13	:	×
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)		
Linux Kernel v2.6.13 Configuration	-	-
Arrow keys navigate the menu. (Enter) selects submenus>. Highlighted letters are hotkeys. Pressing (Y) includes, (N) excludes, (M) modularizes features. Press (Esc) to exit, (?) for Help, (/) for Search. Legend: [*] built-in [] excluded (M) module (>) (> inhanced Real Time Clock Support [*] \$3C24X0 RTC Driver [*] \$3C24X0 RTC Driver [*] \$KV2440 IEDx Driver (> SKV2440 HELLO Driver (> SKV2440 HELLO Driver (> Stouble Talk PC Internal speech card support (> Siemens R3964 line discipline itape, the floppy tape device driver> (> AW driver (/dev/raw/rawN) (OBSOLETE) PM devices>		

> Configure serial port:

Í





High ≺M∳	- Device Drivers>. w keys navigate the menu. <enter> selects submenus>. lighted letters are hotkeys. Pressing <y> includes, <n> excludes, modularizes features. Press <esc><esc> to exit, <? > for Help, Search. Legend: [*] built-in [] excluded <m> module < ></m></esc></esc></n></y></enter>	
	IEEE 1394 (FireWire) support> 120 device support> Notwork device support>	
	SDN subsystem> nput device support> Character devices>	
	12C support> H rdware Monitoring support> 13 serial bus support>	
-		

Get into the sub-menu "Serial drivers":

à

▼ root@HJ:/opt/EmbedSky/kernel-2.6.13	- 🗆 X
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
Linux Kernel v2.6.13 Configuration	^
- Character devices	1
Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m> modularizes features. Press <esc> <esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m> module < ></m></esc></esc></m></n></y>	
Specialix RIO system support Stallion nultiport serial support Serial drivers	8
[*] legacy (BSD) PTY support (256) N ximum number of legacy PTY in use PM>	
<pre></pre>	
	¥

Select the following options:





inux Ke	编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H) rnel v2.6.13 Configuration
	Serial drivers
	w keys navigate the menu. <enter> selects submenus>.</enter>
-	lighted letters are hotkeys. Pressing ⟨Y⟩ includes, ⟨N⟩ excludes, modularizes features. Press ⟨Esc⟩ ⟨Esc⟩ to exit, ⟨?⟩ for Help, ⟨/⟩
	Search. Legend: [*] built-in [] excluded
	[*] Console on 8250/16550 and compatible serial port
	(4) N x inum number of 8250/16550 serial ports
	[] xtended 8250/16550 serial driver options
	Samsung S3C24XD Serial port support
	[*] Support for console on S3C24X0 serial port
	<pre></pre>

Select "< Exit>" twice to go back to the menu "Device Drivers".

> Configure Yaffs file system:

1

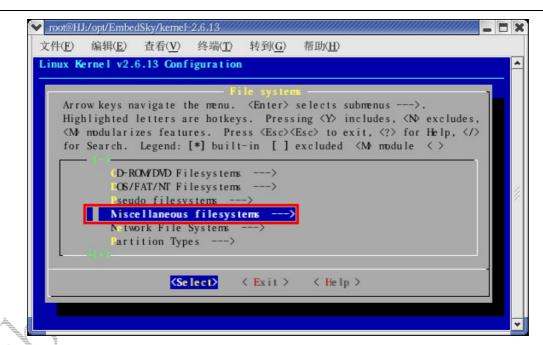
Select "File Systems" in the main menu and get in:

▼ root@HJ:/opt/EmbedSky/kernel-2.6.13	- 🗆 X
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
Linux Kernel v2.6.13 Configuration	^
Linux Kernel Configuration Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes <m <esc="" features.="" modularizes="" press=""> <esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <n> module <> fower management options> N tworking> File systems> File systems> fortiling support> 'Select> < Exit > < Help ></n></esc></m></n></y></enter>	
	•

Select "Miscellaneous filesystems" in the menu "File Systems" and get in:

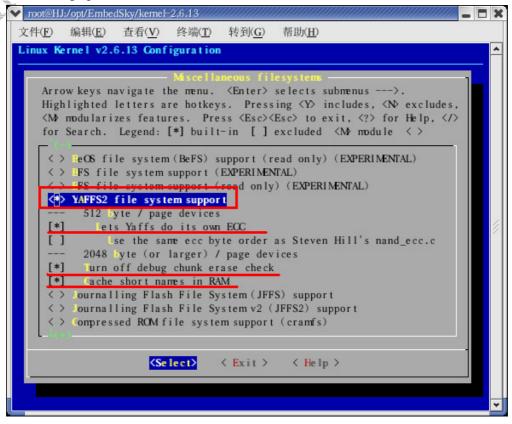






Select the following options:

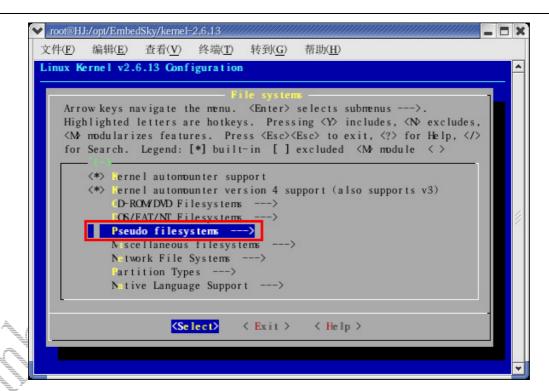
Á



Caution: Select "Pseudo filesystems" to make sure Yaffs file system works correctly:







Select "Virtual memory file system support (former shm fs)":

ź

▼ root@HJ:/opt/EmbedSky/kemel-2.6.13	- 🗆 X
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)	
Linux Kernel v2.6.13 Configuration	*
Pseudo filesystem Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m> modularizes features. Press <esc><esc> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <n> module <>> [*] /proc file system support [*] /proc file system support [*] /proc file system support [*] /utomatically mount at boot [] febug devfs [] /dev/pits Extended Attributes [*] Virtual memory file system support (former shm fs) [] mpfs Extended Attributes</n></esc></esc></m></n></y></enter>	

Select "<Exit>" to go back to the sub-menu "File System".

Select "Kernel automounter support" and "Kernel automounter version 4 support (also supports v3)" to make sure the kernel supports automounter:





	ernel v2.6.13 Configuration
Hig ≺M≯	 File systems ow keys navigate the menu. <enter> selects submenus>.</enter> hlighted letters are hotkeys. Pressing <y> includes, <n> excludes,</n></y> modularizes features. Press <esc> to exit, <? > for Help, Search. Legend: [*] built-in [] excluded <m> module <></m></esc>
	<pre>[*] Inotify file change notification support [] Cuota support <*> Fernel autonounter support</pre>
	Kernel automounter version 4 support (also supports v3) CD-RCW/DVD Filesystems> EOS/FAT/NT Filesystems>

Configure NFS file system:

1

The option "IP: BOOTP support" has been selected in the previous network device configuration:

▼ root@HJ:/opt/EmbedSky/kernel=2.6.13	
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
Linux Kernel v2.6.13 Configuration	^
Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes, <m <esc="" features.="" modularizes="" press=""> to exit, <? > for Help, for Search. Legend: [*] built-in [] excluded <m <="" module=""> [] P: advanced router [*] P: kernel level autoconfiguration [] P: BOOTP support [*] IP: BOOTP support [*] P: tunneling </m></m></n></y></enter>	
	-

Select "Network File System" in the menu "File System" and get in:





				转到(<u>G</u>)	帮助(<u>H</u>)	
.inux P	ernel v2.	6.13 Conf	iguration	n S		
Hig ≺M≱	hlighted modulari	letters a zes featu	he menu. rehotkey res.Pre	ys. Press ess <esc><</esc>	selects submenus>. ing <y> includes, <n> e Esc> to exit, <? > for l excluded <m> module <</m></n></y>	excludes, Help,
	Nisco Nisco	ork File	stems filesvst Systems	> tems>		
	Pseud Nisco Netwo Part	do filesy ellaneous	stems filesvst Systems es>	> tens>		

Select the following options:

À

-

✓ root@HJ:/opt/EmbedSky/kernel-2.6.13 文件(E) 编辑(E) 查看(W) 终端(T) 转到(G) 帮助(H)	_ 🗆 X
Linux Kernel v2.6.13 Configuration	<u>^</u>
Network File Systems Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> includes, <n> excludes <m> modularizes features. Press <esc> to exit, <? > for Help, <!--<br-->for Search. Legend: [*] built-in [] excluded <mp <="" module=""></mp></esc></m></n></y></enter>	
<pre></pre>	
Root file system on NFS Secure RPC: Kerberos V mechanism (EXPERIMENTAL) Secure RPC: SPKMS mechanism (EXPERIMENTAL) SMB file system support (to mount Windows shares etc.) SMB file system support (advanced network filesystem for Samba, Window and one state of the system for Samba, Window and one state of the system for Samba, Window and one state of the system for Samba state of the system for	
<pre></pre>	-
	•

Select "<Exit>" to go back to the menu "File Systems".

Configure the other file systems:

Select "DOS/FAT/NT Filesystems" in the menu "File System" and get in:





			iguration				
Hig ≺M≱	hlighted modulari	letters a zes featu	he menu. re hotkey res. Pre	/s. Press ess <esc><</esc>	selects su ing <y> in Esc> to ex</y>	bmenus>. cludes, <n> ex it, <? > for He ≺M≯ module <</n>	lp,
	<*> ern	OMOND Fi	unter vei lesystem	sion 4 su	pport (als	o supports v3)	
	seu	FAT/NE Fi do filesy ellaneous	stens	tems>	<i>k</i>		
					< Help		

Select the following options:

~	root@H.	l:/opt/Embe	dSky/kernel	-2.6.13				
Ż	:件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)		
L	inux K	ernel v2.	6.13 Conf	iguratio	1			^
	Higl ≺M≱	n lighted modulari	letters a zes featu Legend: (*> N (*> F (437) (so88 (> N	he menu. re hotkey res. Pro [*] buil DOS fs su AT (Windo efault of 59-1) Det FS file s	ys. Press ess <esc> t-in [] upport ows-95) fs codepage fault iocl system sup</esc>	selects si sing <y> in (Esc> to e: excluded support for FAT parset for oport</y>		
			< <u>Se</u>	lect>	< Exit >	< He lp	>	
								*

Select "<Exit>" to go back to the menu "File Systems".

Select "Ext2 extended attributes" to support EXT2 file system:





LINUX N	ernel v2.6.13 Configuration
Hig ≺M⊅	File systems — ow keys navigate the menu. <enter> selects submenus>. nlighted letters are hotkeys. Pressing <y> includes, <n> excludes, modularizes features. Press <esc> to exit, <? > for Help, Search. Legend: [*] built-in [] excluded <m→ <="" module=""></m→></esc></n></y></enter>
	<pre> <*> Second extended fs support L Ext2 extended attributes [] Ext2 POSIX Access Control Lists [] Ext2 Security Labels [] Ext2 execute in place support < > Ext3 journalling file system support </pre>
-	

Most information about kernel configuration has been introduced in the upper contents. However, a more profound understanding towards kernel configuration needs more experiences.

4. 3 Making root file system

The root file system is loaded by Linux when initializing. The user can execute the command "root=" to set the device that is corresponding to the root file system. The following contents introduce the process building Linux file system:

4.3.1 Components of root file system

Linux root file system usually contains the following directories:

- ♦ /dev—the directory of device file node. Mount the device file in this directory.
- /proc—mount proc file system in this directory.
- ♦ /bin—basic storage system commands are in this directory.
- ♦ /etc—system start-up configuration scripts, like rcS, inittab, fstab and so on are in this directory.
- ◆ /lib—directory of system default dynamic link liberary.
- ◆ /usr—user directory, includes "/usr/bin", "/usr/sbin" and so on.
- ♦ /sbin—basic storage system commands are in this directory.
- /tmp—temporary directory, it is not necessary.
- /var—this directory contains general variables used by system, and the size of it usually changes.

4.3.2 BusyBox compiling

BusyBox is a UNIX system tool set which contains most of the ordinary commands. The service of BusyBox



is utilized by executing link instructions.

ź

The following steps illustrate how to compile BusyBox:

Step1, decompress BusyBox. The source code of BusyBox is the file "BusyBox.tar.bz2" under the directory "Linux". Copy the file to the directory "/opt/EmbedSky/" in PC and execute the command "tar xvfj BusyBox.tar.bz2 -C /" to decompress it:

1	root@HJ:/opt/EmbedSky	
	:件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
I	oot@HJ EmbedSky]# tar xvfj BusyBox-1.2.0.tar.bz2 -C /	*
0	t/EnbedSky/BusyBox-1.2.0/	
0	t/EnbedSky/BusyBox-1.2.0/scripts/	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/inputbox.c	
o	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/msgbox.c	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/dialog.h	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/colors.h	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/BIG.FAT.WARNING	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/checklist.c	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/menubox.c	2
o	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/textbox.c	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/util.c	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/lxdialog/yesno.c	
0	t/EnbedSky/BusyBox-1.2.0/scripts/config/Kconfig-language.txt	+

Step2, get into the directory "/opt/EmbedSky/BusyBox-1.2.0/" and execute the command "make menuconfig":

▼ root@Er	nbedSky:/op	t/EmbedSky	/BusyBox-	1.2.0			- [- 2
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)			
		Box-1.2.0	2					*
•		Box-1.2.0.	0					
		Box-1.2.0. Box-1.2.0.		cmd				
1	<i>y y</i>		1	unstrippe	d			
		Box-1.2.0			-			
		Box-1.2.0						
					usyBox-1.2.	0/		2
[root@Em	bedSky Bu	syBox-1.2	.0]# make	nenuconf	ig		 	•

Step3, select "Load an Alternat Configuration File" in the following diagram:



Í



Arrow keys navigate the menu. <enter> selects submenus>. Highlighted letters are hotkeys. Pressing <y> selectes a feature, while <n> will exclude a feature. Press <esc> to exit, <? > for Help, for Search. Legend: [*] feature is selected [] feature is usybox Settings> Applets Archival Utilities> Coreutils></esc></n></y></enter>		vkeys na	avigate t		ox contrigu	121101		
Console Utilities> Pebian Utilities> Iditors> Finding Utilities> Init Utilities> Login/Password Management Utilities> Linux Ext2 FS Progs> Linux Module Utilities> Linux System Utilities> N scellaneous Utilities> N stworking Utilities> N-tworking Utilities> Shells> System Logging Utilities> Lad an Alternate Configuration File Save Configuration to an Alternate File	while	<n> wil</n>	letters a ll exclud r Search. Usybo Applet archiv Greut Gonsol ebian ditor indin nit U ogin/ Linux Linux N scel N twor roces shells System	re hotkey e a featu Legend: x Setting al Utiliti e Utilitie y Utilities Password Ext2 FS F Module Uf System Uf laneous U king Util s Utiliti >	<pre>//s. Press ire. Press ire. Press is [*] feat //ss> / ies> / ies> / / Managemen Progs / tilities / tilities / tilities / tilities / tilities / tilities / tilities / tilities / tilities / tilities<th>t Utilities</th><th>lectes a featu >> to exit, <?? ected [] fea</th><th>> for</th></pre>	t Utilities	lectes a featu >> to exit, ?<br ected [] fea	> for

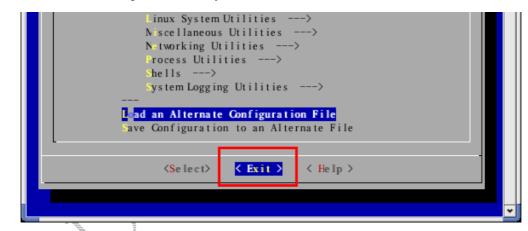
Step4, enter the name of configuration file "config_TQ2440" and press return-key to continue:

V root@	EmbedSky:/o	pt/EmbedSky	/BusyBox-	1.2.0				-		×
文件(<u>F</u>) 编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)					
									-	^
H wi	ighlig nile < Er elp, < to educed	oter the na o load. A onfiguratio oort. onfig_TÇ24	he menu. ame of th ccept the on you la 40 K Ck	ne config e name shu ast retric	selects su	e blank to	o. ure, ure is			
										۲





Step5, select "<Exit>" and press return-key to continue:



Select "<Yes>" to save the configuration:

|--|

Step6, execute the command "make" to start compiling BusyBox:

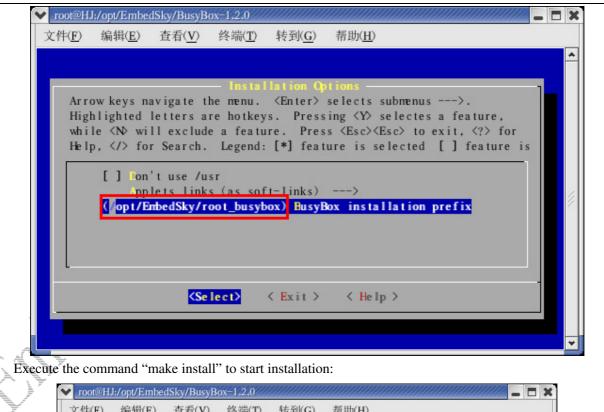


Step7, intall BusyBox after compiling. The installation path is "/opt/EmbedSky/root_busybox/": Set the installation path: "Busybox Settiings" -> "Installation Options" -> "BusyBox Installation prefix"



Í



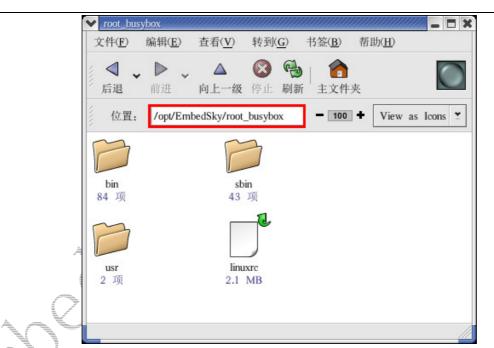


root@H.	l:/opt/Embe	dSky/BusyB	ox-1.2.0			
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)	
CC 1 ibl	ob/safe_s	trtod.o				
OC 1 ibl	b/safe_s	trtoul.o				
OC 1 ibi	b/bb_get	grgid.o				
OC 1 ibi	b/bb_xge	tgrnam.o				
OC 1 ibi	b/bb_xge	tpwnam.o				
OC 1 ibi	b/bb_get	pwuid.o				
CC 1 ibi	ob/bb_get	ug.o				
CC 1 ibi	b/get_ug	_id.o				
OC 1 ibi	b/llist_	add_to.o				
CC 1 ibi	ob/llist_	add_to_en	d.o			
OC 1 ibi	ob/llist_	pop.o				
CC 1 ibi	ob/llist_	free.o				
CC 1 ibi	ob/bb_ope	ndir.o				
CC 1 ibl	ob/bb_xop	endir.o				
AR cru	libbb/li	bbb.a				
LINK bu	usybox_un	stripped				
STRIP H	ousybox					
DOC bus	sybox.pod					
DOC Bu	syBox.txt					
DOC Bu	syBox.1					
	syBox.htm			-		
root@HJ	BusyBox-	1.2.0]# n	ake insta	11		

Step8, the 4 directories "bin/", "sbin/", "usr/bin/" and "usr/sbin/" and the file "linuxrc" appear under the directory "/opt/EmbedSky/root_busybox/" after installation: ("linuxrc" is actually the shortcut of the file "bin/busybox")







After compiling, add all the files mentioned in "3.3.1 节" to the directory "root_busybox". And then the BusyBox file system is built completely.

4. 3. 3 Building root file system

ź

Decompress "root_condense.tar.bz2" under the directory "Linux" to "/opt/EmbedSky". Execute the command "tar xvfj root_condense.tar.bz2 -C /":

🔽 root@HJ:/opt/EmbedSky	X
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)	
[root@HJ EnbedSky]# tar xvfj root_condense.tar.bz2 -C /	٠
opt/EnbedSky/root_condense/	
opt/EnbedSky/root_condense/linuxrc	
opt/EnbedSky/root_condense/web/	
opt/EnbedSky/root_condense/web/webcam-result.tenplate	
opt/EnbedSky/root_condense/web/webcam.html	
opt/EnbedSky/root_condense/web/webcam.cgi	
opt/EnbedSky/root_condense/web/SwingWorker.java	
opt/EnbedSky/root_condense/web/leds.html	
opt/EnbedSky/root_condense/web/leds.cgi	4
opt/EnbedSky/root_condense/web/led-result.template	
opt/EnbedSky/root_condense/web/JWebcamPlayer.java	
opt/EnbedSky/root_condense/web/JWebcamPlayer.jar	
opt/EnbedSky/root_condense/web/index.html	
opt/EnbedSky/root_condense/web/images/	
opt/EnbedSky/root_condense/web/images/bottom.jpg	
opt/EnbedSky/root_condense/web/images/bar_right_soft.jpg	
opt/EnbedSky/root_condense/web/images/bar_mid_soft.jpg	
opt/EmbedSky/root_condense/web/images/bar_left_soft.jpg	
opt/EnbedSky/root_condense/web/images/bar_right_thick.jpg	*

Copy the BusyBox compiled in "4.3.2 ⁺⁺;" to the corresponding directory under "root_condense".

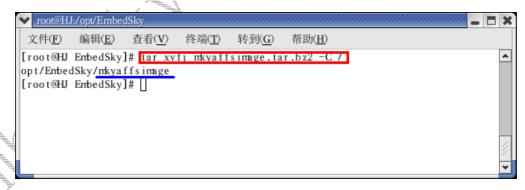


4.3.4 Making Yaffs root file system image

The software "mkyaffsimage" is needed when making Yaffs file system image.

"mkyaffsimage" is in the compression package "mkyaffsimage.tar.bz2" and "crosstools_all.tar.bz2" under the directory "Linux" in CD-ROM. "mkyaffsimage" has been installed previously when installing cross-compiler. The user can re-install it here. Decompress the file "mkyaffsimage.tar.bz2", and find "mkyaffsimage":

(the decompression command is shown in the red frame, and the executable program decompressed which is under the directory "/opt/EmbedSky/" is highlighted by the blue underline.)



Execute the command "./mkyaffsimage root_condense/ root_condense.img" under the directory "/opt/EmbedSky/" to make Yaffs root file system image.

▼ root@HJ:/opt/EmbedSky	- 🗆 🛪
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助	か(<u>H</u>)
[root@HJ EmbedSky]# tar xvfj mkyaffsimage.tar.bz opt/EmbedSky/mkyaff <u>simage</u>	2 -C /
[root@HJ EnbedSky]# ./nkyaffsimage root_condense mkyaffsimage: image building tool for YAFFS buil	
Processing directory root_condense/ into image f Object 257, root_condense//linuxrc is a symlink	
Object 258, root_condense//web is a directory Object 259, root_condense//web/webcam-result.tem	nlate is a file 1 data chunks w
ritten	

The red frame marks the created file:

▼ root@HJ:/opt/EmbedSky	_ 🗆 X
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
[root@HJ EmbedSky]# ls root_	*
root_busybox root_condense.ing root_nfs.tar.bz2 root_qtopia_tp root_condense root_nfs root_qt_mouse	
[root_condense root_nrs root_qt_nbase	
	-

Follow the steps of burning file system introduced previously to burn "root_condense.img" to platform.



4.4 Compiling u-boot

The file "u-boot.tar.bz2" in CD-ROM is the source code package of u-boot for TQ2440.

4.4.1 Decompressing u-boot

Execute the decompression command: tar xvfj u-boot.Tar.bz2 -C /

	root@Embe	dSky:~							//// - [= X
	文件(<u>F</u>) 练	高辑(<u>E</u>) 査	看(<u>V</u>) 約	冬端(<u>T)</u> 著	专到(<u>G</u>)	帮助(<u>H</u>)				
ſ	root@Enbed	Sky root]	# tar xvf	fj u−boot	.tar.bz	-C /				*
<u>×</u>										4
A										*
The file i	s auto-deco	mnressed	to the di	rectory "	/ont/Emb	edSkv/i	1-boot-1.1.6	/".		
	EmbedS	-						" . "///////////////////////////////////		×
ar -	文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V)</u>	转到(<u>G</u>)	书签(<u>B</u>)	帮助(<u>F</u>	Ð			-
			A							
	后退	前进	向上一级	停止 刷	♥	牛夹			\bigcirc	
	1			1						
	位置:	/opt/Emb	edSky	24 7火			- 100 +	View as		
		-		~			~			<u>^</u>
	r	-								
	root	busybox		root_conder	ise		root_nfs			
		5 项		15 项			17 项			_
		3		2			Z			1
		t_mouse		root_qt_t	þ	u	-boot-1.1.6			
	16.5	;项	11 / + + 10	15 项			43 项			•
	远中了	u-boot-1.1.6	(月 43	- 坝)					1	

4.4.2 Configuring u-boot

Í

Execute the command: make TQ2440_config





▼ root@EmbedSky:/opt/EmbedSky/u=boot=1.1.6	- 🗆 X
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
[root@EnbedSky EnbedSky]# cd /opt/EnbedSky/u-boot-1.1.6/	*
[root@EmbedSky u-boot-1.1.6]# make SKY2440_config Configuring for SKY2440 board	
[root@mbedSky u-boot-1.1.6]#	
	2
	•

3.4.3 Compiling u-boot

Execute the command: make

	✓ root@EmbedSky:/opt/EmbedSky/u=boot=1.1.6
Manual V	文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
-	[root@EnbedSky EnbedSky]# cd /opt/EnbedSky/u-boot-1.1.6/
	[root@EnbedSky u-boot-1.1.6]# make SKY2440_config
	Configuring for SKY2440 board
	[root@EmbedSky u-boot-1.1.6] make
	for dir in tools examples post post/cpu ; do make -C \$dir _depend ; done
	make[1]: Entering directory `/opt/EnbedSky/u-boot-1.1.6/tools'
	make[1]: Nothing to be done for _depend'.
	make[1]: Leaving directory /opt/EnbedSky/u-boot-1.1.6/tools
	make[1]: Entering directory '/opt/EmbedSky/u-boot-1.1.6/examples'
	make[1]: Nothing to be done for _depend'.
	make[1]: Leaving directory '/opt/EnbedSky/u-boot-1.1.6/examples'
	make[1]: Entering directory '/opt/EmbedSky/u-boot-1.1.6/post'
	make[1]: Nothing to be done for _depend'.
	make[1]: Leaving directory '/opt/EnbedSky/u-boot-1.1.6/post'
	Π
I	

Compiling is complete:

1



Í



文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)		
r - xyzN	odem.o						
r - cmd_							
	Leaving of						
	•					rm-linux/gcc-3.4.1-glibc-2.	
					-	board/SKY2440/libSKY2440.a	
						24x0.a lib_arm/libarm.a fs/c	
						fs/jffs2/libjffs2.a fs/reise	
						isk/libdisk.a rtc/librtc.a d	
						1.a drivers/nand_legacy/libr	
						ommon/libcommon.a sed -n -	
100 Description 10	(u_boot						
cd /opt/EnbedSky/u-boot-1.1.6 && /opt/EnbedSky/crosstools_3.4.1_ t/arm-linux/gcc-3.4.1-glibc-2.3.3/bin/arm-linux-ld -Bstatic -T /opt/Enbed							
	~	~					
	6/board/S	KY2440/u-	boot.lds	-Ttext Ox	33F80000	\$UNDEF_SYM cpu/arm920t/sta	
rt.o \			1.1				
					~	a board/SKY2440/libSKY2440.	
						24x0.a lib_arm/libarm.a fs/	
						fs/jffs2/libjffs2.a fs/reis	
erfs/libreiserfs.a fs/ext2/libext2fs.a net/libnet.a disk/libdisk.a rtc/librtc.a dtt/libdtt.a drivers/libdrivers.a drivers/nand/libnand.a drivers/nand_legacy/lib							
						conmon/libconmon.aend-gro inux/gcc-3.4.1-glibc-2.3.3/l	
	rm-linux/3	~		1_\$011110	at/arnr11	Inux/gec-3.4.1-g110e-2.3.3/1	
i i brgeera		Map u-boo		u-boot			
/ont/Ent		1	1		Linux/ac	cc-3.4.1-glibc-2.3.3/bin/arr	
/opt/land	-					-	
-linuv-o	ojcopy j						
-linux-o	edSky/cros	estonle 2	4 1 soft	tlost/srn	-1 muy/ ac	-3 4 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 -	
/opt/Emb	edSky/cros					cc-3.4.1-glibc-2.3.3/bin/arr	

Caution: Make sure that the cross-compiler supports softfloat, otherwise the compiling would probably fail.





5.1 Installing BSP of TQ2440

TQ2440 BSP needs to be configured when using WinCE image compiled by PB. The following contents introduce the installation steps:

Step 1, open the directory "F \WINCE500\PLATFORM":



Step 2, copy "WinCE\WinCE_5.0\SMDK2440" in CD-ROM to the directory "WINCE500\PLATFORM":

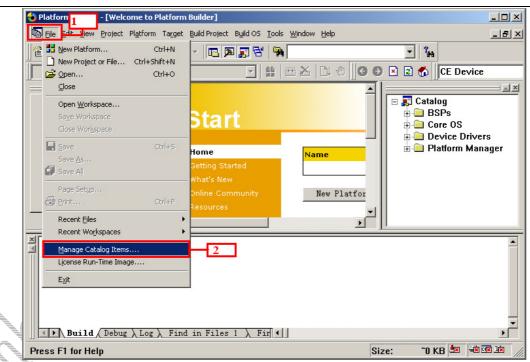
C PLATFORM		
文件(E) 编辑(E) 查看(∀) 收藏(A) 工具(T) 帮助(H)	
Ġ 后退 🔹 🕥 🖌 🏂 🔎 搜	索 🝺 文件夹 🛄 -	
地址(D) 🗁 F:\WINCE500\PLATFORM		▼ 🔁 转到
文件和文件夹任务	COMMON MAINSTONEII	EMULATOR SMDK2410
 ₩eb ₩eb ₩eb ※ 共享此文件夹 ※ 以电子邮件形式发送 该文件夹内的文件 	SMDK2440	

Step 3, remove the read-only property of all files under the directory "SMDK2440";

Step 4, run "Platform Builder 5.0", select "Manage Catalog Features" of the menu "File", and get into BSP package manager menu:







Step 5, select "Import" to load the file "F:\WINCE500\PLATFORM\SMDK2440\smdk2440.cec" in the following Manage Catalog Features interface:

File	Version	Vendor	Description		OK
generic.cec	5.00	Microsoft	Generic driver type declar		
audio.cec	5.00	Microsoft	Audio Catalog Items		Remove
display.cec	5.00	Microsoft	Display Catalog Items		
1394. cec	5.00	Microsoft	IEEE 1394 Driver Support		[Import
keybmouse, cec	5.00	Microsoft	Keyboard and Mouse Catalog		L'Emport.
netcard. cec	5.00	Microsoft	Ethernet Catalog Items		Refresh
irda. cec	5.00	Microsoft	IrDA Catalog Items		<u>nerresn</u>
parallel.cec	5.00	Microsoft	Parallel Port Catalog Items		
pci.cec	5.00	Microsoft	PCI Bus Catalog Items		
pomoia.cec	5.00	Microsoft	PCMCIA Catalog Items		
printing. cec	5.00	Microsoft	Printing and Printer Drivers		
serial.cec	5.00	Microsoft	Serial Port Catalog Items		
smartcard.cec	5.00	Microsoft	Smart Card Support and Dri		
storage.cec	5.00	Microsoft	Storage Device Support and		
usb. cec	5.00	Microsoft	USB Catalog Items	Ţ	
	E 00	H	A DCD C.I.J. TI		

Click "import" and open "smdk2440.cec":

Í





Import Catalog	items	? ×
查找范围(<u>t</u>):	🗁 SMDK2440 💽 🧿 📂 🖽 -	
ましたの文档 (で) (で) (で) (で) (で) (で) (で) (で)	DRIVERS eboot eboot_org FILES GWE INC KERNEL ib NBOOT target smdk2440.cec 1	
-	文件名 (M): sm dk2440. cec 了打开	Ŧ@)
	文件类型 ①): Catalog Item Files (*.cec) 🔽 取	消

Select "smdk2440.cec" and click "OK" to continue.

File	Version	Vendor	Description	OK
cepc. cec	5.00	Microsoft	CEPC BSP Catalog Items	
dbau1000. cec	5.00	Microsoft	DBAu1000 BSP Catalog Items	Remove
dbau1100. cec	5.00	Microsoft	DBAu1100 BSP Features	
dbau1500. cec	5.00	Microsoft	DBAu1500 BSP Catalog Items	Import
emulator.cec	5.00	Microsoft	Emulator BSP Catalog Items	- Empore
geode. cec	5.00	Microsoft	Geode BSP Catalog Items	Refresh
mainstonei	5.00	Microsoft	Intel MainstoneII BSP Cata	<u>nerresn</u>
sg2_vr4131	5.00	Microsoft	SG2_VR4131 BSP Catalog Items	
sg2_vr5500	5.00	Microsoft	SG2_VR5500 BSP Catalog Items	
platman. cec	5.00	Microsoft	Platform Manager Features	
coreos. cec	5.00	Microsoft	Windows CE Core OS components	
wcetk. cec	5.00	Microsoft	Windows CE Test Kit Client	
sourcetags	5.00	Microsoft	Source code information fo	

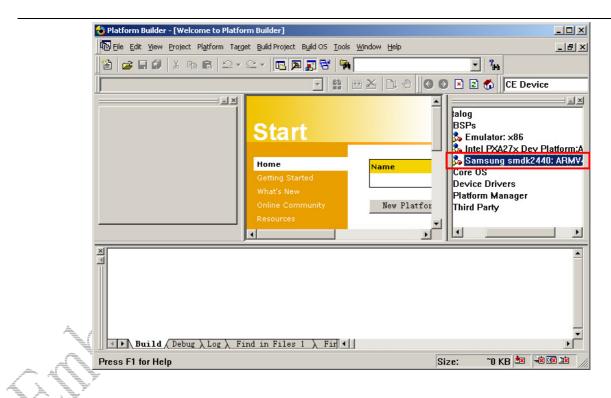
Step 6, after BSP installation, the option "Samsung SMDK2440: ARMV4I" is auto-added to the PB interface, below the option "BSPs" of "Catalog".

As the following red frame indicates:

1

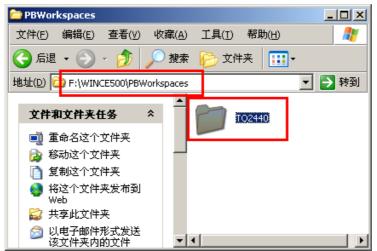






5.2 Compiling the example projects in CD-ROM

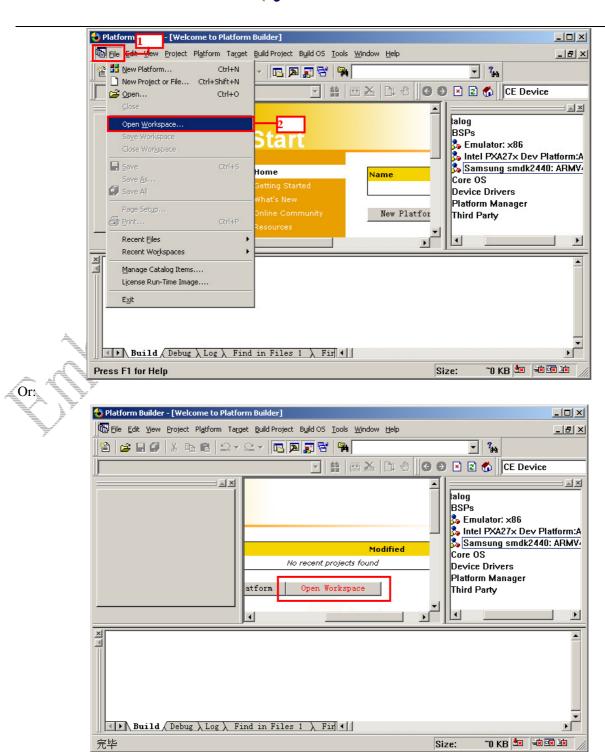
Step1, find the "TQ2440" project files under the directory "WinCE\WinCE_5.0" in CD-ROM. Copy these files to the directory "F:\WINCE500\PBWorkspaces" and remove the read-only property (if there is no PBWorkspaces directory, you can create one.).



Step2, click "Open Workspace" in the menu "File" or in interface of PB, as the following red frame indicates:







Open the project file "TQ2440" and find "TQ2440.pbxml":



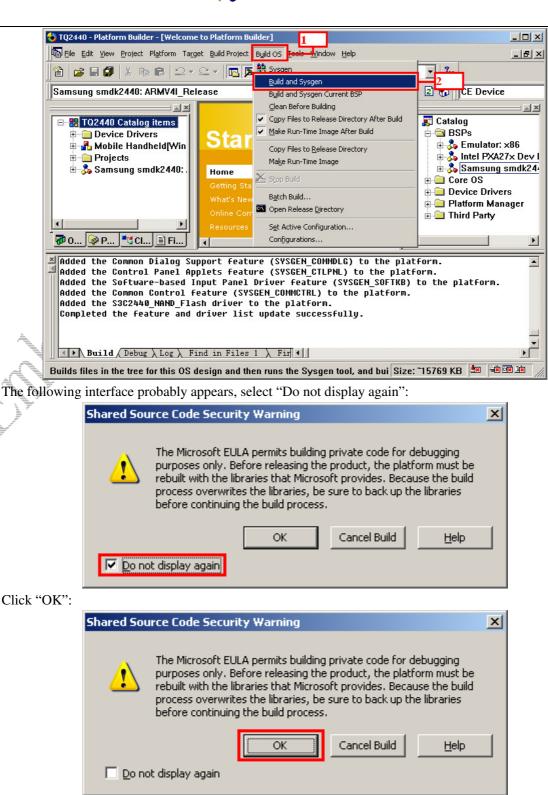


	Open Workspace
	_ 查找范围 (I): 🔁 TQ2440 🔽 🥥 🤣 📂 🖽 -
	TQ2440.pbxml 1
	文件名(M): TQ2440. pbxml 打开(D) 文件类型(T): Workspaces (. pbxml) 取消
The pro	ect: TQ2440 - Platform Builder - [Welcome to Platform Builder] Elle Edit View Project Platform Target Build Project Build OS Tools Window Help
	Samsung smdk2440: ARMV4I_Release 🔄 😫 🖀 🖄 🗅 🕙 🛛 🛛 🛣 🛣 CE Device
	Image: Start Image: Start Image: Start Image: Start
	Added the Common Dialog Support feature (SYSGEN_COMMDLG) to the platform. Added the Control Panel Applets feature (SYSGEN_CITPNL) to the platform. Added the Software-based Input Panel Driver feature (SYSGEN_SOFTKB) to the platform. Added the Common Control feature (SYSGEN_COMMOTRL) to the platform. Added the S3C2440_NAND_Flash driver to the platform. Completed the feature and driver list update successfully.
	完毕 Size: ~15769 KB 💆 地图 脑

Step3, click "Build and Sysgen" in the menu "Build" to start compiling:



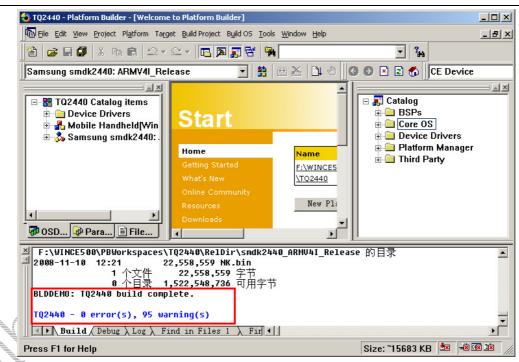




Compiling is complete:



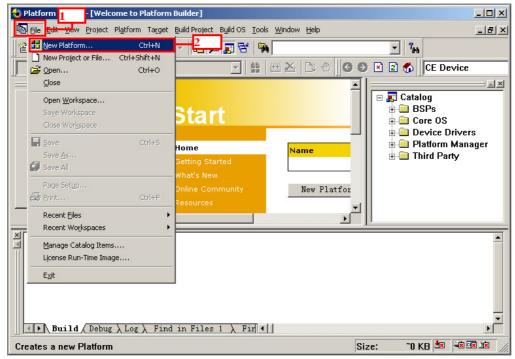




Step4, 2 kernel files "NK.bin" and "NK.nb0" are created after compiling. Only "NK.bin" is useful which is under the directory "F:\WINCE500\PBWorkspaces\TQ2440\RelDir\smdk2440_ARMV4I_Release".

5.3 Customizing user project files

The following contents introduce how to customize the user project files: Step 1, click "New Platform" in the menu "File" or in the "Start" interface:

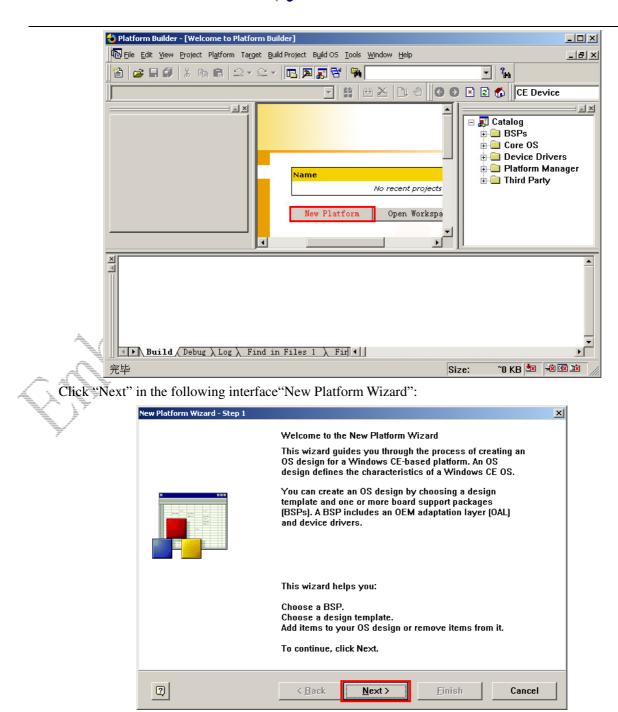


Or:

¢,







Step 2, input the project name and the location of saved files (usually adopt the default location) in the interface "Workspace Name And Location" and click "Next" to continue:



1



	- Step 2 me And Location endly name for your workspace.	×
	Name: TQ2440 Path: F:\WINCE500\PBWorkspaces\TQ2440	
2	<pre> 2 4 Control Con</pre>	Cancel

Step 3, select "SAMSUNG SMDK2440: ARMV4I" in the interface "Board Support Package (BSPs)" and click "Next" to continue:

New Platform Wizard - Step 3 Board Support Packages (BSPs) A BSP contains a set of device drivers that are	e added to your OS design.
Available BSPs: ■EMULATOR: X86 ■INTEL_PXA27X DEV PLATEORM-ARMV41 ■SAMSUNG SMDK2440: ARMV41 1	Select one or more BSPs for your OS design. Samsung smdk2440 Development Platform
	Note: Only BSPs supported by installed 2 re displayed in the list.
2 < Back	Next > Einish Cancel

Step 4, select "Mobile Handheld" in the interface "Design Template" and click "Next" to continue:



ź

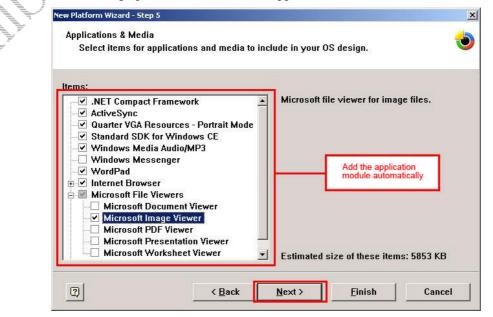


New Platform Wizard - Step 4

Design Template A design template is a pre	-defined selection	of Catalog items.
Available design templates: Custom Device Digital Media Receiver Enterprise Terminal Enterprise Web Pad Gateway Industrial Controller Internet Appliance IP Phone Mobile Handheld Set-Top Box Tiny Kernel Windows Thin Client		Choose the design template that is most closely aligned with the purpose of your target device. Provides the starting point for a range of personal digital assistants (PDAs) or mobile devices with a clamshell-and-keyboard design.
3	< Back	<u>N</u> ext > <u>F</u> inish Cancel

xI

Step 5, select the following options in the interface "Applications & Media" and click "Next" to continue:

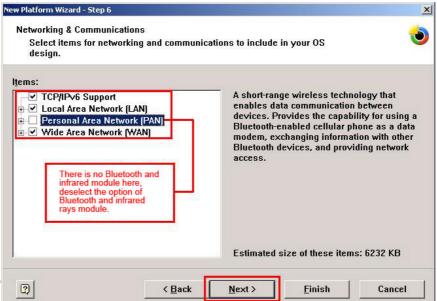


Step 6, adopt the default configurations in the interface "Networking & Communications" and click "Next" to continue:





	_			
w Pl	atfo	rm W	lizard	-

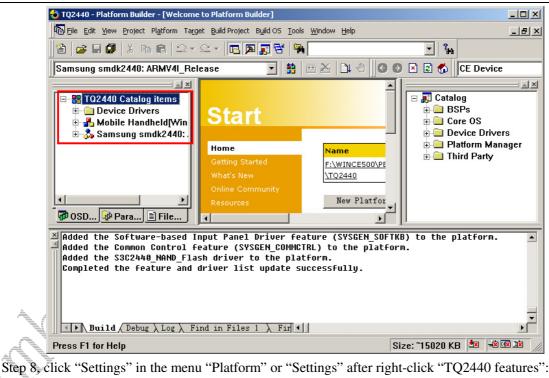


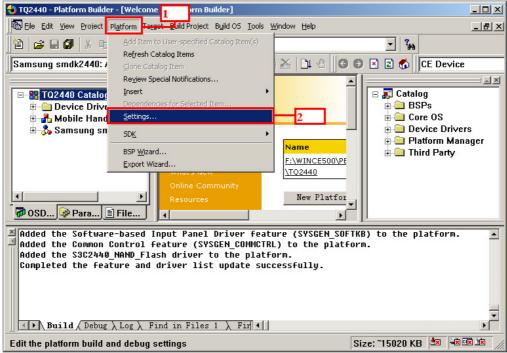
Step 7, click "Finish" in the interface "New Platform Wizard – Step 7" to complete project customizing:

New Platform Wizard - Step 7				×
Completing the Ne	w Platform W	Vizard		<u> </u>
You have successfully com	pleted the New P	latform Wizaı	d.	
You have created an OS d platform. By default, Platfo configuration and a Releas	orm Builder provid	les a Debug	gn.	
Options: • Modify build options for the design without closing this		onfigurations of y	our OS	
To close this wizard, click Fi	nish.			
				Ŧ
2	< <u>B</u> ack	<u>N</u> ext ≻	<u> </u>	Cancel







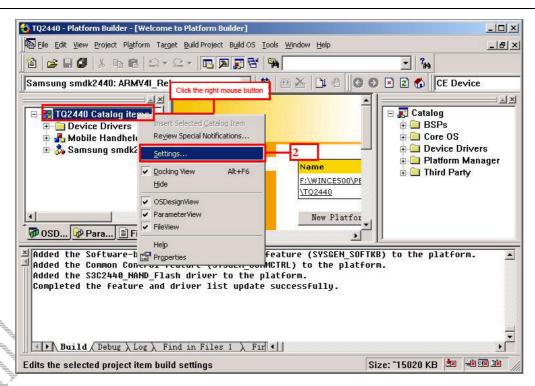


Or:

-







Step 9, select "Locate" page in the interface "Platform Settings" and select Chinese support:

latform Settings
<u>C</u> onfiguration:
Samsung smdk2440: ARMWAT P.1 ease
General Locale Build Uptions Environment Custom Build Actions Image Settings
Locales:
□ 中文(台湾) □ 中文(香港特别行政区) □ 中文(新加坡)
□ 中文(新加坡) □ 中文(中国)
<u>p</u>
De <u>f</u> ault language:
中文(中国)
[中文(中国) C_ <u>o</u> depages:3
C <u>o</u> depages:3
Codepages: 437 (OEM - United States) 708 (Arabic - ASMO 708) 709 (Arabic - ASMO 708)
C <u>o</u> depages:3
Codepages: 437 (OEM - United States) 708 (Arabic - ASMO 708) 720 (Arabic - Transparent ASMO) Clear All
Codepages: 3

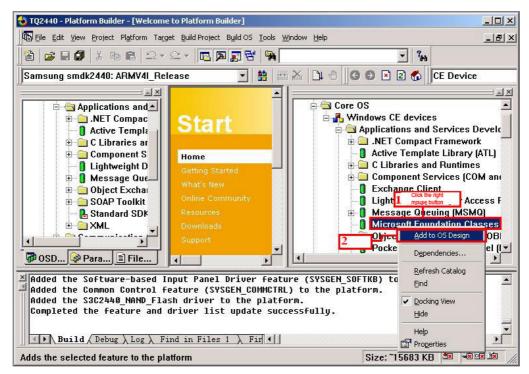
Select the page "Build Options". De-select "Enable CE Target Control Support" and "Enable KITL" and click "OK" to continue:





	Platform Settings
A Contract of the second secon	Samsung smdk2440: ARMV4I_Release
	2
	OK Cancel

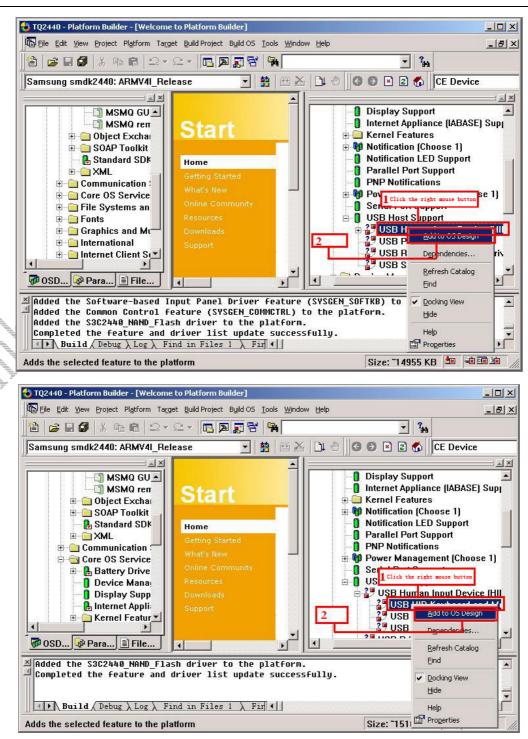
Step 10, add MFC components as the following diagram. Open "Catalog->Core OS->Windows CE devices >Applications and Services Devleopment" and right-click the option and select "Add to Platform":



Step 11, add USB keyboard and mouse supporting features. Open "Catalog->Core OS->Windows CE devices->Core OS Service->USB Host Support->USB Human Input Device (HID) Class Driver", right-click it and select "Add to Platform". Right-click "USB HID Keyboard and Mouse" and select "Add to Platform":



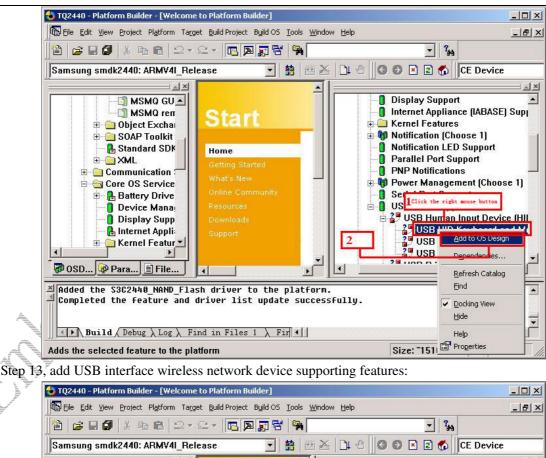


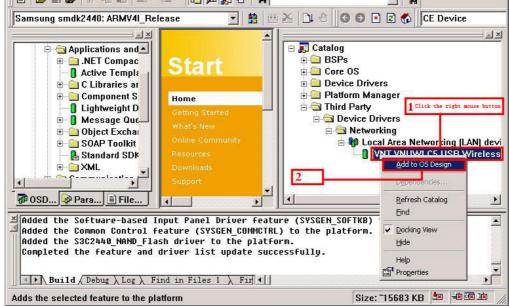


Step 12, add USB mobile storage device supporting features:







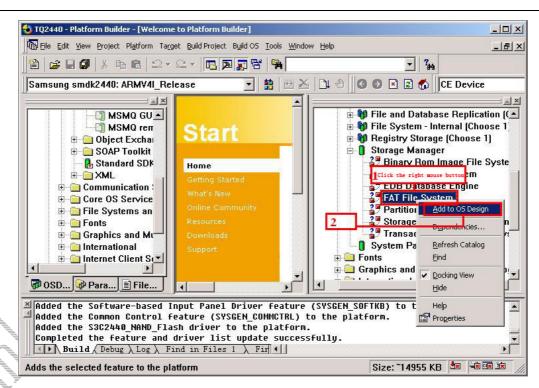


Step14, add file system supporting features. The register can be saved only when HIVE is supported by file system. The following contents introduce the process of file system configuration:

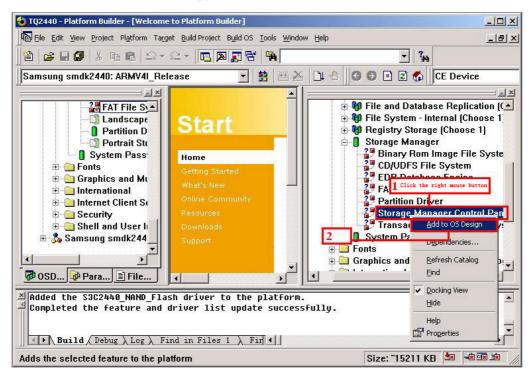
Add FAT file system supporting feature:







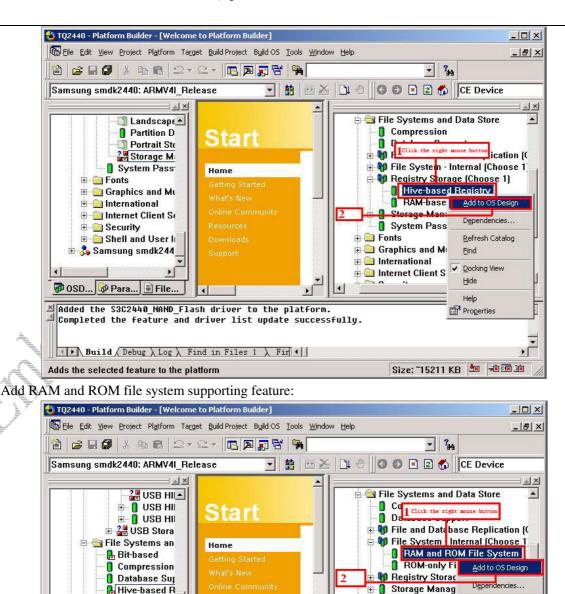
Add "Storage Manager Control Panel Applet"



Add HIVE supporting feature:







Step 15, add font and input method supporting features:

Adds the selected feature to the platform

🗄 🚦 Storage Manz

🗄 🦲 Graphics and My

🗄 📄 Fonts

7 OSD... 🐼 Para... 🖹 File..

4

System Pass

•

Added the S3C2440_NAND_Flash driver to the platform.

Build / Debug / Log / Find in Files 1 / Fir 4

System Passwi

🗄 🦲 Graphics and Mult

🗄 🧰 Internet Client Ser

🗄 🦲 International

🗄 🦲 Fonts

Refresh Catalog

-

+[

Docking View

Find

Hide

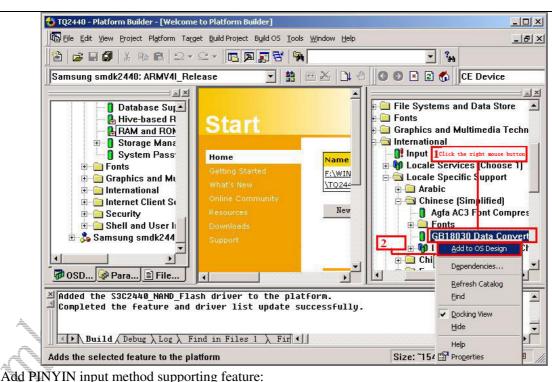
Help

-

Size: ~15352 KB 🔄 🗐 🗐







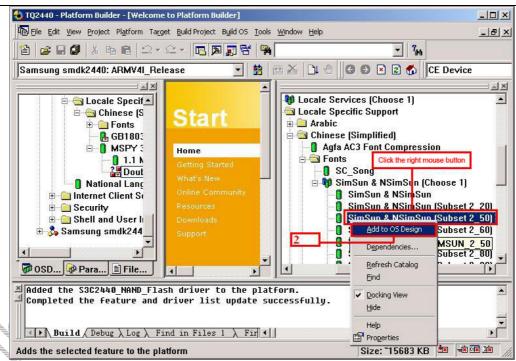
も TQ2440 - Platform Builder - [Welcome to Platform Builder] - 0 × 🚯 Eile Edit View Project Platform Target Build Project Build OS Iools Window Help _ 8 × 🎦 😅 🖬 🕼 📭 😰 그 - 오 - 🖪 🗖 📅 🖬 - 744 Samsung smdk2440: ARMV4I_Release 💽 🏥 🕮 🚵 斗 🕙 🕝 🕑 🖻 🗟 🏠 CE Device × 🗄 🔄 Locale Specific Support 🗄 🚞 Graphics and M 📥 . 🗄 🔄 International 🗄 🦲 Arabic Start Input Method 🗄 🔄 Chinese (Simplified) Agfa AC3 Font Compress 🖻 <u> </u>Locale Specif 🖻 🔄 Chinese (S 🕀 🦲 For Home GB^{1Click} the right a Name 🗄 🧰 Fonts F:\WIN 🗄 🎁 Input Method 🛙 ditor (Cho 🔒 GB1803 <u>\TQ244</u> B MSPY 3.0 for Window National Lanç 2. Double Spelling (S-2. Double Spelling (S-🗄 🦲 Internet Client St New + Security Add to OS Design 🗄 🦲 Shell and User li ±] S 🗄 💑 Samsung smdk244 Dependencies.. E Chinese Refresh Catalog 🗄 🦲 English Find 7 OSD... 🖗 Para... 🖹 File.. 4 F Docking View 4 ≝Added the S3C2440_NAND_Flash driver to the platform. Hide Completed the feature and driver list update successfully. Help Properties Build / Debug / Log / Find in Files 1 / Fir 4 Size: "15682 KB 🖄 🛋 🗐 速 Adds the selected feature to the platform

Add font supporting feature:



4





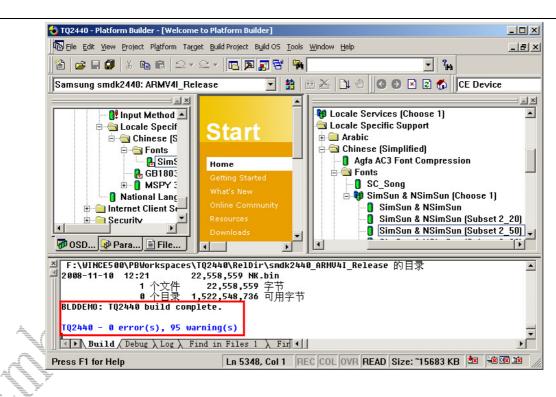
Step 16, configure network parameters: IP address, DNS, gateway and so on by modifying "platform.reg" reference key values.

も TQ2440 - Platform Builder - [F:\\FILE:	5\platform.reg *]	
Eile Edit View Project Platform Targe	t Build Project Build OS Tools Window Help	_ @ ×
🖺 😂 🖬 🕼 🐰 🖿 💼 🖾 🔹	2 - IB 🛛 🖓 😽 🙀	▼ ² Hà
Samsung smdk2440: ARMV4I_Rele	ease 🔽 😫 🖾 👗 🖱 🥝 🕖	📧 🗟 🐔 CE Device
TQ2440 parameters	"SoftwareMacAddress2"=dword:09 "SoftwareMacAddress3"=dword:18 "SoftwareMacAddress4"=dword:00 "SoftwareMacAddress5"=dword:07	File Systems and Compression Database Sup
smdk2440 smdk2440 Hardware Speci platform.bib platform.dat platform.dat platform.reg Project Specific	[HKEY_LOCAL_MACHINE\Comm\DM9CE1\Pa "EnableDHCP"=dword:0 "DefaultGateway"="192.168.1.2" "LLInterface"="" "UseZeroBroadcast"=dword:0 "IpAddress"="192.168.1.6" "Subnetmask"="255.255.255.0" "DNS"="202.96.128.166"	Generation of the second
OSD Para File.	;"EnableDHCP"=dword:1 ;"DefaultGateway"="您的开 <mark>我</mark> 在F ;"LLInterface"="" :"UseZeroBroadcast"=dword:0	Graphics and Mu
Added the S3C2440_NAND_Flass Completed the feature and d Build (Debug) Log) Fin	river list update successfully.	×
Ready	Ln 701, Col 31 REC COL OVR READ Siz	2e: ~15472 KB 🖄 🕫 💷 🏼

Step 17, compile the project. Click "Build and Sysgen" in the menu "Build" to start compiling: Compiling is complete:







5.4 Export SDK

The users can export the specific SDK for their customized platforms. SDK is the general name of head file, library file, document, platform manager and dynamic link library.

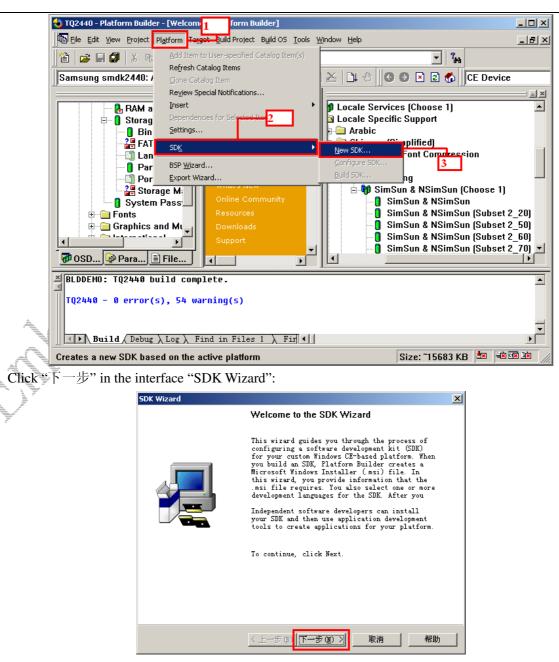
The developer can use SDK to program for a certain platform. This part of chapter introduces the process exporting SDK.

5.4.1 Configure SDK

Click "New SDK" of "SDK" in the menu "Platform":







Enter the project name and company in the interface "Product Properties" and click "T - #" to continue:



Seder

SDK Wizard	×
Product Properties Provide information to uniquely identify your SDK.	1
The .msi file for the SDK requires the following information: <u>Product name that is displayed when .msi file</u> TQ2440	
Manufacturer name: 广州天嵌计算机科技有限公司 Locale	2
U.S. English	
Product version (format: 00.00.0000) Majo <u>r</u> : 5 Minor: 0 B <u>u</u> ild: 0	
3 < 上一步 (2) 下一步 (2) 》 取消	帮助

Select the library and development languages in the interface "Development Languages" and click "T - #" to continue:

SDK Wizard Development Languages	4
An SDK can support multiple development languages and multiple frameworks for application development.	9
Select the development languages that you want your SDK to support.	
 ✓ eMbedded Visual C++ 4.0 support ✓ ATL for Windows CE applicat; ✓ MFC for Windows CE Applicati ✓ MFC for Windows CE ActiveX cont; ✓ MFC for Windows CE ActiveX cont; 	
2	
〈上一步⑫〉下一步⑭〉〉 取消 帮	助

Click "Finish" to complete SDK Wizard:

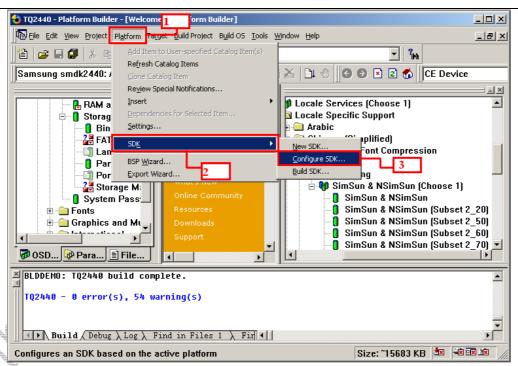
Í

SDK Wizard	×
Completing the SDK Wizard	*
You have successfully completed the SDK Wizard.	
You have configured a software development kit (SDK) for your custom Windows CE-based platform.	
Options: Close this wizard and then build your SDK. Close this wizard and continue configuring your SDK.	
To close this wizard, click <i>Finish</i> .	
<上一步 ® [Inish] 取消 帮助	

Select "Configure SDK" of "SDK" in the menu "Platform":







Configure the page "Install" in the interface "SDK Settings" as the following diagram:

5DK Settings	×
	Help Emulation CPU Transports
SDK target directory path: F:WHINDE500/PEWorkspaces/TQ2440/SDK	Bro <u>w</u> se
QS design name: TQ2440	
MSI file name:	
TQ2440_SDK.msi	J
\odot Compress files into the MSI database file (more compa	
$\mathbb C$ Leave files uncompressed in installation subfolder (f	
Launch MSI installation file after successful b	
	用(A) 帮助

Configure the page "CPU" as the following diagram:

4

DK Settings	×
Startup Servers Development Languages Additional Files Halm Install Product Properties EULA/Readme Splash Screen CPU	Emulation Transports
The following list shows which configuration of your OS design is supported for each CPU family that your SDK supports. To choose a supported configuration for a CPU family, select the entry for Configurations	
CPU fooly Configuration PARMV4I Samsung smdk2440: ARMV4I_Release	
<u></u> dit	
确定 取消 应用 (A)	帮助





Configure the page "Product Properties" as the following diagram:

SDK Settings	3
Startup Servers Development Languages Additional Files He Install Product Properties EULA/Readme Splash Screen CP	- (¹
	U Transports
The .msi file for the SDK requires the following information:	_
Product name that is displayed when .msi file	т
	1
Manufacturer name:	
广州天嵌计算机科技有限公司	
Locale	_
U.S. English	
Product version (format: 00.00.0000)	
Major: 5 Minor: 0 Build: 0	
majoz. 0 m2tos. 0 22:22. 0	

Configure the page "Development Languages" as the following diagram:

SDK Settings	×
Install Product Properties EULA/Readme Splash Screen CPU Transpo Startup Servers Development Languages Additional Files Help Emulat Select the development languages that you want your SDK to support.	
 ✓ eMbedded Visual C++ 4.0 support ✓ AJL for Windows CE applicat: ✓ MFC for Windows CE ActiveX conts ✓ MFC for Windows CE ActiveX conts ✓ .NET Compact Framework 	
确定 取消 应用 (<u>A</u>) 帮助	h

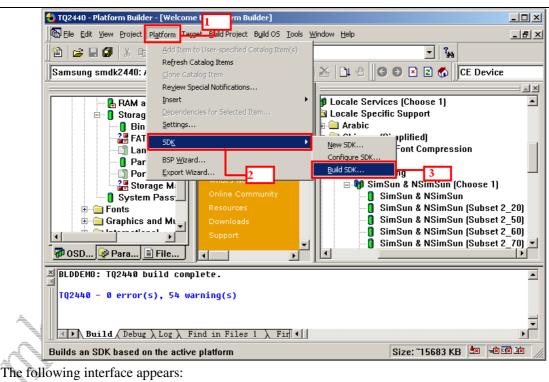
Click " \mathfrak{m} \mathfrak{z} " to finish the configuration.

5.4.2 Compiling SDK

Select "Build SDK" of "SDK" in the menu "Platform":







SDK Wizard	X
Building Your SDK Flease wait while your SDK is being built. Click Done when the MSI file construction is complete.	1
Committing database changes Creating 'required' feature Adding required files Committing database changes Rolling sysgen'ed headers and libs Samsung smdk2440: ARMV4I_Release T Remarking SDK 40: F: \WINCESOU\PEWorkspaces\TQ2440\SDK\TQ2440_SDK.msi	*
Done 取消	帮助

Click "Done" in the following diagram to finish compiling.

Í



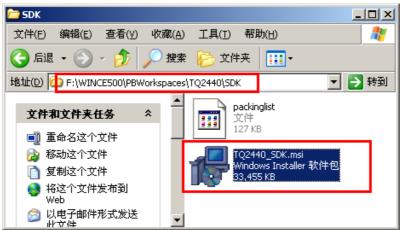


Building Your SDK				_
Please wait while :		built. Cli	ck Done when	the 🔻
MSI file construct	ion is complete.			
adding F:\WINCE50				
adding F:\WINCE50 adding F:\WINCE50				
adding F: \WINCE50				
Completed successfully	,			
Committing database cl	nanges			
MSI file construction	completed. O err	or(s), 0 war	ning(s).	
Export SDK to: F:\WING	W500\PBWorlesson	\T02440\SDK	T02440 SDK -	
Export DER CO. I. WING	Looo (i bi oi Kspaces	(IQ2440 (DDA)	(192440_DDK. m.	·· 두
•				
SDK_exported_to:				
F:\WINCE500\PBWorkspa	ces\TQ2440\SDK\TQ	2440_SDK.msi		
			_	

5.4.3 Finishing the compiling

Í

The SDK installation package appears in the directory "F:\WINCE500\PBWorkspaces\TQ2440\SDK" after compiling:



5.5 Communicating with PC synchronously by using ActiveSync

The software "ActiveSync" can be used for communication between TQ2440 and PC, and it supports file transmission, remote debugging and other functions.

5. 5. 1 Installing ActiveSync

The ActiveSync installation application is under the directory "Windows 平台工具\ActiveSync" in CD-ROM.





Step 1, double-click "ActiveSync_4.1_setup.exe" and click "下一步" to continue:



😽 Microsoft ActiveSync 4.1 InstallShi	eld Wizard	×
许可证协议	-4	
诸仔细阅读下面的许可证协议。		
[_
MICROSOFT 软件许可条款		3
MICROSOFT ACTIVESYNC 4.1		
本许可条款是 Microsoft Corpora		
]]达成的协议。请阅读本条款的内容。本 家用来接收该软件的媒体(若有)。本条	•
我接受该许可证协议中的条款(A)	1 打印(P)	
⑦ 我不接受该许可证协议中的条款(D)	2	
nstallShield		
	< 上一步(B) 下一步(N) > 取消	

Step 3, enter the information "用户名" and "单位", and click "下一步" to continue:





<mark>得 Microsoft Active5ync 4.1 Install5hield Wizard 用户信息</mark> 请输入您的信息。	×
用户姓名(<u>U</u>): <mark>Arthur</mark> 单位(<u>O</u>): 厂州天嵌计算机科技有限公司	<u>1</u>
	2
InstallShield <上一步(B)	下一步(1) > 取消

Step 4, select installation path "F:\Program Files\Microsoft ActiveSync\" and click " $\top - \#$ " to continue:

单击"下一步"安	装到此文件夹,或单击"	更改"安装到不同的	的文件夹。	
	osoft ActiveSync 4.1 安装	-201 •		
	gram Files\Microsoft Active			更改(⊆)
	Installation location	s can be modified	here.	
卷	磁盘空间		要求	
C:	8674MB	2870MB	23MB	
(E3.36/2012)	11GB	3884MB	OKB	:
🗐D;	11GB	8242MB	4012KB	1
D: E:		1426MB	OKB	Commences and
10000000000000000000000000000000000000	9981MB			
⊜ E:	9981MB			

Step 5, click " $m cy \mbox{\sc start}$ in the following interface to start installation:





	👘 Microsoft	ActiveSync 4.1 InstallSh	nield Wizard		×
	已做好安约	装程序的准备			A
		开始安装。			
		171 2432.428			And in case of the local division of the loc
	单击"安装	炭"开始安装。			
	InstallShield —				
			<上一步(B)		2消
Installing:					
	🙀 Microsoft	ActiveSync 4.1 InstallSh	nield Wizard		
	正在安装	Microsoft ActiveSync 4.1			A
		·····································			
					And in case of the local division of the loc
		InstallShield Wizard 正在安要几分钟的时间。	装 Microsoft ActiveSyr	ıc4.1,请稍候。 这需	
	157	要几分钟的时间。			
		状态:			
		V ()E) -			
	The literature				
	InstallShield —		(
			< 上一步(B)	下一步(10) > 1	(消

Click "完成" to finish the installation:





 i∰ Microsoft ActiveSync 4.1	InstallShield Wizard
Microsoft ActiveSync 4.1	
InstallShield Wizard 成功地安	装了 Microsoft ActiveSync 4.1 。 单击"完成"退出向导。
Instalishield	<上一步(B) 完成(E) 取消
打连接	X
	进行连接
	あなまなた ないた DC
	要将设备连接到此 PC:
	1. 将电缆连接到此 PC
	2. 打开设备
	3. 将设备放置在其底座中
	ActiveSync应该自动检测设备。
	如果没有自动检测您的设备,请单击"下一步"。 如果您使用的是红外线 (IR) 或 Bluetooth,请单击"帮助"。
	< 上一步 (3) 下一步 (2) > [取消] 帮助

Click "取消". The following icon appears in the taskbar:

ź

⊕<mark>,</mark> **∃**")

5.5.2 Installing synchronizing communication USB driver

Burn and start up WinCE image first. Connect PC and platform with USB wire. If there is no driver in PC, the prompt "发现新硬件" will appear, and you need to complete the installation according to the following steps. The driver installation program is under "WinCE\WinCE_4.2\SMDK2440\DRIVERS\USB\FUNCTION" of BSP package in CD-ROM.



Step 1, the following interface appears after the USB wire has been connected. Select "是,仅这一次" and click "下一步" to continue:



Step 2, select "从列表或指定位置安装(高级)" and click "下一步" to continue:



Step 3, select "在搜索中包含这个位置" and click "浏览" to locate the USB driver (under the directory "Windows 平台工具\WinCE 同步驱动" in CD-ROM). Click "下一步" to continue:





找到新的硬件向导
法到外的是FFF科书 请选择您的搜索和安装选项。
◎ 在这些位置上搜索最佳驱动程序(②)。
使用下列的复选框限制或扩展默认搜索,包括本机路径和可移动媒体。会安装找 到的最佳驱动程序。
□ 搜索可移动媒体 (软盘、CD-ROM
 ✓ 在搜索中包括这个位置 @): H:\Windows平台工具\WinCE同步驱动 ✓ 浏览 (B)
 不要搜索。我要自己选择要安装的驱动程序 (D)。
选择这个选项以便从列表中选择设备驱动程序。Windows 不能保证您所选择的驱 动程序与您的硬件最匹配。
3
〈上一步 @ 下一步 @ 〉 取消

Step 4, the following diagram appear. Select "仍然继续" in the second interface and continue:

找到新的硬件向导	
肖导正在安装软件,请稍候	
Anchor VSB EZ-Link Cable	
正在设置系统还原点并备份旧文件,以防将来需要还 原系统。	
< 上一步 (1) 下一步 (1) > 取消	





硬件安装	
	正在为此硬件安装的软件: Anchor USB EZ-Link Cable
	没有通过 Windows 徽标测试,无法验证它同 Windows XP 的相容性。(<u>告诉我为什么这个测试很重要。</u>) 继续安装此软件 会立即或在以后使系统变得不稳定。 Bicrosoft 建议您现在停止此安装,并同硬件供应商 联系,以获得通过 Tindows 徵标测试的软件。
	仍然继续 (2) 停止安装 (3)

Step 5, USB driver installing is complete. Click "完成" and ActiveSync will run automatically.







进行连接	×
	进行连接
	要将设备连接到此 PC:
	1. 将电缆连接到此 PC
	2. 打开设备
	3. 将设备放置在其底座中
	ActiveSync 应该自动检测设备。
	如果没有自动检测您的设备,请单击"下一步"。 如果您使用的是红外线 (IR)或 Bluetooth,请单击"帮助"。
	< 上一步 (B) 下一步 (B) > [取消] 帮助

5.5.3 Utilize ActiveSync synchronizing software

This part follows the upper steps.

1

Instruction: Make sure to remove all the U disk, SD card and other devices before using synchronizing function of WinCE.

Step 1, select "是" in the interface "合作关系" and click "下一步" to continue:

新建合作关系		×
	建立合作关系	
	要想在移动设备和此计算机之间同步信息,必须先在 二者之间建立合作关系。	
	要建立合作关系吗?	
	 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2	
	C 否()	
	无需同步信息。将移动设备设置为"来宾",以便 在移动设备和此计算机之间复制或移动信息。	
	2	
	< 上一步 @ 下一步 @ > 取消 帮助	

Step 2, click "下一步" in the interface "同步设置" and continue:





要同步特定类型的信息,认	青选中其复选框。要停止同]步,请清除其复选框	0
移动设备	台式计算机		1
□ 🧠 Pocket Access □ 🔁 文件	Microsoft 数据库 同步文件		
L	the	chooses se two	
	opt	ions	

Step 3, click "完成" in the interface "设置完毕" to finish the configuration of ActiveSync



Step 4, the device connection interface "TQ2440" appears:

Í





	🔂 Microsoft /	ActiveSync		
	文件(E) 视图	(V) 工具(I) 帮!	助(円)	
	 ② ② ○ ○	日本 详细信息 浏览	E 选项	
ſ	TQ2440			
ſ	己连接	•		
Ī	信息类型	状态		
A				
=				
4				
7				

The synchronizing icon in the lower right corner turns green:



Step 5, the menu "移动设备" appears in "我的电脑":

💈 我的电脑			
文件(E) 编辑(E) 查看(V)	收藏(<u>A</u>)	工具(I) 帮助(H)	1
🕞 后退 🔹 💮 🕤 🏂	▶ 搜索	🦻 文件夹 🛛 🎹 ▾	
地址(D) 😼 我的电脑		•	▶ 转到
系统任务	*	www.pvD 驱动器 (J:) 其他	
▶ 更改一个设置 其它位置	*		_

If select "否" in Step1:

CateDig



新建合作关系		×
	建立合作关系	
	要想在移动设备和此计算机之间同步信息,必须先在 二者之间建立合作关系。 要建立合作关系吗?	
	 ○ 是(Y) 建立合作关系,以便在移动设备和此计算机之间 同步信息。 ○ 蘆(0) 1 无需同步信息。将移动设备设置为"来宾",以便 在移动设备和此计算机之间复制或移动信息。 	
	2	
	<上一步(B) 下一步(B) > 取消 帮助	

The connection interface "来宾" appears:

🛞 Micro	soft Acti	veSync		
文件(E)	视图(⊻)	工具(<u>1</u>)	帮助(<u>H</u>)	
관 同步	F 🕑 F	目程安排	📡 浏览	
来宾				
己连接				U
				隐藏详细信息 🛠
信息类型	1	状态		

Instruction: the upper steps appear only in the first using. In the further using, the user need only modify the device name with other settings unchanged.

5. 5. 4 Transferring file with ActiveSync

Step 1, after the previous operation, the platform has been synchronized with PC. Click "移动设备" and all directories in platform could be found:





🔋 移动设备						<u> </u>
文件(E) 编辑(E) 查看(V)	收藏(<u>A</u>)	工具(<u>I</u>) 兼	ŝ助(∐)			1
😋 后退 🔹 🕥 🕤 🏂	🔎 搜索	ြ у ў́не	•			
地址(D) 🔋 移动设备					•	转到
文件夹任务	*		D			1
🗐 重命名这个文件夹 🍙 移动这个文件夹		Application Data	EmbedSky	My Documents	Program Files	
 ☐ 复制这个文件夹 ★ 删除这个文件夹 			\supset			
		Temp	Windows	网络	控制面板	
甘它位署	* -					

Step 2, copy a file from PC to the upper directory "EmbedSky". The following interface appears:

文件转换
当移动设备和该台式计算机进行同步时,Microsoft ActiveSync 可能需要转换文件。
┘ 注意:转换后的文件可能不包括初始文件中的所有信息。
(

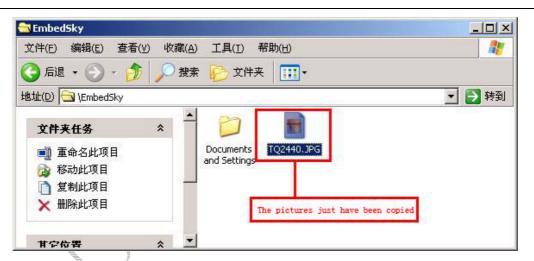
Click "确定" to continue:

复制并转换为移动设备格式	×
TQ2440	
转换(无)	
正在复制(从~media~到~EmbedSky~)	
	取消

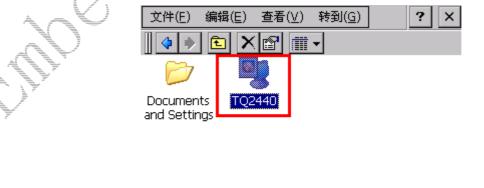
Step 3, after copying process complete, the file appears in the directory "EmbedSky":







Step 4, the copied file could be found under the directory "EmbedSky" in "我的电脑": (instruction: The size of free space in Nand Flash is about 30M, please be aware of the free space.)





5.6 Capturing screen with ActiveSync and Platform Builder connection

Make sure the connection and synchronization between platform and PC is OK.

5.6.1 Configuring platform

Connect USB wire and net line, and start up WinCE in platform. If the platform doesn't get connected to USB device, please restart the platform.

5.6.2 Configuring PC

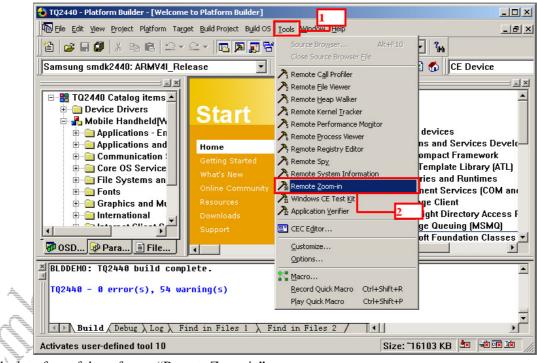
1

Use the remote image scaling tools in PB.





Step 1, click "Remote Zoom-in" in the menu "Tools" in PB:



The interface of the software "Remote Zoom-in":

Í

😽 Windows CE Remote i	Zoom-in	
<u>File View Connection H</u>	<u>t</u> elp	
	🛨 🎯 Windows CE Default Platform	
	<u>OK</u>	
Ready	Disconnected	

Step 2, click "Cancel" in the upper diagram, and click "Configure Windows CE Platform Manager" in the menu "Connection":





[s CE Remote Zoom 1		
	<u>F</u> ile ⊻iew			
		Connect to Device	100>	
		Disconnect from Device		
		Configure Windows CE Platform Manage	er <u>2</u>	
A				
M	Configure W	indows CE Platform Manager 🛛 🛛 🛛 🛛	Disconnected	

Select "Add Device" and add a new device named "TQ2440".

1

Windows CE Platform Manager Configuration	
Select a platform or device to	
🛨 🎯 Windows CE Default Platform	Add Device
	Delete
	'roperties
	About
ŪK	



Í



	Windows CE Platform Manager Configuration
	Select a platform or device to
- (mark)	Windows CE Default Platform Add Device Default Darice Delete TQ2440 'roperties About About
	ŪK

Click "Properties" or double-click "TQ2440", and the configuration interface "Device Properties" appears. Select "Microsoft ActiveSync" in the option box "Transport":

Eelect a platform or device to	Add Device Delete Troperties About
--------------------------------	---





Device Properties	
Device	
TQ2440	
Select a transport and a startup server. Choose Terverify that you can establish a connection to your device with the selected transport.	target
Startup Microsoft ActiveSync	nfigure
3 OK Can-	4

Click "OK" to finish the configuration:

Í

	Windows CE Platform Manager Configuration	
	Select a platform or device to	
	☐ ∰ Windows CE Default Platform >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	<u>A</u> dd Device <u>D</u> elete 'roperties <u>Ab</u> out
Step 3, click the icon "	", in the upper left corner.	



Í



🖓 Windows CE Remote Zoom-in	
<u>File View Connection Help</u>	
90 10 10 10 10 10 10 10 10 10 10 10 10 10	
Connect to device Disconnected	11.
Or click "Connection->Connect to Device":	
Windows CE Remote Zu	
Port En En Image: Second sec	
Configure Windows CE Platform	
Connect to device Disconnected	1

The following interface appears after clicking "connect". Select "TQ2440" and click "OK" to continue:





50	elect a Windows CE Device
	⊡-∰ Windows CE Default Platform
Alternation and the second sec	2 <u>OK</u> <u>Cancel</u>

Step 4, PC and the platform start connecting after the upper operation (the net wire needs to be connected):

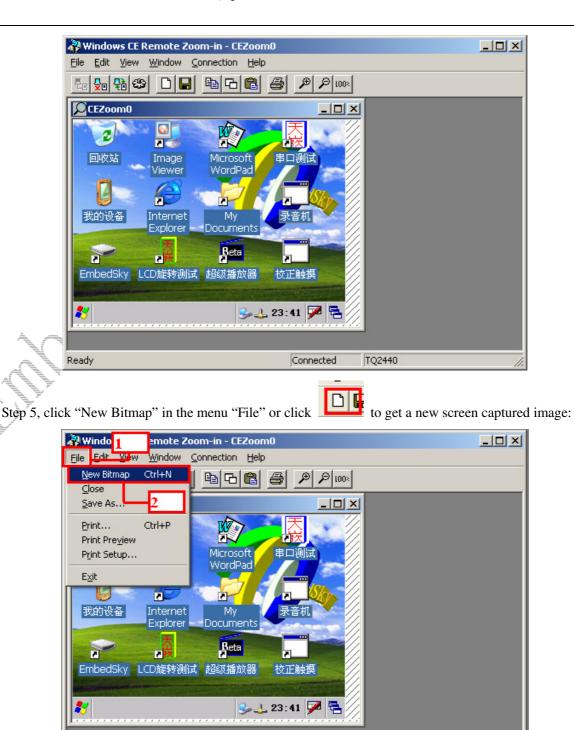
Con	necting to device
	ice 2440
	Establishing platform manager connection to device
	(<u>Cancel</u>)

Connecting is complete:

Í







Or:

Download bitmap into new window

1

Connected

TQ2440







Get the second screen captured image:

Í

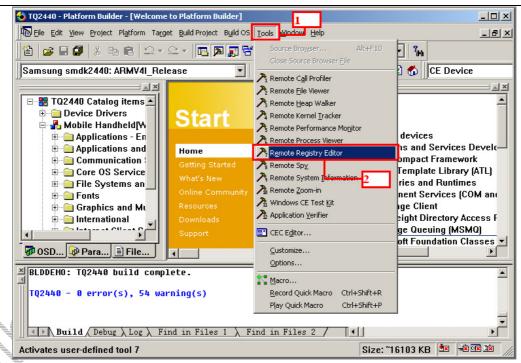


5.7 Check the platform register information based on connection between ActiveSync and Platform Builder

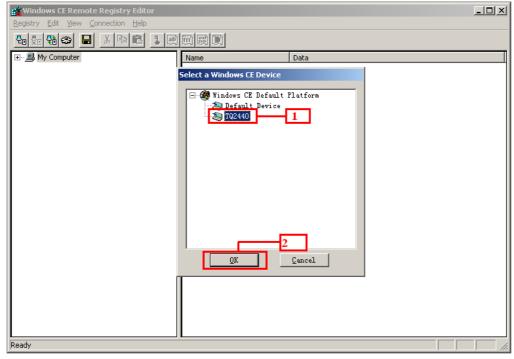
Make sure the connection and synchronization between platform and PC is OK. Step1, open "Remote Registry Editor" in the menu "Tools" in PB and run remote register editor:







Step2, the following interface appears. All information has been set previously. Click "OK" to continue:



Make sure the net wire has been connected:

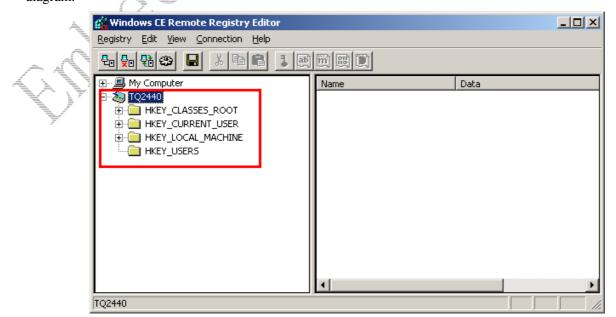
4





С	onnecting to device
	Device
	Establishing platform manager connection to device
A	
<i></i>	

Step3, open the device file "TQ2440" after connection complete. The register is shown in the following diagram:



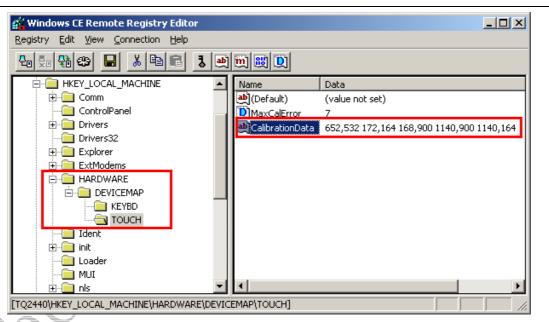
The touch screen register value:

-



Í





What we introduce here is the basic operation of WinCE development. There are a lot more applications about PB and platform connection, which will introduced in the following examples.





Chapter 6 Qt/Embeded graphics development

Qt is application development frame supporting multi-Operating System. The development language is C++. Qt provides a unified and exquisite graphics program interface and a unified network and database operation program interface.

Qt is given to the developer in the form of SDK (software development kit), including graphics designer, Makefile designer, font internationalization tool and cross-platform C++ class libraries

Qt resources:

 \geq

Trolltech homepage: http://www.trolltech.com

FTP supporting anonymous access: <u>ftp://ftp.trolltech.com</u>

Newsgroup server: nntp.trolltech.com

Non-official Qt document Chinese translation group: http://www.qiliang.net/qt/index.html

Qt/Embedded is a C++ SDK customized for user graphics interface and application development. It supports embedded Linux on different processors.

The resources for building Qt/Embeded development environment:

- Tmake kit: tmake-1.11.tar.gz
 - Used for creating project Makefile;
 - Qt/Embeded installer: qt-embeded-2.3.7.tar.gz
 - Used for installing Qt/Embeded;
- > X11 version installer of Qt: qt-x11-2.3.2.tar.gz
 - Used for creating some necessary tools;
- Qtopia installer: qtopia-free-1.7.0.tar.gz
 - Provide graphics interface for handheld device;
- Script build and script setenv
 - They are respectively compiling script and path configuration script.

There is a rule for selecting the tool package: the version of Qt for X11 needs to be older than Qt/Embeded. Because the source file created by the tools "uic" and "designer" of Qt for X11 installer will be compiled together with Qt/Embeded library. A more updated version of Qt for X11 might encounter compatibility problem with Qt/Embeded library.

Qt/Embeded source code package is namely the file "Qte.tar.bz2" under the directory "Linux \Linux-2.6.13\" in CD-ROM. After decompression, the file is located at:

- > X86 version: /opt/EmbedSky/Qte/x86-qtopia/
- > The ARM version supporting USB mouse and keyboard: /opt/EmbedSky/Qte /mouse-qtopia/
- > The ARM version supporting touch screen: /opt/EmbedSky/Qte /touch-qtopia/

(caution: the upper 3 versions get different compiling scripts, but their source codes are the same. What's more, please use the cross-compiler version 3.3.2)



6.1 Simulating Qt/Embedde in PC

6.1.1 Configuring the running evironment

The Qt/Embeded simulation in PC needs the support of Qt/Embeded library files. The user needs to modify the file "/etc/ld.so.conf" to fit it to Qt/Embeded development platform. (although Redhat includes Qt library, the version might not be suitable.)

#gedit /etc/ld.so.conf //modify the contents as follows:

```
/opt/EmbedSky/Qte/x86-qtopia/qt/lib
```

/opt/EmbedSky/Qte/x86-qtopia/qtopia/lib

/usr/kerberos/lib

/usr/X11R6/1ib

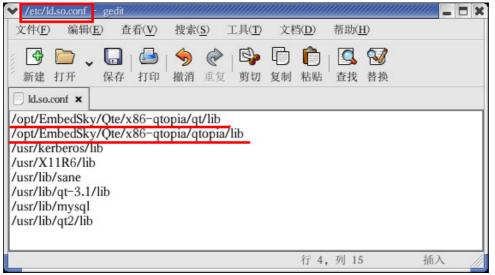
```
/usr/lib/sane
```

/usr/lib/qt-3.1/lib

/usr/lib/mysql

/usr/lib/qt2/lib

Or as the following diagram:



5.1.2 Installation and compiling

Executing the following commands to install and compile the tool scripts provided by us: #cd /opt/EmbedSky/Qte/x86-qtopia/ #./build





root@HJ	l:/opt/Embe	dSky/Qte/x8	36-qtopia				3
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)		
				e/x86-qto	pia/		*
root@HJ	x86-qtop	ia]# ./bu	i ld				
							1
							~

It is needed about 20 minutes for installing and compiling.

5.1.3 Enable the created library

#ldconfig //Enable the modified file "/etc/ld.so.conf". V root@HJ:/opt/EmbedSky/Qte/x86-qtop 转到(<u>G</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 帮助(H) 文件(E) edit.o bookmarkeditimpl.o mainwindebug.o ../../kong-embed/kdesrc/khtml/ecma/li 🔺 bkjs_html_i.la ../../konq-enbed/kdesrc/kjs/libkjs.la -lpcre -lpcreposix ../../ko nq-embed/kdesrc/khtml/libkhtml_i.la make[5]: Leaving directory /opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed/src' make[4]: Leaving directory /opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed/src' make[3]: Leaving directory /opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed/src' make[3]: Entering directory /opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed/src' make[3]: Nothing to be done for `all-am'. make[3]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed' make[2]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed' make[2]: Entering directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em' make[2]: Nothing to be done for `all-am'. make[2]: Leaving directory `/opt/EnbedSky/Qte/x86-qtopia/konq-em' make[1]: Leaving directory `/opt/EnbedSky/Qte/x86-qtopia/konq-em' [root@HJ x86-qtopia]# ldconfig [root@HJ x86-qtopia]#

5.1.4 Simulating Qtopia in PC

#. set-env //Configuring environment parameters to support Qtopia simulation. (caution: The blank character between "." and "set-env" is indispensable!)

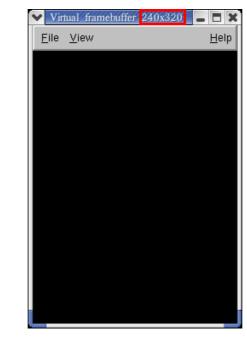
#qvfb & //The default display size is 240×320 .





🔽 root@HJ:/opt/EmbedSky/Qte/x86=qtopia	X
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)	
<pre>make[4]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed/src make[3]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed/src make[3]: Entering directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed' make[3]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed' make[2]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed' make[2]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed' make[2]: Entering directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/ make[2]: Nothing to be done for `all-am'. make[2]: Nothing to be done for `all-am'. make[2]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em' make[1]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em'</pre>	and the second second
[root@HJ x86-qtopia]# ldconfig [root@HJ x86-qtopia]# .set-env [root@HJ x86-qtopia]# qvfb &	
[1] 10366 [root@HJ x86-qtopia]# Using display 0	11/

The running interface of "qvfb":



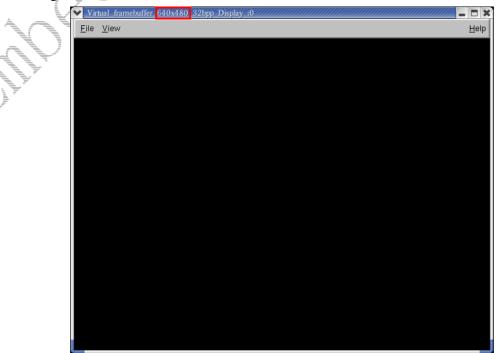
Enter the following command to modify the display size to 640×480 : #qvfb –width 640 –height 480 &





▼ root@HJ:/opt/EmbedSky/Qte/x86-qtopia	>
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)	
<pre>make[2]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed make[2]: Entering directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em' make[2]: Nothing to be done for `all-am'. make[2]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em' make[1]: Leaving directory `/opt/EmbedSky/Qte/x86-qtopia/konq-em' [root@HJ x86-qtopia]# ldconfig [root@HJ x86-qtopia]# . set-env [root@HJ x86-qtopia]# . set-env</pre>	I' 🗖
[1] 10366 [root@HJ x86-qtopia]# Using display 0 [1]+ Done gyfb	
[1]+ Done qvfb [root@HJ x86-qtopia]# qvfb -width 640 -height 480 &	
[1] 10388 [root@HJ x86-qtopia]# Using display 0	

The running interface.



Execute the command:

#qpe







The simulation interface:



6.2 Programming

6.2.1 Hello routine

Find the routine "hello" under the directory "/opt/TQ/Qt/x86/", or write a program referring to the routine "Hello". (this part of chapter does not introduce how to write a program, but how to compile routine.)



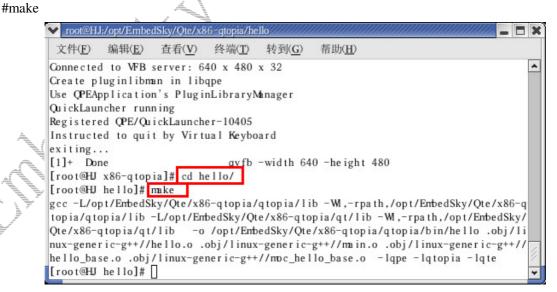
6.2.2 Configuring environment parameters

Execute the script "set_env". (caution: Complete the steps in "5.1.2" and "5.1.3" before compiling the routine "Hello".)

#cd /opt/EmbedSky/Qte/x86-qtopia/

#. set-env (caution: The blank character between "." and "set-env" is indispensable!)

#cd hello



The executable file "hello" is created under the directory "/opt/EmbedSky/Qte/x86-qtopia/qtopia/bin/" after compiling is complete.

6.2.3 Making desktop starter file

Create a new text file, and add the following contents (the following contents include: program name, icon name and so on). Change the file name into "xxxx.desktop", and save the file to the directory "\$QPEDIR/apps/Applications/".

Take the application "hello" for example:

[Desktop Entry] Comment=A Hello Program Exec=hello Icon=hello Type=Application Name=Hello2440

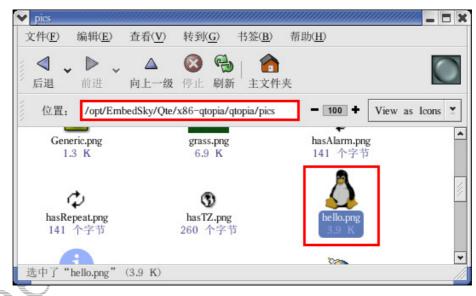
6.2.4 Making the icon

We make a 32×32 icon of PNG format. The naming method is the same as the one naming "xxxx.desktop" introduced in the upper contents. We here name the icon "hello.png" and copy it to the directory





"/opt/EmbedSky/Qte/x86-qtopia/qtopia/pics/":



6.2.5 Copying Hello.desktop

Í

Copy the finished file "hello.desktop" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/apps/Applications/". Copy the desktop starter "Hello2440" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/ apps/Applications/":

 Applicatio 	ns								_ 🗆 X
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	转到(<u>G</u>)) †	书签(<u>B</u>)	帮助(<u>H</u>)			
▲ 后退	▶ 、 前进	▲ 向上一级		副新	一 主文件	夹			\bigcirc
位置:	/opt/Emb	edSky/Qte/	x86-qtop	ia/qto	pia/apps/	Applications	- 100 +	View as	Icons 🛓
		[•
Calculator	r	c [alendar	ĩ		Clock			
Contacts		He	llo2440			Help Brows	er		
		ſ							▼

6.2.6 Running hello solely

#qvfb –width 640 –height 480 & #hello –qws





✓ root@HJ:/opt/ 文件(<u>F</u>) 编辑				帮助(<u>H</u>)		
[1]+ Done		qv fb	-width 64	0 -height 4	480	
[root@HJ x86-	gtopia]# cd			0		
root@HJ hell	1 1					
		86-gtopia	/ɑtopia/li	b -Wrpa	th,/opt/EmbedSky/Q	Dte/x86-a
					-Wrpath./opt/H	•
					a/qtopia/bin/hello	
		•			o .obj/linux-gener	
· ·				•	-lqpe -lqtopia -l	~
root@HJ hell					iqpe iqtopia i	iqie
1] 10443	0]# dv10 -wi	utii 040 -	neight 460	o de		
	1.4	1 0				
root@HJ hell	o]# Using di	splay 0				
hello -qws						
Connected to						
Øserver Socket	: failed to	bind or 1	isten to t	he socket		
Create plugin	libman in li	bqpe				3

The running interface.

Í

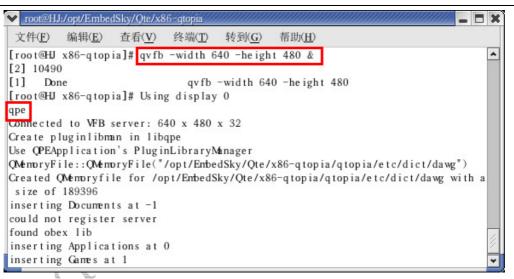
Virtual framebuffer 640x480 32bpp Display :0	_ 🗆 X
<u>E</u> ile <u>V</u> iew	<u>H</u> elp
🔀 Test my first Qtopia Application	• •
Hello	
Hello, Qtopia world!	
rielo, stopia work:	

6.2.7 Running hello in Qtopia

#qvfb –width 640 –height 480 & #qpe







The running interface:



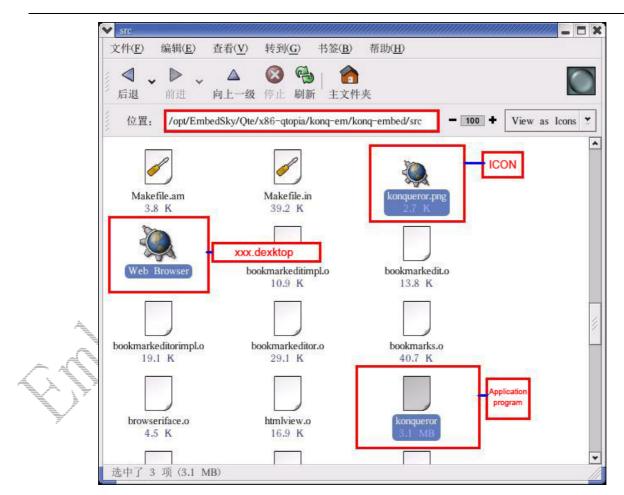
6.3 Transplanting Web explorer

The Web explorer has been compiled when compiling Qtopia. It is under the directory "konq-em". The following contents introduce how to add Web explorer to Qtopia:

Step1, find following 3 files under the directory "konq-embed/src" of "/opt/EmbedSky/Qte/x86-qtopia/konq-em":





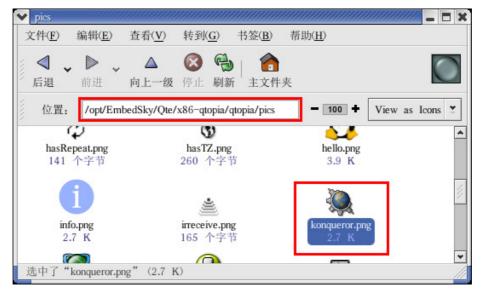


File instruction:

Í

- "konqueror.png" is the icon file of Web explorer.
- "Web Browser" is the desktop starter of Web explorer.
- "conqueror" is the application of Web explorer.

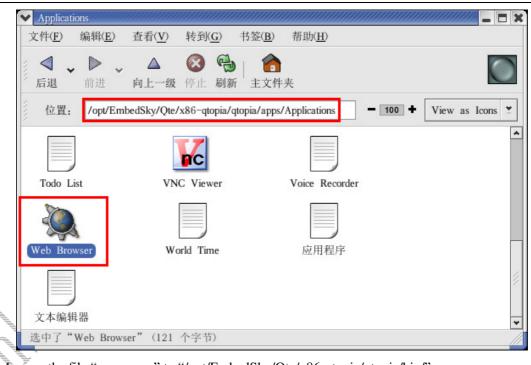
Step2, copy the file "konqueror.png" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/pics/":



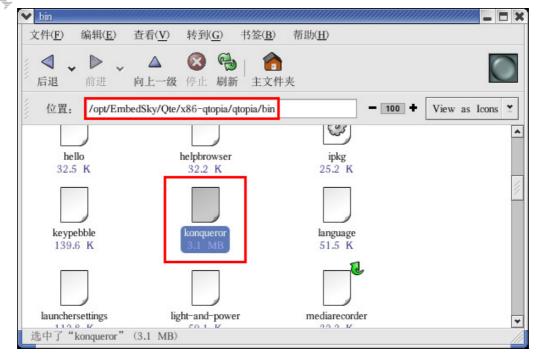
Step3, copy the file "Web Browser" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/apps/Applications/"







Step4, copy the file "conqueror" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/bin/":

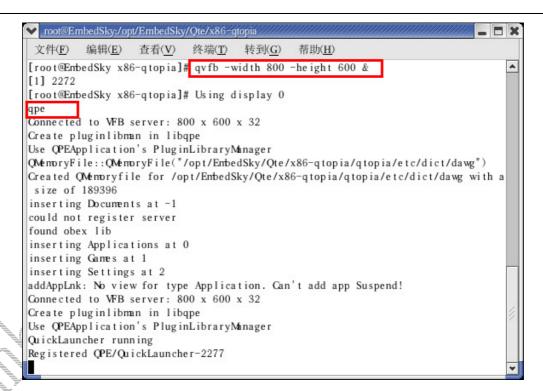


Step5, executing: #qvfb –width 800 –height 600 & #qpe

Í







Simulating qvfb:

4



The following errors appear in the first opening of Web explorer



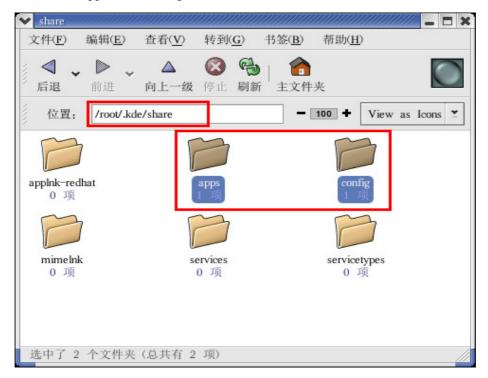


Y							
■ 应	用程序	<u>e</u>	游戏	Se St	ettings		Documents
	23		Ø		Δ	?	- Ke
Calculator	Calendar	Clock	Contacts	File Manager	Hello2440	Help Browser	Image Viewer
4		al Error	\sim	-	Anc		, 🍭
World Time	文z	or you execu directory KE executable i share\apps\k and	ne KDEDIR envi ite this binar DEDIR points 1 is located the Chtml\css\html	ry in the con to to or in t ere have to b	rrect path. I the directory	n the 🗌	
		share∖config Exiting		<u>OK</u>	ß		
				OK	5		

Solution:

Í

1, create 2 directories "apps" and "config" under "/root/.kde/share/":



2, create the directory "khtml" under "apps" and create the directory "css" under the directory "khtml". Copy the file "html4.css" under "/opt/EmbedSky/Qte/x86-qtopia/konq-em/konq-embed/kdesrc/khtml/css/" to





"/root/.kde/share/apps/khtml/css":

✓ css			×
文件(<u>F</u>)	编辑(<u>E</u>) 查看(<u>V</u>)	转到(<u>G</u>) 书签(<u>B</u>) 帮	助(<u>H</u>)
■ 「」 「」 「」 「」 「」 「」 「」 「」 「」 「」 」 「」 」	▶ ▲ 前进 向上一级	 	\bigcirc
位置:	/root/.kde/share/app	s/khtml/css – 100	I
html4.css 8.2 K			
选中了"	html4.css" (8.2 K)		

3, copy the file "charsets" under "/opt/EmbedSky/Qte/touch-qtopia/konq-em/konq-embed/kdesrc/kdecore/" to "/root/.kde/share/config/":

>	config										×
	文件(<u>F</u>)		编辑(<u>E</u>)	查看(<u>V</u>)	转到(<u>(G</u>)	书签(<u>B</u>)	帮助(<u>H</u>	D		
11111	《 后退	~	▶ ↓	. 🛆 向上一级	③ 停止	同新	合 主文作	十 夹		\subset	
	位置	Ϊ.:	/root/.k	de/share/conf	īg			100 +	View as	Icons	¥
	that K										
	选中了	" c	harsets "	(4.8 K)							

4, the Web explorer can be accessed successfully after the upper operation.

Access "<u>www.embedsky.net</u>" by using Web explorer:





天的	和技社区 ARM9开发社区,ARM技术交流i	论坛,Embe	ded,嵌入b	式,ARM9开发板,S3C2410.S3C2440 J
	n 😚 😳 👌 🔍 🖏			,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
nttp://	/www.embedsky.net/bbs/			
Y.	天嵌产品各种资料下载中心	(ſ	by runner547 – 2008–7–6 17:52
Boot	loader论坛			
	版块	主题	帖数	最后发表
	U-boot定制开发 u-boot 的移植,编译,硬件驱,基于 于U-boot功能扩展等	0	0	从未
Linu	x开发论坛			
	版块	主题	帖数	最后发表
	Linux 定制开发 Linux内核移植,修改,编译;相关 硬件驱动程序开发等。	3	6	Makefile文件的编写规则(翻译版 by 亚瑟王 – 2008–7–2 15:23
	Linux 应用程序开发 Linux可视化界面,Qt/Qtopia等;基 基于Linux的嵌入式应用程序开发相关 关	2	2	嵌入式∟inux 调试 – GDB by mayi – 2008–6–23 13:03
Winc	e 开发论坛			
Mario	版块	主题	帖数	最后发表
		土趔	帕釵	最后反表

6.4 Building development environment based on ARM

We take touch screen version for example here, the same as operations of keyboard and mouse version. Make sure the cross-compiler version is "3.3.2". The user could enter "arm-linux-gcc –v" in hyper terminal to check the version. If it is not right, please consult "2.5.1" to install the cross-compiler correctly.

6.4.1 Installing development environment

#cd /opt/EmbedSky/Qte/touch-qtopia/ #./build

6.4.2 Make running script

-

Make the running script to fit Qtopia to the platform. Qtopia running script:





Vopt/E	mbedSky/i	root_gtopia_tp/	bin/qtopia	n – gedit					_
文件(<u>F</u>)	编辑(<u>E</u>) 查看(<u>V</u>)	搜索(S) 工具(<u>T)</u> 文	档(<u>D</u>)	帮助(]	<u>H</u>)	
新建		保存 打印	-	② 5 重复 剪		₽ わい おいらう	公 查找	☑ 替换	
@ qtopia	1 × 🕘 n	ın_hello 🗙							
		/IE=/root IR=/opt/qt							
export export export #export	set KDE set QWS set QWS set QWS	DIR=/opt/q DIR=/opt/k S_KEYBOA S_MOUSE_ /S_MOUSE H=\$OPEDII	de RD="US PROTO _PROTO	="TPane O="USB:	:/dev/te	ouchse)"	
export export #export export export export	set KDE set QWS set QWS set QWS set PATI set LD	DIR=/opt/k S_KEYBOA S_MOUSE_ /S_MOUSE H=\$QPEDIF LIBRARY	de RD="U: PROTO PROTO Vbin:\$I PATH=\$	="TPane O="USB: PATH \$OTDIR/	:/dev/te /dev/in /lib:\$Ql	ouchsc put/mo	ouse0")" 	
export export #export export export	set KDE set QWS set QWS set QWS set PATI set LD	DIR=/opt/k S_KEYBOA S_MOUSE_ /S_MOUSE	de RD="U: PROTO PROTO Vbin:\$I PATH=\$	="TPane O="USB: PATH \$OTDIR/	:/dev/te /dev/in /lib:\$Ql	ouchsc put/mo	ouse0"	"	

The command lines "#export set QWS_MOUSE_PROTO="USB:/dev/input/mouse0" and "#export set QWS_MOUSE_PROTO="TPanel:/dev/touchscreen/0" are corresponding to selecting USB mouse or selecting touch screen. If the user needs qtopis start-up information to be printed, execute the command "> /dev/null 2>/dev/null".

run_hello running script: (it is used when running "hello" solely)

/opt/EmbedSky/root_gtopia_	tp/bin/run_hello – gedit			_ 🗆 X
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>	 搜索(<u>S</u>) 工具(<u>T</u>) 	文档(<u>D</u>) 养	晋助(<u>H</u>)	
● qtopia ★ ◎ run_hello ★ #!/bin/sh	▶ ��	□ □ □ [复制 粘贴]	 查找 替换 	
export set HOME=/root export set QTDIR=/opt/export set QPEDIR=/opt export set KDEDIR=/opt export set QWS_KEYBO export set QWS_MOUSI #export set QWS_MOUSI #export set PATH=\$QPEL export set LD_LIBRARY /opt/qtopia/bin/hello -qy	/qtopia /kde DARD="USB:/dev/inj E_PROTO="TPanel:// E_PROTO="USB:/d DIR/bin:\$PATH / PATH=\$OTDIR/lit	dev/touchscree ev/input/mous b:\$OPEDIR/lit	se0"	
		行 12,	列 51 插	λ //

6.4.3 Compiling hello for ARM

cd /opt/EmbedSky/Qte/touch-qtopia/

#. set-env





cd hello

#make

The executable file "hello" is created under the directory "/opt/EmbedSky/Qte/touch-qtopia/qtopia/bin/" after compiling.

6.4.4 Installing hello

Copy the executable file "hello" and environment configuration parameters and running script of "hello" to U disk:

#mount /dev/sda1 /mnt/usb //This USB directory is for mounting U disk.

#cp /opt/EmbedSky/Qte/touch-qtopia/qtopia/bin/hello /mnt/usb//Copy the executable file "hello"#cp /opt/EmbedSky/Qte/touch-qtopia/hello/run_hello /mnt/usb//Copy the running script of "hello"#cp /opt/EmbedSky/Qte/touch-qtopia/hello/hello.desktp /mnt/usb//Copy the icon file of "hello"#umount /mnt/usb//Copy the icon file of "hello"Copy "hello" to platform: (insert the U disk containing files of "hello" to USB Host interface on platform)#mount /dev/sda1 /mnt//Copy "hello" to "/opt/qtopia/bin "#cp /mnt/hello /opt/qtopia/bin//Copy the running script to "/opt/qtopia/bin "#cp /mnt/run_hello /bin//Copy the running script to "/bin"

//Copy

the

icon

file

to

#cp /mnt/hello.desktop /opt/qtopia/apps/Applications

"/opt/qtopia/apps/Applications"

umount /mnt

6.4.5 Running hello solely on the platform

Run the running script copied in "6.4.4" and configure the environment parameters. The executable file "hello" runs automatically:

#run_hello &

As shown in the following diagram:

Test my first Qt	opia Application 🛛 🕙
	Hello



6.4.6 Running hello in Qtopia on the platform

Find the qtopia script under the directory "/bin/" of platform and execute it. And then "hello" starts to run. #qtopia

6.4.7 Make desktop starter file

Create a text file and add the following contents: (the following contents include: program name, icon name and so on). Change the file name into "xxxx.desktop", and save the file to the directory "\$QPEDIR/apps/Applications/".

Take the application "hello" for example:

[Desktop Entry]

Comment=A Hello Program

Exec=hello

Icon=lyt_demo

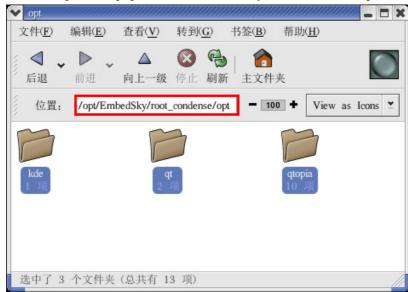
Type=Application

Name=Hello2440

6.4.8 Make file system containing Qtopia

"root_condense" is a simplied file system. Add the relevant files of Qtopia to "root_condense", and then we can get the file system containing Qtopia.

Add 3 directories "kde", "qt" and "qtopia" under the directory "root_condense/opt/"

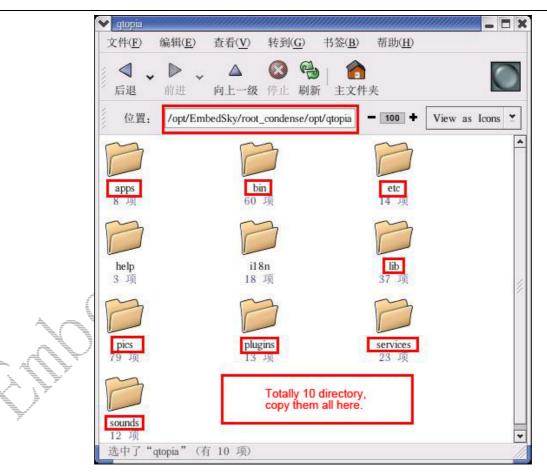


Add Qtopia main application:

Copy directories shown in the following diagram under "/opt/EmbedSky/Qte/touch-qtopia/qtopia/" to "/opt/EmbedSky/root_condense/opt/qtopia/":

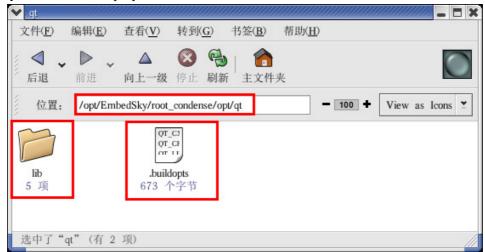






Add libraries needed by Qtopia:

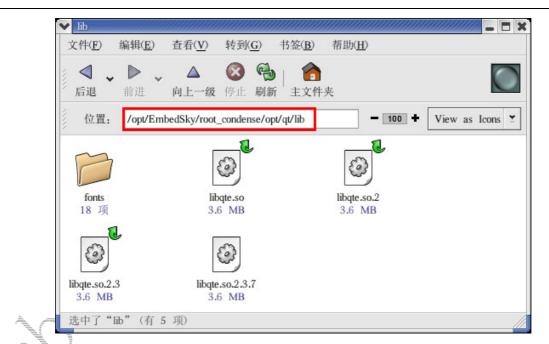
Create the directory "lib" under "/opt/EmbedSky/root_condense/opt/qt/" and copy the file ".buildopts" to "/opt/EmbedSky/root_condense/opt/qt/":



Copy all the files and directories under "/opt/EmbedSky/Qte/touch-qtopia/qt/lib/" to "/opt/EmbedSky/root_condense/opt/qt/lib/":

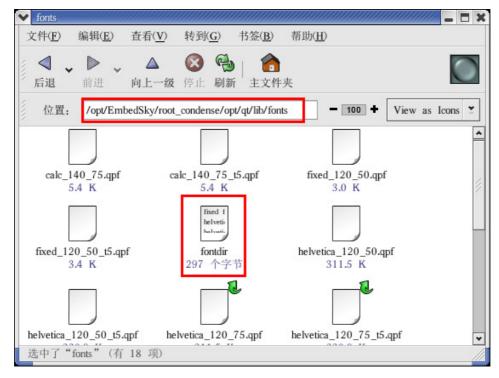






Delete some fonts under "fonts" and modify the file "fontdir" under "file" corresponding with the remaining fonts.

(it is suggested to use the fonts contained in CD-ROM provided by us.)



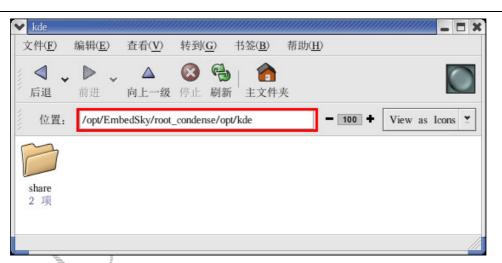
Add Web explorer:

Í

Add the directory "share" under "kde":







Add 2 directories "apps" and "config" under "share":



Create "khtml" directory under "apps" and create "css" directory under "khtml". Copy the file "html4.css" under "/opt/EmbedSky/Qte/touch-qtopia/konq-em/konq-embed/kdesrc/khtml/css/" to "/opt/EmbedSky/root_condense/opt/kde/share/apps/khtml/css/"

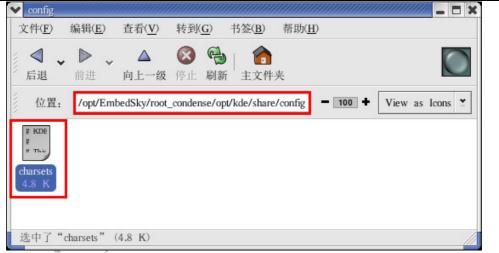
✓ CSS								- 🗆 X
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	转到(<u>G</u>)	书签(<u>B</u>)	帮助(<u>H</u>)			
▲ ~	▶ 、 前进	▲ 向上一级	 停止 刷新 		·夹			\bigcirc
位置:	/opt/Emb	edSky/root_	_condense/op	ot/kde/share	e/apps/khtml/o	css – 100	➡ View as	Icons 😫
html4.css 8.2 K								
选中了"h	tml4.css"	(8.2 K)						

Copy "charsets" under "/opt/EmbedSky/Qte/touch-qtopia/konq-em/konq-embed/kdesrc/kdecore/" to "/opt/EmbedSky/root_condense/opt/kde/share/config":

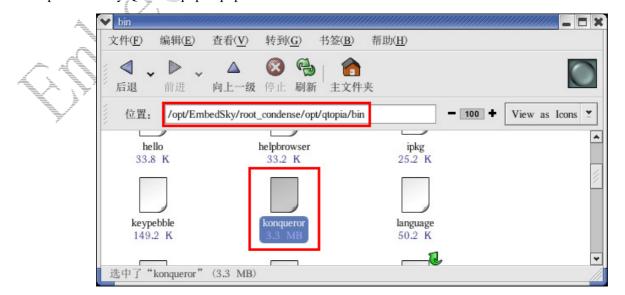


-





Copy "konqueror" under "/opt/EmbedSky/Qte/touch-qtopia/konq-em/konq-embed/src" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/bin/":



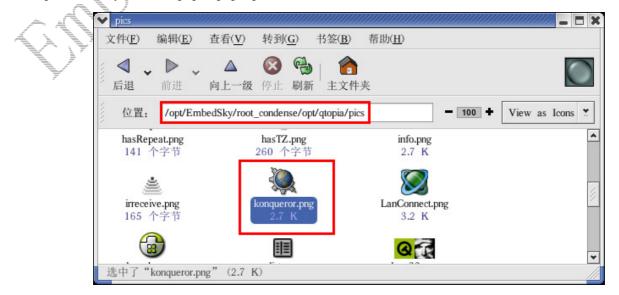
Copy "Web Browser" under "/opt/EmbedSky/Qte/touch-qtopia/konq-em/konq-embed/src" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/apps/Applications/":







Copy "konqueror.png" under "/opt/EmbedSky/Qte/touch-qtopia/konq-em/konq-embed/src" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/pics/":



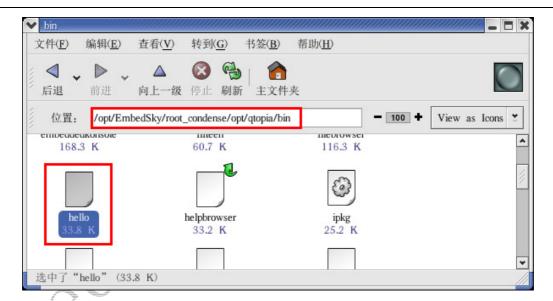
Add the application "Hello":

The executable file "hello" has been copied to "qtopia/bin/" when compiling "Hello" application:



Í





Copy the desktop starter of hello from "/opt/EmbedSky/Qte/touch-qtopia/hello/" to "/opt/EmbedSky/Qte/x86-qtopia/qtopia/apps/Applications/":

	✓ Applications				X
$\langle X \rangle$	文件(<u>F</u>) 编辑(<u>E</u>)	查看(<u>V</u>) 转到(<u>G</u>)	书签(<u>B</u>) 帮助(<u>H</u>)		
	【 ↓ 】 后退 前进	▲ ② ● ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○	合 主文件夹		\bigcirc
	位置: /opt/E	mbedSky/root_condense/op	t/qtopia/apps/Applications	- 100 + View as	s Icons 🞽
	Hello2440	VNC 查看器	Web Browser		•
	世界时间	今天	图片查看器		*
	选中了"Hello244	0"(102 个字节)			

The file system containing Qtopia is now complete. The user could create a directory "Documents" under "/root_condense/root/" and place files like MP3 under this created directory. After platform start-up, the corresponding files could be found under the menu "Documents" in Qt interface.



Í





The user can use the software "mkyaffsimage" to make Yaffs file system by following the method introduced previously, and then burn it to the platform for use.

Caution: The touch screen correction is needed when using touch screen file system containing Qt for the first time. The 5 correction points are respectively at upper left, lower left, upper right, lower right and center of the LCD. After touch screen correction, select simplified Chinese as the supporting language.





Chapter 7 Experiment of driver development

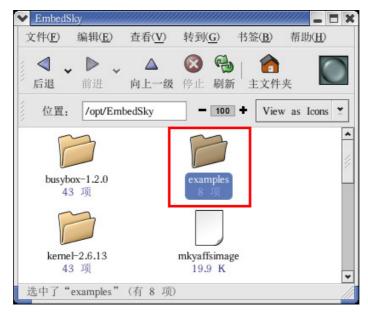
7.1 Application development in Linux

Caution: The cross-compiler of version 3.4.1 is needed for application development under Linux. Make sure you have installed the right version of compiler in PC:

🖌 root@EmbedSky;~	×			
文件(<u>F</u>) 编辑(<u>E</u>) 查看(<u>V</u>) 终端(<u>T</u>) 转到(<u>G</u>) 帮助(<u>H</u>)				
[root@nbedSky root]#_arm-linux-gcc -v	*			
Configured with: /opt/crosstool/crosstool-0.28/build/arm-linux/gcc-3.4.1-glibc-2 .3.3/gcc-3.4.1/configuretarget=arm-linuxhost=i686-host_pc-linux-gnupref ix=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm-linux/gcc-3.4.1-glibc-2.3.3wi th-float=softwith-headers=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm-linux/				
gcc-3.4.1-glibc-2.3.3/arm-linux/includewith-local-prefix=/opt/EmbedSky/crosst ools_3.4.1_softfloat/arm-linux/gcc-3.4.1-glibc-2.3.3/arm-linuxdisable-nlse nable-threads=posixenable-symvers=gnuenablecxa_atexitenable-language s=c,c++enable-sharedenable-c99enable-long-long				
Thread model: posix gcc version 3.4.1 root@EmbedSky root]#	*			

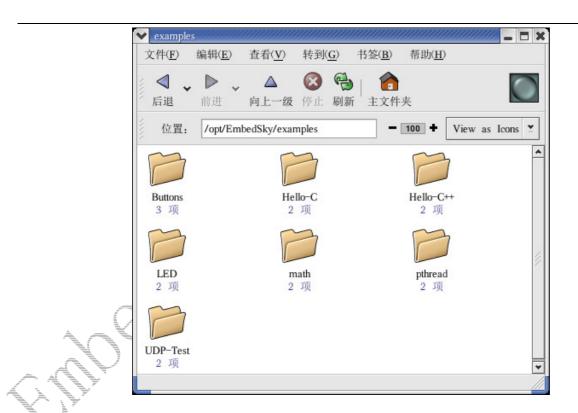
Execute the command "# arm-linux-gcc -v" to check the version of cross-compiler. If the version is not correct, please consult "2.5.1" to re-install the cross-compiler.

The codes mentioned in the following contents can be found in "examples.tar.bz2" of "Linux" in CD-ROM. Locate these files in "/opt/EmbedSky/examples/" after decompressing "examples.tar.bz2":









7. 1. 1 Hello EmbedSky experiment

Hello EmbedSky includes experiments of C language and the ones of C++.

≻ Experiment of C language:

NAME:hello-c.c COPYRIGHT:www.embedsky.net

#include <stdio.h>

}

Í

Execute the command "arm-linux-gcc –o hello-c hello-c.c" or "make" to cross compile the code. And then run the compiled program "hello-c":





● 超级终端 - 超级终端	
て件(2) 編輯(2) 査看(2) 呼叫(2) 传送(2) 帮助(2)	
<pre>channel QPE/Server added channel QPE/IME added Create pluginlibman in libqpe Use QPEApplication's PluginLibraryManager QuickLauncher running Unable to open /usr/share/zoneinfo/zone.tab Timezone data must be installed at /usr/share/zoneinfo/ Unable to open '/usr/share/zoneinfo/America/New_York' TimeZone::data Can't create a valid data object for 'America/New_York' TzCache::location unable to find America/New_York channel QPE/Qplication/quicklauncher added channel QPE/QuickLauncher-797 added Registered QPE/QuickLauncher-797 [root@EmbedSky /]# hello-c ####################################</pre>	

► Experiment of C++:

The following C++ source code is under the directory "/opt/EmbedSky/examples/Hello-C++/hello-c++.c++":

NAME:hello-c++.c++ COPYRIGHT:www.embedsky.net

#include <iostream>
#include <cstring>
using namespace std;

class String

Í

private: char *str; public: String(char *s) //Input character string { int lenght=strlen(s); str = new char[lenght+1]; strcpy(str, s);

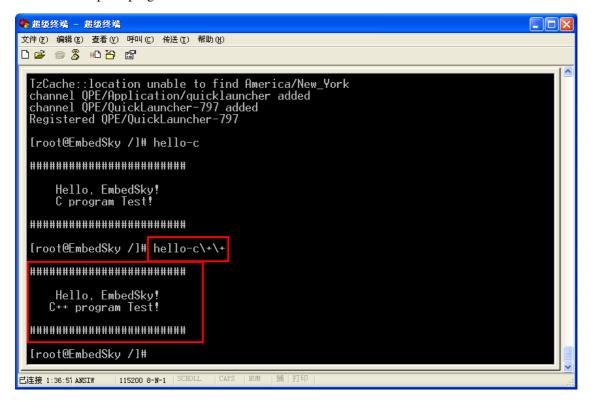


1



void display() //Function of printing information cout << str <<endl; }; int main(void) String s1 = "\n##############################"; //Character string String s2 = " Hello, EmbedSky!"; String s3 = "C++ program Test!"; String s4 = "\n##########################",n"; s1.display();//Call the function of printing s2.display(); s3.display(); s4.display(); return 0; }

Execute the command "arm-linux-g++ –o hello-c++ hello-c++.c++" or "make" to cross compile the code. And then run the compiled program "hello-c":





7.1.2 Experiment of calling math function

The following test code calls the squaring root function.

, i

NAME:mathtest.c	
COPYRIGHT:www.embedsky.net	

#include <stdio.h></stdio.h>	
#include <stdlib.h></stdlib.h>	
#include <math.h></math.h>	
int main(void)	
double a=168.168;	
printf("\n#############################;n");	//Print information
<pre>printf("\nsqrt(%f)=%f\n", a, sqrt(a));</pre>	//Call squaring root function
printf("\n#######################\n\n");	
return 0;	
}	

Execute the command "arm-linux-gcc –o mathtest mathtest.c -lm" (be cautious of the parameter "-lm") or "make" to cross compile the code. And then run the compiled program "mathtest":

🔗 超级终端 - 超级终端	
文件 (2) 编辑 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助 (3)	
<pre>[root@EmbedSky /]# Iroot@EmbedSky /]# mathtest ###################################</pre>	
已连接 2:08:5% ANSIN 115200 8-N-1 SCROLL CAPS NVM 捕 打印	



7.1.3 Experiment of thread programming

The brief introduction of the following thread programming example: A thread reads data from a shared buffer and prints them out; and another thread writes data into the shared buffer at the same time and prints them out. The shared buffer access is based on mutex principle.

```
The source code:
NAME:pthread.c
COPYRIGHT:www.embedsky.net
*****
#include<stddef.h>
#include<stdio.h>
#include<unistd.h>
#include"pthread.h"
void reader_function(void);
void writer_function(void);
char buffer;
int buffer_has_item=0;
pthread_mutex_t mutex;
main()
{
    pthread_t reader;
    pthread_mutex_init(&mutex,NULL);
    pthread_create(&reader,NULL,(void*)&reader_function,NULL);
    writer_function();
void writer_function(void)
    while(1)
        pthread_mutex_lock(&mutex);
        if(buffer_has_item==0)
        {
        buffer='s';
        printf("write test\n");
        buffer_has_item=1;
        ļ
    pthread_mutex_unlock(&mutex);
```

//





}
void reader_function(void)
{
while(1)
{
pthread_mutex_lock(&mutex);
if(buffer_has_item==1)
{
buffer='\0';
printf("read test\n");
buffer_has_item=0;
pthread_mutex_unlock(&mutex)

1

Execute the command "arm-linux-gcc -static -o pthread pthread.c -lpthread" (caution: the parameter "-lpthread" is indispensable. "-lpthread" means calling the library "libpthread") or "make" to cross compile the code. And then run the compiled program "mathtest":

▶ ● 超级终端 - 超级终端	
文件 (2) 編輯 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助 (4)	
write test	
read test	
write test	
read test	
write test	
read test	
write test	
read test	
write test	
read test write test	
read test	
write test	
read test	
write test	
read test	
write test	
read test	
write_test	
read test	
write test	
read test	
write test Troot@EmbedSky /]#	
T OUTELMACTORY 7 Ju	

7.1.4 Experiment of UDP network programming

TCP/IP provides a connectionless transport layer protocol: UDP (User Datagram Protocol). The difference between UDP and TCP/IP is caused by the difference between connectionless socket programming and





connection-oriented socket programming. Every single receiving or sending UDP package contains the address of sender and receiver.

A data package socket of class "SOCK_DGRAM" needs to be created before transporting, by calling the following function:

sockfd = socket(AF_INET,SOCK_DGRAM,0);

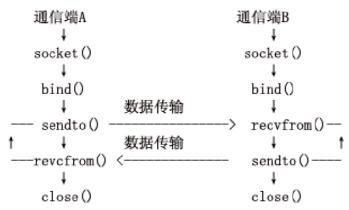
The operation of sending and receiving begins right after the socket creating because of connectionless. The receiver needs to tell sender the receiving port. The 2 functions "sendto" and "recvfrom" are used for sending and receiving. The following contents show how to call these 2 functions:

int sendto (int s, const void *msg, int len, unsigned int flags, const truct sockaddr *to, int tolen);

int recvfrom (int s,void *buf, int len, unsigned int flags, struct sockaddr *from, int formlen);

"s" is the name of socket; "insg" and "buf" point to sending buffer and receiving buffer respectively; "len" represents the length of buffer; "flag" is the option flag. The value is 0 here because it is not used in the example; "to" and "from" point to the destination and source address, including IP address and port information; "tolen" and "fromlen" represent the length of sending and receiving socket address structure. The 2 functions return the length of sending or receiving bytes. The return value is -1 when error occurs.

The general process of connectionless transmission:



In upper diagram, the sender and receiver have bound their address ports. However, in some conditions, one side of the sender or receiver does not have to bind the address port which can be allocated by kernel. During the communication, the side without binding sends the data package preceding the other side, and the receiver extracts port information of the sender from the package in order to get to know the exactly address of the un-bingding side.

The same as function read() and write(), process blocking always occurs in recvfrom() and sendto(). And it is possible to receive an empty package, which is different from TCP/IP. In this case, the application set "msg" of sendto() into "NULL" and set "len" into "0".

NAME:UDP.c COPYRIGHT:www.embedsky.net

#include <sys/types.h>
#include <sys/socket.h>





#include <arpa/inet.h>
#include <stdio.h>

#define BUFLEN 255

int main(int argc, char **argv)

struct sockaddr_in peeraddr, localaddr;

//peeraddr is used to preserve the IP and port address of the peer; localaddr is used to preserve the local

```
socket address.
```

{

int sockfd;

char recmsg[BUFLEN+1];

int socklen, n;

if(argc!=5){

printf("%s <receive IP address> <receive port> <send IP address> <send port>\n", argv[0]);
exit(0);

```
sockfd = socket(AF_INET, SOCK_DGRAM, 0);
```

if(sockfd<0){

printf("socket creating err in udptest\n");

```
exit(1);
```

}

socklen = sizeof(struct sockaddr_in);

memset(&peeraddr, 0, socklen);

peeraddr.sin_family=AF_INET;

peeraddr.sin_port=htons(atoi(argv[2]));

if(inet_pton(AF_INET, argv[1], &peeraddr.sin_addr)<=0){

printf("Wrong receive IP address!\n");

```
exit(0);
```

}

memset(&localaddr, 0, socklen);

localaddr.sin_family=AF_INET;

if(inet_pton(AF_INET, argv[3], &localaddr.sin_addr)<=0){

printf("Wrong send IP address!\n");

exit(0);

exit(2);

}
localaddr.sin_port=htons(atoi(argv[4]));
if(bind(sockfd, &localaddr, socklen)<0){
 printf("bind local address err in udptest!\n");</pre>

336





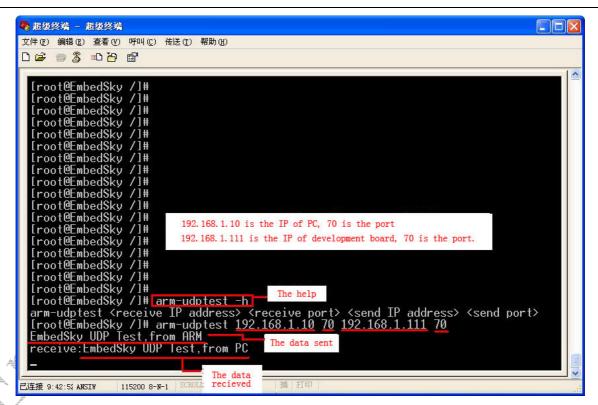
if(fgets(recmsg, BUFLEN, stdin) == NULL) exit(0); if(sendto(sockfd, recmsg, strlen(recmsg), 0, &peeraddr, socklen)<0){ printf("sendto err in udptest!\n"); exit(3); } for(;;){ /*recv&send message loop*/ n = recvfrom(sockfd, recmsg, BUFLEN, 0, &peeraddr, &socklen); if(n<0){ printf("recvfrom err in udptest!\n"); exit(4); }else{ //receiving package successfully recmsg[n]=0; printf("receive:%s", recmsg); if(fgets(recmsg, BUFLEN, stdin) == NULL) exit(0); if(sendto(sockfd, recmsg, strlen(recmsg), 0, &peeraddr, socklen)<0){ printf("send to err in udptest!\n"); exit(3); } }

Compile the test program running in the platform and PC by executing "arm-linux-gcc -o arm-udptest UDP.c" and "gcc –o x86-udptest UDP.c" or by executing the command "make".

Run "arm-updtest" in the platform:







Run "x86-udptest" in PC:

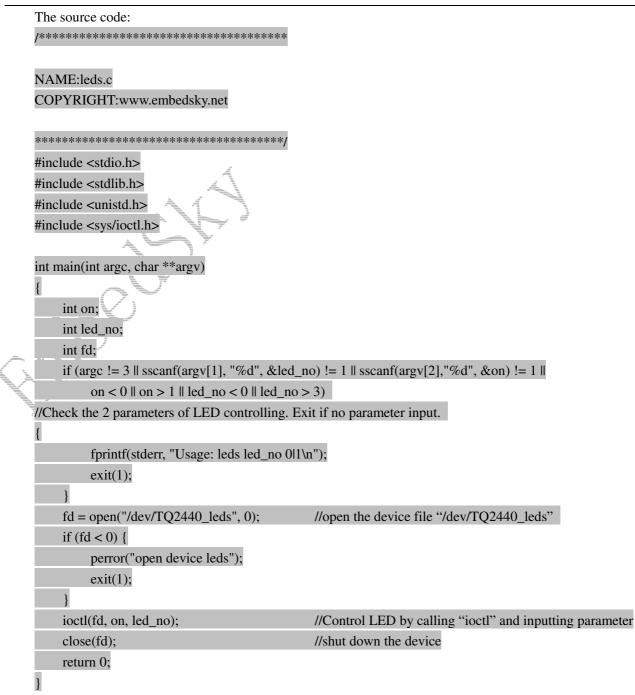
-		
root@HJ 86-udpt root@HJ nbedSky	编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H) J root]# x86-udptest -h The help test <receive address="" ip=""> <receive port=""> <send address="" ip=""> <send port=""> J root]# x86-udptest 192.168.1.111 70 192.168.1.10 70 y UDP Test, from PC The data sent EnbedSky UDP Test, from ARM The data sent The data recieved</send></send></receive></receive>	
	192.168.1.10 is the IP of PC, 70 is the port 192.168.1.111 is the IP of development board, 70 is the port.	

7.1.5 Experiment of controlling LED

LED controlling application manages the start-up, writing and shut-down action of LED driver, in order to control LED swiching on and swithing down.







Cross-compile the test program by executing "arm-linux-gcc -o leds leds.c" or "make", and download the executable program "leds" to platform and execute it.

7.1.6 Experiment of user button controlling

The button controlling program starts the device by using blocking method. The controlling program is blocked into "read" function when no press-action happens. When press-action is detected, "read" function returns, and the program outputs action code by calling "printf" function.

The source code:





NAME:buttons.c COPYRIGHT:www.embedsky.net

#include <sys/stat.h> #include <sys/types.h> #include <fcntl.h> #include <sys/time.h> #include <unistd.h> #include <stdio.h> int main(void) int fd; struct input_event { struct timeval time; unsigned short type; unsigned short code; long value; } Point; fd = open("/dev/input/event1", 0, 0);//Open button device if (fd < 0){ perror("open /dev/input/event1:"); exit(1); } for (;;) { read(fd,&Point,sizeof Point); //Capture button state information printf("Type: %d Code: %d \n", Point.type, Point.code); //print out button information } return 0; }

Cross-compile the test program by executing "arm-linux-gcc -o buttons buttons.c" or "make", and download the executable program "buttons" to platform to execute it.





<pre>mbedSky /]# mbedSky /]# m</pre>	

7.2 Example of driver development in Linux

Caution: Make sure the cross-compiler of version 3.4.1 has been installed.

▼ root@En	nbedSky:~							. 🗆 🗴
文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)			
	edSky roo							^
	specs from 3.3/lib/gc				_3.4.1_soft	float/arm-linu	1x/gcc-3.4	4.1-
-	-				28/bu i ld/a	rm-linux/gcc-3	3.4.1-glił	bc-2
.3.3/gcc-3.4.1/configure target=arm-linux host=i686-host_pc-linux-gnu pref ix=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm-linux/gcc-3.4.1-glibc-2.3.3 wi								
-								
th-float=softwith-headers=/opt/EnbedSky/crosstools_3.4.1_softfloat/arm-linux/ gcc-3.4.1-glibc-2.3.3/arm-linux/includewith-local-prefix=/opt/EnbedSky/crosst								
-						rm-linuxdis		
nable-threads=posixenable-symvers=gnuenablecxa_atexitenable-language s=c,c++enable-sharedenable-c99enable-long-long								
			enable-c9	9enabl	le-long-lon	g		
	del: posi	x						
gcc versi								
[root@Emb	oedSky roo	t]#						-

Execute the command "# arm-linux-gcc -v". If the information above the red underline in the upper program, it indicates that the cross-compiler of version 3.4.1 has been installed; If there is no such information, please consult "2.5.1"

The location of the source code:

- -The source code of experiment 1: /opt/EmbedSky/kernel-2.6.13/drivers/char/TQ2440_hello.c
- -The source code of experiment 2: /opt/EmbedSky/kernel-2.6.13/ drivers/char/TQ2440_leds.c
- -The source code of experiment 3: /opt/EmbedSky/kernel-2.6.13/ drivers/input/keyboard/TQ2440_buttons.c



7.2.1 Hello EmbedSky experiment

We provide an example in the following contents to illustrate the general steps of Linux driver development.

It is suggested to use the macro module_init() and module_exit() to record the initialization function and exit function.

There are 2 methods to load driver in Linux:

- In the process of system start-up, load driver module by code itself.
- After system start-up, load driver module by using the instructions "insmod" and so on.

The driver module is loaded or unloaded by system calling; Or the driver entry is put to a certain place when compiling, and a group of specific codes find the entry and load this driver after kernel start-up. The macro-definition "module init()" and "module exit()" are in the kernel source code file "include/linux/init.h".

The following example program is used for calling print command to load and unload driver.

The source code:

NAME:TQ2440 hello.c COPYRIGHT:www.embedsky.net

****** #include <linux/config.h> #include <linux/module.h> #include <linux/kernel.h> #include <linux/fs.h> #include <linux/init.h> #include <linux/devfs fs kernel.h> #include <linux/miscdevice.h> #include <linux/delay.h> #include <asm/irq.h> #include <asm/arch/regs-gpio.h> #include <asm/hardware.h>

MODULE LICENSE("Dual BSD/GPL");

static int __init TQ2440_hello_init(void) //Driver initialization function { printk("<1>\n Hello,EmbedSky!\n"); printk("<1>\nThis is first driver program.\n\n"); return 0;

static void __exit TQ2440_hello_exit(void)

//Driver exit function

342

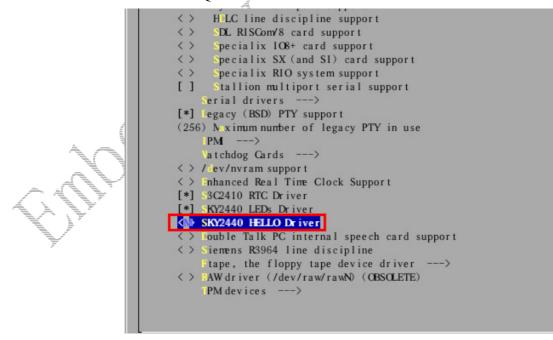




Exit!\n"); printk("<1>\n printk("<1>\nGoodbye EmbedSky!\n\n");

<pre>module_init(TQ2440_hello_init);</pre>	//Driver module initialization macro
module_exit(TQ2440_hello_exit);	//Driver module exit macro

When configuring the kernel, select "M" for "TQ2440_HELLO" and set the path "Device Drivers->Character devices->TQ2440 HELLO Driver":

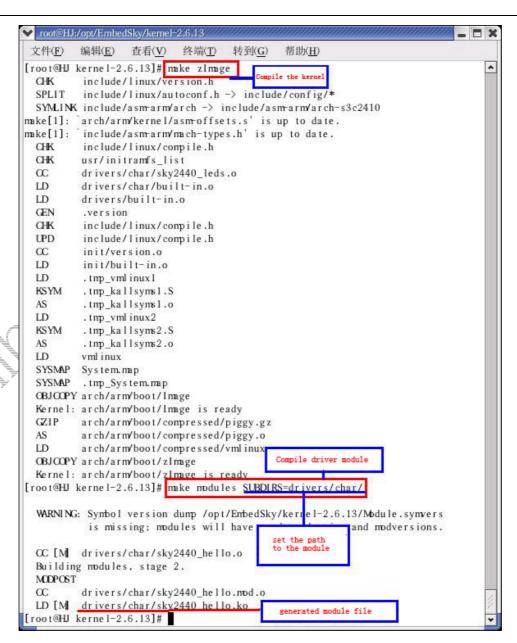


After configuration, compile the kernel first and then compile the module.

Í







Download the compiled file "TQ2440_hello.ko" to platform:





2 超级终端 - 超级终端	
文件 (2) 编辑 (2) 查看 (2) 呼叫 (2) 传送 (1) 帮助 (1)	
Unable to open /usr/share/zoneinfo/zone.tab Timezone data must be installed at /usr/share/zoneinfo/ Unable to open '/usr/share/zoneinfo/America/New_York' TimeZone::data Can't create a valid data object for 'America/New_York' TzCache::location unable to find America/New_York channel QPE/Application/quicklauncher added channel QPE/QuickLauncher-797 added Registered QPE/QuickLauncher-797	
[root@EmbedSky /]# ins insmod install [root@EmbedSky /]# insmod sky2440_hello.ko — <mark>load module</mark> Halla EmbedSky!	
Hello,EmbedSky! This is first driver program.	
[root@EmbedSky /]# rmmod sky2440_hello.ko Exit! Goodbye EmbedSky!	
[root@EmbedSky /]# _ 已连接 11:04:《ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印	•

7.2.2 Experiment of LED driver

The user LED driver controls the LED by controlling the 4 I/O ports of MCU. The following contents introduce the structure of character device driver program.

NAME:TQ2440_leds.c COPYRIGHT:www.embedsky.net

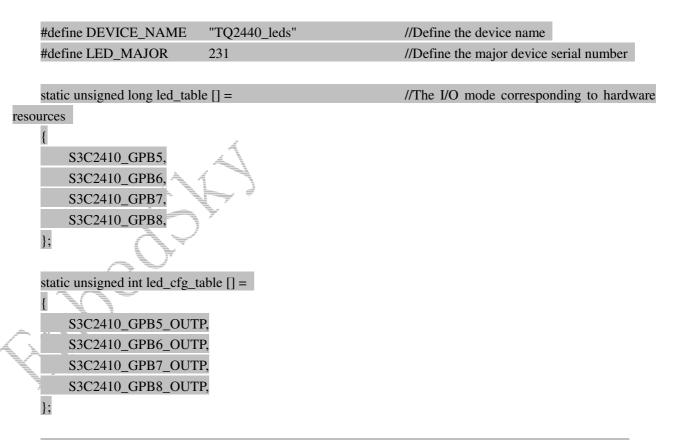
_

#include <linux/config.h>
#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/fs.h>
#include <linux/init.h>
#include <linux/devfs_fs_kernel.h>
#include <linux/miscdevice.h>
#include <linux/delay.h>
#include <asm/irq.h>
#include <asm/arch/regs-gpio.h>

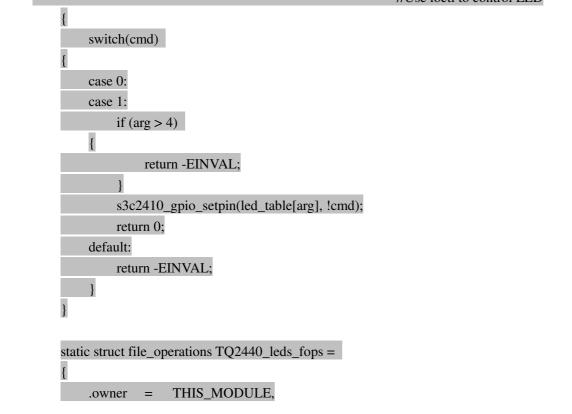




#include <asm/hardware.h>

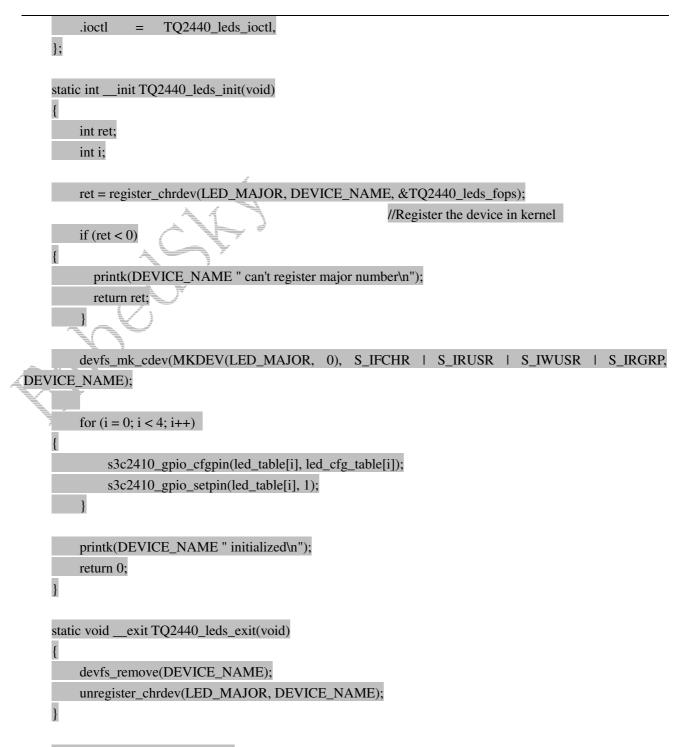


static int TQ2440_leds_ioctl(struct inode *inode, struct file *file, unsigned int cmd, unsigned long arg) //Use ioctl to control LED









module_init(TQ2440_leds_init);

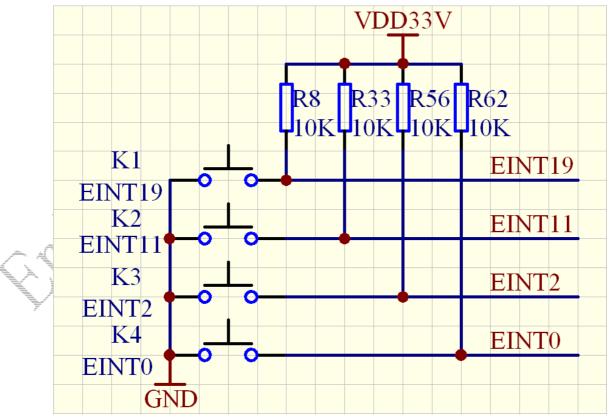
module_exit(TQ2440_leds_exit);

Select "Y" in the option "TQ2440_LED" and set the path "Device Drivers->Character devices->TQ2440 LED Driver". The process of compiling is the same with compiling "TQ2440_HELLO" introduced previously. The LED driver has been included in kernel image in CD-ROM, and the user could use the LED controlling application compiled previously to operate the driver.



7.2.3 Experiment of user keyboard driver

The control mode of keyboard on platform is interrupt. The 4 user button are corresponding to the interrupt sources 0, 2, 11 and 19 (please consult the schematic diagram in CD-ROOM).



The steps of using the external interrupt:

- Register interrupt function, and call it when interrupt occurs.
- Capture the voltage level of pin when interrupt occurs.
- Clear the interrupt state.

The source code:

Í

NAME:TQ2440_buttons.c COPYRIGHT:www.embedsky.net

#include <linux/config.h> #include <linux/errno.h> #include <linux/kernel.h> #include <linux/module.h> #include <linux/slab.h> #include <linux/slab.h>





#include <linux/init.h>
#include <linux/delay.h>
#include <linux/interrupt.h>
#include <linux/device.h>
#include <asm/io.h>
#include <asm/irq.h>
#include <asm/arch/regs-gpio.h>

#define TQ2440_BUTVERSION 0x0001	
----------------------------------	--

#define DEVICE_NAME "TQ2440-buttons"//Device name

#define REPEAT_DELAY HZ/10

#ifdef DEBUG

#define dprintk(msg...) printk(KERN_DEBUG "TQ2440_buttons: " msg);
#else

#define dprintk(msg...)

#endif

-

MODULE_AUTHOR("Yellow <think3133@yahoo.com.cn>"); MODULE_DESCRIPTION("TQ2440 buttons Driver"); MODULE_LICENSE("GPL");

struct TQ2440_button {
 int irq;
 int pin;
 int pin_setting;
 int keycode;
 char *name;
 int last_state;
 struct timer_list timer;
};

static struct TQ2440_button TQ2440_buttons[] = //Device structure

{

{ IRQ_EINT0, \$3C2410_GPF0, \$3C2410_GPF0_EINT0, KEY_3, "TQ2440_K1", 0 },
{ IRQ_EINT2, \$3C2410_GPF2, \$3C2410_GPF2_EINT2, KEY_2, "TQ2440_K2", 0 },
{ IRQ_EINT11, \$3C2410_GPG3, \$3C2410_GPG3_EINT11, KEY_1, "TQ2440_K3", 0 },
{ IRQ_EINT19, \$3C2410_GPG11, \$3C2410_GPG11_EINT19, KEY_ESC, "TQ2440_K4", 0 },
};

struct TQ2440_buttons_private //Private device data structure





	struct input_d	ev dev;
	spinlock_t	lock;
	int	count;
	int	shift;
	char	phys[32];
};		

static struct TQ2440_buttons_private priv;

static irqreturn_t TQ2440_buttons_keyevent(int irq, void *dev_id, struct pt_regs *regs) //Keyboard action processing function

{

di la constante da la constant

struct TQ2440_button *button = (struct TQ2440_button *)dev_id;

int down;

if (!button)

return IRQ_HANDLED;

down = !(s3c2410_gpio_getpin(button->pin));

if (button->last_state == down)
return IRQ_HANDLED;

button->last_state = down;

dprintk("%s button %s\n",button->name, down ? "pressed" : "released");

input_report_key(&priv.dev, button->keycode, down); input_sync(&priv.dev);

if (down)

mod_timer(&button->timer, jiffies + REPEAT_DELAY);

return IRQ_HANDLED;

static void TQ2440_buttons_timer_callback(unsigned long data)

struct TQ2440_button *button = (struct TQ2440_button *) data; int down;

down = !(s3c2410_gpio_getpin(button->pin));





if (down) {
 dprintk("Timer: %s button %s\n",button->name, down ? "pressed" : "released");
 input_report_key(&priv.dev, button->keycode, down);
 input_sync(&priv.dev);
 mod_timer(&button->timer, jiffies + REPEAT_DELAY);
}

static int __init TQ2440_buttons_probe(struct device *dev)//Keyboard loading function

int i;

memset(&priv, 0, sizeof(struct TQ2440_buttons_private)); init_input_dev(&priv.dev); priv.dev.evbit[0] = BIT(EV_KEY); sprintf(priv.phys, "input/TQ2440_buttons0");

priv.dev.private = &priv; priv.dev.name = DEVICE_NAME; priv.dev.phys = priv.phys; priv.dev.id.bustype = BUS_HOST; priv.dev.id.vendor = 0xDEAD; priv.dev.id.product = 0xBEEF; priv.dev.id.version = TQ2440_BUTVERSION;

for (i = 0; i < ARRAY_SIZE (TQ2440_buttons); i++) {
 set_bit(TQ2440_buttons[i].keycode, priv.dev.keybit);
 s3c2410_gpio_cfgpin(TQ2440_buttons[i].pin,TQ2440_buttons[i].pin_setting);
 request_irq (TQ2440_buttons[i].irq, TQ2440_buttons_keyevent,\
 SA_SAMPLE_RANDOM, TQ2440_buttons[i].name, &TQ2440_buttons[i]);
 set_irq_type(TQ2440_buttons[i].irq, IRQT_BOTHEDGE);

init_timer(&TQ2440_buttons[i].timer); TQ2440_buttons[i].timer.function = TQ2440_buttons_timer_callback; TQ2440_buttons[i].timer.data = (unsigned long)&TQ2440_buttons[i];

printk(KERN_INFO "%s successfully loaded\n", DEVICE_NAME);

input_register_device(&priv.dev);

return 0;

1





static int TQ2440_buttons_remove(struct device *dev)//Keyboard removing function
{

int i;

for (i = 0; i < ARRAY_SIZE (TQ2440_buttons); i++) {	
disable_irq(TQ2440_buttons[i].irq);	
free_irq(TQ2440_buttons[i].irq,&priv.dev);	
input_unregister_device(&priv.dev);	
return 0;	

static struct device_driver TQ2440_buttons_driver = //Driver format structure

.name	= DEVICE_NAME,
.bus	= &platform_bus_type,
.probe	= TQ2440_buttons_probe,
.remove	= TQ2440_buttons_remove,

int __init TQ2440_buttons_init(void)
{
 return driver_register(&TQ2440_buttons_driver);/Load the driver
}

void __exit TQ2440_buttons_exit(void)
{
 driver_unregister(&TQ2440_buttons_driver);

module_init(TQ2440_buttons_init);

}

module_exit(TQ2440_buttons_exit);

The kernel configuration list and compiled kernel in CD-ROM do not contain "TQ2440_buttons" driver, however, it could be modified. The file system "root_nfs" has been added with compiled keyboard module. And the path is "/opt/EmbedSky/root_nfs/lib/sky_buttons.ko".

//Unload the driver

Compiling method: Execute the command "make modules SUBDIRS=drivers/input/keyboard" after kernel compiling, and the module "sky_buttons.ko" is created under the directory "drivers/input/keyboard". And then download this module to file system to support mounting and unmounting:

CateDig



🗞 超级终端 - 超级终端	X
文件 (2) 编辑 (2) 查看 (2) 呼叫 (2) 传送 (2) 帮助 (3)	
D 🚔 📾 🍒 🗈 🎦 🖆	
Setting up QCop to QPE/System channel QPE/Card added channel QPE/Server added channel QPE/IME added Create pluginlibman in libqpe Use QPEApplication's PluginLibraryManager QuickLauncher running Unable to open /usr/share/zoneinfo/Zone.tab Timezone data must be installed at /usr/share/zoneinfo/ Unable to open /usr/share/zoneinfo/America/New_York' TimeZone::data Can't create a valid data object for 'America/New_York' TzCache::location unable to find America/New_York channel QPE/Application/quicklauncher added channel QPE/Application/quicklauncher-797 Iroot@EmbedSky /1# insmod /lib/sky2440_buttons.ko SKY2440-buttons successfully loaded Iroot@EmbedSky /1# rmmod /lib/sky2440_buttons.ko Trying to free free IRQ16 Trying to free free IRQ16 Trying to free free IRQ55 Trying to free free IRQ55 Trying to free free IRQ63 Iroot@EmbedSky /1#	

The processes of loading and unloading are shown in the upper diagram.

3 methods of driver editing are introduced in the upper contents. Please pay more attention to the functions registering driver and uninstalling driver.

7.3 Experiment of Non-OS application

This experiment is based on the integrated development environment ADS 1.2.

7.3.1 Configuring the experiment environment

The configuration steps:

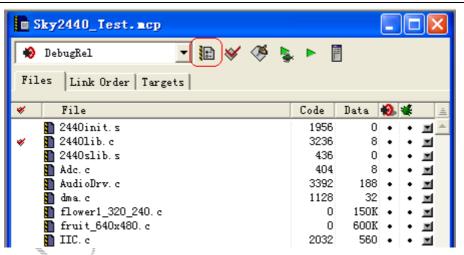
Í

Step1, decompress the test application: enter into the directory and open the project "TQ2440_test.mcp" by using ADS:



Í





Step2, click the icon in red circle in the upper diagram or press "Alt+F7" to enter into the interface "Release settings" (or select "DebugRel" as target to enter into "DebugRes Settings"), or enter in from "Edit": Check the settings in "Language Settings": "Architecture or Processor" is "ARM920T" or not:

DebugRel Settings	?×
Target Settings Panels 🛛 ARM Assembler	
 Target Target Settings Access Paths Build Extras Runtime Settings File Mappings Source Trees ARM Target Language Settings ARM Assembler ARM C Compiler ARM C++ Compiler 	•
Thumb C Compiler - Equivalent Command Line - cpu ARM920T - fpu None	
- Linker	~
ARM fromELF	
Editor	
Pactory Setting: Revert Import Panel Export P	anel
OK Cancel Ag	pply

Check Linker: "ARM Linker"--- "R0 base" is 0x3000000 or not:





Target Settings Panels	ARM Linker	
 → Target → Target Settings → Access Paths → Build Extras → Runtime Settings → File Mappings → Source Trees → ARM Target ⇒ Language Settings → ARM Compiler → ARM C++ Compiler → Thumb C Compiler → Thumb C++ Com ⇒ Linker → ARM fromELF ⇒ Editor 	Output Options Layout Listings Extras Linktype Simple image Partis Simple Ox30000000 Rwpi © Simple Ox30000000 Rwpi Split Scattered Split Split Symbol gditing Equivalent Command Line -info totals -entry 0x30000000 -ro-base 0x30000000 -map - List.txt -first 2440init.o(Init)	Thoose Thoose Thoose
	Pactory Setting: Revert Import Panel Exp	ort Panel
	OK Cancel	Apply

Step3, after the check, click the file with a prefixed mark (this mark is auto-added), and click "make" to start compiling. If there is no modification, skip this step. Caution: select "Target" mode "DebugRel":

🖻 s	ky2440_Test.mcp			[<
•	DebugRel 🗾 🌆 🧭 🦉 🐚	. 🕨 🗄				
Fil	es Link Order Targets	2				
*	File	Code	Data	10	٠	<u>–</u>
	2440init.s	1956	0	٠	٠	٠
*	📓 24401ib.c	3236	8	٠	٠	
	📓 2440slib.s	436	0	٠	٠	
	Adc. c	404	8	٠	٠	
	👔 AudioDrv. c	3392	188	٠	٠	
	J L	1100				

Step4, after compiling, finish the connection of the target board (Jtag, power and serial port). Start up the board and then enter into bootloader download mode; Start up H-JTAG and H-JTAG and auto-detect CPU (if no CPU is detected, please check the configuration of H-JTAG); Click "Debug" after detecting CPU:

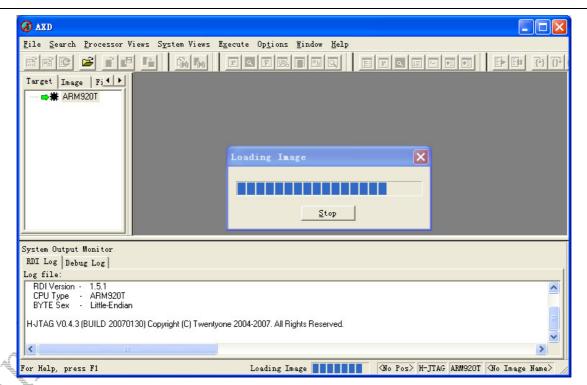
Sky2440_Test.mcp							<
1	DebugRel 🗾 🔝 🖋 🧏 🖕	× 🕨 [
Fil	Debug						
*	File	Code	Data	۰.	*		±.
	🚺 2440init.s	1956	0	٠	٠		-
*	🚺 2440lib.c	3236	8	٠	٠		
	🚺 2440slib.s	436	0	٠	٠		

The process of loading image:

1







Step5, after loading image is finished, enter into the following AXD interface. Select "Options" ---"Configure target" to start configuring:

🐼 AXD - [ARE7IDEI - F:\IQ\Test\naked run\src\2440init.s]								
🖗 File Search Processor Views System Views Execute Options Window Help 🛛 - 🖻								
nini 🕑 🖻 主 🖳 🗛 I	🙀 🖪 🖸 Disassembly Mode 🔸							
Target Image Fi ◀ ▶ 107	Configure <u>I</u> nterface	•						
	[ENTR: Configure <u>T</u> arget							
109	streq Configure Processor)	0x070000ea						
110]	·						
111	<u>S</u> ource Path							
+ 112	b Res							
113] 🗸 Status <u>B</u> ar							
114		efined mode						
115	b Handleı <u>P</u> rofiling	errupt						
116	b HandlerPabort ;handler for PAb	ort						
117	b HandlerDabort ;handler for DAb	ort						

Step6, click "Add" to add "H-JTAG.DLL" ("H-JTAG.DLL" is under the folder "H-JTAG"). Select "ADP, ARMUL" and click "Remove" to remove it. And click "OK" to finish the configuration.

Caution: The configuration is needed only in the first use.

1





C	hoose Tar	get				? 🔀
	Target Envi Target ADP ARMUL H-JTAG	RDI 1 1	5 File C:\PROGRA~1\\Bin\Remot <u>C:\PROGRA~1\\Bin\ARMul</u> D:\\H-JTAG.dll	e_A. dll ate. dll	Version 1.2.0.805 1.2.0.805 V0.4.3	Add Remove Rename Save As Configure
	🕔 target	environ	a target environment from ument to the list. Note tha figured at least once befor	t a target e	nvironment	Help

7.3.2 Experiment of test

Í

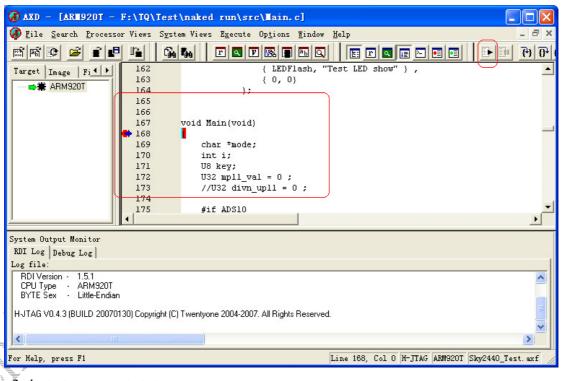
Step1, click the icon "go" in the following red circle or click "go" twice in the menu "Execute". Run the assembler:

🛿 AXD - [ARM920T - F:\TQ\Test\naked run\src\2440init.s]	
🕼 File Search Processor Views System Views Execute Options Window Help	- 8 ×
	6) (b)
Target Image Fi + 109 streq r0,[r0,-r10,ror #1] ;DCD 0x070000ea	-
ABM920T	
111 + 112 b ResetHandler	_
114 b HandlerUndef ;handler for Undefined mode	
115 b HandlerSWI ; handler for SWI interrupt	
116 b HandlerPabort shandler for PAbort	
117 b HandlerDabort ;handler for DAbort	
118 b . reserved	
119 b HandlerIRQ ;handler for IRQ interrupt	
120 b HandlerFIQ ;handler for FIQ interrupt	
121	
122 • Rhv2n	
System Output Monitor RDI Log Debug Log Log file: RDI Version - 1.5.1 CDU Version - 4.5.1	~
CPU Type - ARM920T BYTE Sex - Little-Endian	
H-JTAG V0.4.3 (BUILD 20070130) Copyright (C) Twentyone 2004-2007. All Rights Reserved.	
	~
	>
or Help, press F1 Line 112, Col O H-JTAG ARM920T Sky2440_Test	. axf

Click "go" to run the main program:







Step2, the test program starts to run:

Í

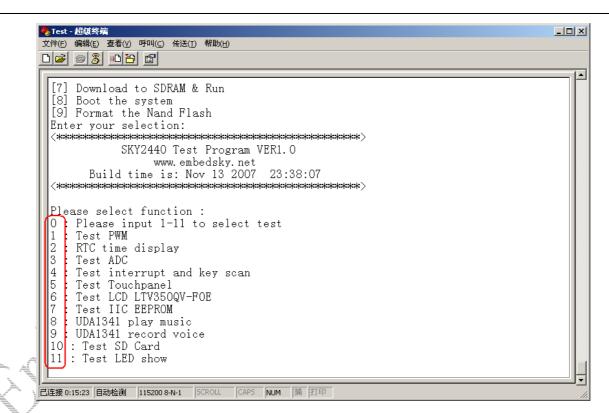
🖉 AXD - [ARM920T - F:\TQ\Test	\naked run\src\Lain. c]
🐼 File Search Processor Views Syst	em Views Execute Options Mindow Melp 🛛 📃 🚽 🗙
ri 🗈 🖆 🛄 😘	
Target Image Fi + 162	{ LEDFlash, "Test LED show" } ,
163	{ 0, 0}
164 165	};
165	
167	void Main(void)
🖶 168	
169	char *mode;
170	int i;
171	U8 key;
172	U32 mpl1_val = 0 ; //U32 divn upl1 = 0 ;
173	//US2 divin_upii = 0 ;
175	#if ADS10
)
System Output Monitor	
RDI Log Debug Log	
Log file:	
BDI Version - 1.5.1	
CPU Type - ARM920T	
BYTE Sex - Little-Endian	
H-JTAG V0.4.3 (BUILD 20070130) Copyright (C)	Twentuone 2004-2007 All Bights Beserved
The final former (beings 20010100) copyright (c)	
<	
For Help, press F1	Running Image

Step3, switch to hyper terminal, the main menu appears, as shown in the following diagram.

Select "0", the main menu appears again; If the LCD has already been connected, the six color stripes "blue, green, red, yellow, pink and viridity" and "米" shape black line appear on LCD.







Step4, select "1", the PWM display starts to run, press "+/-" (caution: Use the keypad, because the key "shift + " is not supported.) to increase and decrease the frequency; Press "ESC" to exit and go back to the main menu:

※ Test - 超级终端 文件(E) 编辑(E) 查看(业) 呼叫(C) 传送(I) 帮助(出)	<u> </u>
3 : Test ADC 4 : Test interrupt and key scan 5 : Test Touchpanel 6 : Test LCD LTV350QV-FOE 7 : Test IIC EEPROM 8 : UDA1341 play music 9 : UDA1341 record voice 10 : Test SD Card 11 : Test LED show 1 BUZZER TEST (PWM Control) Press +/- to increase/reduce the frequency of BUZZER ! Press 'ESC' key to Exit this program ! Freq = 810 Freq = 820 Freq = 820 Freq = 810 Freq = 810 Freq = 800 Freq = 790 Freq = 770	
已连接 0:17:26 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	11.

Step5, select "2" to run RTC test. It appears that the timer keeps increasing; Press "ESC" to exit and go back to the main menu:



1



文件(E)编辑(E)查看(Y)呼叫(C)传送(I)帮助(B) □ [2] ◎ [3] □ [2] [2]	
Freq = 780	
Freq = 770	
Freq = 770	
Please select function :	
0 : Please input 1-11 to select test	
1 : Test PWM	
2 : RTC time display 3 : Test ADC	
4 : Test interrupt and key scan	
5 : Test Touchpanel	
6 : Test LCD LTV350QV-F0E	
7 : Test IIC EEPROM	
8 : UDA1341 play music	
9 : UDA1341 record voice	
10 : Test SD Card	
11 : Test LED show	
2RTC TIME Display, press ESC key to exit !	
RTC time : 2005-06-19 15:21:30	
RTC time : 2005-06-19 15:21:31	
RTC time : 2005-06-19 15:21:32	
RTC time : 2005-06-19 15:21:33	
RTC time : 2005-06-19 15:21:33	

Step6, select "3" to start ADC button test. Adjust the resistance and the displaying data changes correspondingly. Press "ESC" to exit and go back to the main menu: (the following diagram is oriented to adjusting the resistance "ain2")

	文件(E) 編辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	-0×
Ē		
	AIN3: 0200	
	AIN2: 0235 AIN3: 0230	
	AIN2: <u>0234</u> AIN3: 0160	
	AIN2: <u>0238</u> AIN3: 0240	
	AIN2: 0245 AIN3: 0169	
	AIN2: 0268 AIN3: 0354	
	AIN2: 0287 AIN3: 0139	
	AIN2: 0304 AIN3: 0204	
ļ. Ē	L 弦接 0:30:49 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印	

Step7, select "4" to start button test. The relevant testing information would appear. Press "ESC" to exit and





go back to the main menu:

文件(E)编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H) □ 😰 📨 💁 🛅	
0 : Please input 1-11 to select test 1 : Test PWM 2 : RTC time display 3 : Test ADC 4 : Test interrupt and key scan 5 : Test Touchpanel 6 : Test LCD LTV350QV-FOE 7 : Test IIC EEPROM 8 : UDA1341 play music 9 : UDA1341 record voice 10 : Test SD Card 11 : Test LED show 4 Key Scan Test, press ESC key to exit ! Interrupt occur Kaj is pressed! Interrupt occur Key is released! Interrupt occur Ki is pressed! Interrupt occur Key is released! Interrupt occur Key is released!	

Step8, select "5" to start the touch screen test (connect the touch screen before the test). Touch the screen with a pen and the following information appears. Press any key to go back to the main menu:

1: Test PWM 2: RTC time display 3: Test ADC 4: Test interrupt and key scan 5: Test Touchpanel 6: Test LCD LTV350QV-FOE 7: Test IIC EEPROM 8: UDA1341 play music 9: UDA1341 record voice 10: Test SD Card 11: Test LED show SADC touch screen test Type any key to exit!!! Stylus Down, please count=000 XP=0426, YP=0492 count=001 XP=0429, YP=0294 count=003 XP=0429, YP=0294 count=004 XP=0357, YP=0283 count=004 XP=0357, YP=0203 count=006 XP=0713, YP=0509	

Step9, select "6" to start LCD test. The TFT screen would display the pictures "black \rightarrow white \rightarrow blue \rightarrow green \rightarrow 6 color stripes \rightarrow flower \rightarrow TQ logo". Press any key to go back to the main menu.



À



	1
Please select function : 0 : Please input 1-11 to select test 1 : Test PWM 2 : RTC time display 3 : Test ADC 4 : Test interrupt and key scan 5 : Test Touchpanel 6 : Test LCD LTV350QV-FOE 7 : Test LCD LTV350QV-FOE 7 : Test LCD EEPROM 8 : UDA1341 play music 9 : UDA1341 record voice 10 : Test SD Card 11 : Test LED show 6	
Test LTV350QV-F05 (TFT LCD)! LCD clear screen is finished! press any key to continue! LCD clear screen is finished! press any key to continue! LCD clear screen is finished! press any key to continue! LCD clear screen is finished! press any key to continue! LCD clear screen is finished! press any key to continue! LCD Test Complete! bmp, Any Key To Next!	

Step10, select "7" to start IIC test. CPU writes some data to IIC and then reads them. LCD has been cleared in selection "6", and the 6 stripes would no longer appear:

Ser	ial	Port	: 03	SB Po	ort	Cond	Eigur	atio	on b	felp							
9 :	: UI	DA1:	341	ree	core	d ve	Dice	2									
10	•	Test	t SI) Ca	ard												
11	1	Test	t LI	ED s	sho	U.											
7																	
IIC	; T(est	(Int	teri	rupt	t) (usi	ng I	AT24	4C 0:	2						
Wri	te	tes	st (data	a in	nto	AT:	240	02								
Rea	nd 1	test	t da	ata	fre	om í	AT2	4C 0:	2								
0	1	2	3	4	5	6	7	8	9	а	b	C	d	е	F		
10	11	12	13	14	15	16	17	18	19	1a	1b	1c	1d	1e	1f		
20	21	22	23	24	25	26	27	28	29	2a	2b	2c	2d	2e	2f		
30	31	32	33	34	35	36	37	38	39	3a	3b	3c	3d	3e	3f		
40	41	42	43	44	45	46	47	48	49	4a	4b	4c	4d	4e	4f		
50	51	52	53	54	55	56	57	58	59	5a	5b	5c	5d	5e	5f		
60	61	62	63	64	65	66	67	68	69	6a	6b	6C	6d	6e	6f	Writing data in	
70	71	72	73	74	75	76	77	78	79	7a	7b	7c	7d	7e	7f	AT24C02 and read	
80	81	82	83	84	85	86	87	88	89	8a	8b	8c	8d	8e	8f		
90	91	92	93	94	95	96	97	98	99	9a	9b	9c	9d	9e	9f	Land Development	
aØ													ad		000000		
bØ	b1	b2	b3	b4	b5	bó	b7	b 8	b9	ba	bb	bc	bd	be	bf		
C Ø	c1	c2	c3	C4	c5	CÓ	c7	c 8	c9	ca	CD	CC	cd	ce	Cf		
dØ					7752		77833				7050		dd				
e Ø	e1	e2	e3	e4	e5	66	e7	e8	e9	ea	eb	ec	ed	ee	ef		
FØ	f1	f2	f3	F4	f5	F6	f7	f8	f9	fa	fb	fc	fd	fe	ff		

Step11, select "8" to start sound card test. Press "+" to increase the volume; "-" to decrease the volume; "m" to mute; "p" to pause. Press "ESC" to go back to the main menu:





Test - 超级终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(T) 帮助(H)

D 🚅 🍙 🥈 💼 🔁

Press 'ESC' to quit, '+' to inc volume, '-' to dec volume, 'm' to mute, 'p' to p ause Please select function : 0 : Please input 1-11 to select test 1 : Test PWM : RTC time display : Test ADC 2 13 4 Test interrupt and key scan 1 Test Touchpanel 15 : Test LCD LTV350QV-FOE 6 : Test IIC EEPROM : 8 : UDA1341 play music 9 : UDA1341 record voice : Test SD Card : Test LED show 10 11 Sample Rate = 22050, Channels = 2, 16BitsPerSample, size = 243508 err = 0Now playing the file Press 'ESC' to quit, '+' to inc volume, '-' to dec volume, 'm' to mute, 'p' to p ause

- U ×

已连接 0:59:19 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印

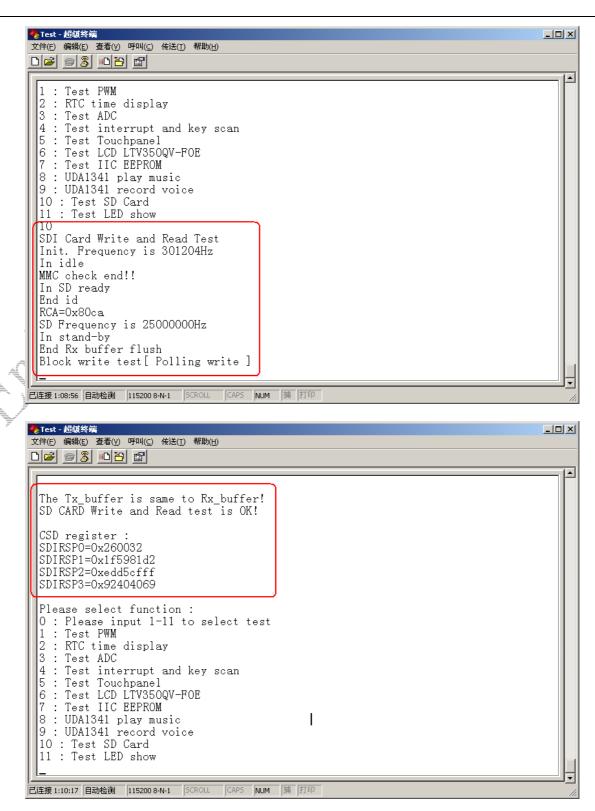
Step12, select "9" to start record test. Press any key to start. Press "ESC" to go back to the main menu:

饮件(E)编辑(E)查看(Y)呼叫(C) 传送(I)帮助(B) □ 🚰 🍘 🔊 ■□ 🛅 📴			
Press'ESC' to quit, '+' to inc volume, '- ause	-' to dec volum	e, 'm' to mute,	'p' to p
Please select function : 0 : Please input 1-11 to select test 1 : Test PWM 2 : RTC time display 3 : Test ADC 4 : Test interrupt and key scan 5 : Test Touchpanel 6 : Test LCD LTV350QV-FOE 7 : Test IIC EEPROM 8 : UDA1341 play music 9 : UDA1341 record voice 10 : Test SD Card 11 : Test LED show			
9 The Frequency of record is 48KHz err = 0			
err - 0 Added 1024 buffer for record <u>Press any to Record</u> Now begin recording, Press 'ESC' to quit			

Step13, select "10" to start SD card test. Insert the SD card before the test begins. After the SD card information has been checked. It auto-goes back to the main menu:







Step14, select "11" to start LED button controlling test. Different buttons are corresponding to different LED display mode. Press "ESC" to go back to the main menu.

Caution: The response of the interrupt is a bit slow, please press button and hold for a while during the test.





冬 Test - 超級终端 文件(E) 编辑(E) 查看(Y) 呼叫(C) 传送(I) 帮助(H)	
key default	
key default	
key default	
TQ LED Testing	
key No. 2	
TQ LED Testing	
key No. 3	
key No. 3	
key No. 3	
□	Ī

Step12, before stop the test, AXD must be stopped first. Click the icon "Stop" or select "Stop" in the menu "Execute":

After the upper operation can user stop the AXD, terminal and the target board:

🐼 AXD - [ARE920T -	F:\IQ\Test	\naked run\src\Tain.c]	
<u>File</u> Search Process	or Views S <u>y</u> ste	m Views Execute Op <u>t</u> ions <u>W</u> indow <u>H</u> elp	- 8 ×
r 🗗 🖆 🖬 🗗			P (9
Target Image Fi → # ARM920T	166 167 168 169 170 171 172 173	<pre>//{ Test_Iic, "Test IIC EEPROM" } ,// lci maskedst { PlayMusicTest, "UDAl341 play music" } , { RecordTest, "UDAl341 record voice" } , { Test_SDI, "Test SD Card" } , { 0, 0} }; void Main(void) char *mode; int i; US key; U32 mpl1_val = 0 ; //U32 divn_upl1 = 0 ; #if ADS10 // _rt_lib_init(); //for ADS 1.0 #endif</pre>	
System Output Monitor	<u> </u>		
RDI Log Debug Log			
Log file:			
ABM BDI 1 51 S ASYNC BDI	L Protocol Convert	er ADS v1 2 (Ruild number 805). Conuriati (a) ARM Limited 2001	
Stop the program		Running Image	st.axf //



7.3.3 Burning TQ2440-Test.bin file

Please consult "2.4 \ddagger " to burn the program into Nor Flash, or consult the "Step7" in "2.6.3 \ddagger " to burn the program into SDRAM, and then jump to the relevant address to begin to run.

Υψ.

_





Chapter 8 Transplantation of Web server

The Web server used in SKY2410 is made up of "boa" and "cgic". The building steps:

8.1 Transplanting boa software

8.1.1 Building compiling environment

The official web site of "boa" is www.boa.org. The download address is:

https://sourceforge.net/project/showfiles.php?group_id=78. The latest version is: boa-0.94.13

After download, decompress the file to the directory "/opt/EmbedSky/", and the directory "boa-0.94.13" is auto-created:

#tar xyfz boa-0.94.13.tar.gz -C /opt/EmbedSky/

8. 1. 2 Configuring the compiling condition

Configure "boa":

#cd /opt/EmbedSky/boa-0.94.13/src

#./configure

Instruction: The prompt of configuration script about how to configure is not correct. So we use this method. The Makefile is auto-created under "boa-0.94.13/src". Modify the Makefile:

#vi Makefile

In line 31 and line 32, find "CC = $gcc \pi CPP = gcc -E$ ". Modify this sentence into "CC = arm-linux-gcc $\pi CPP = arm-linux-g++ -E$ ". Save the modification and exit.

Modify the file "boa.c":

```
#vi boa.c
```

Find the following contents between line 225 and line 227, and annotate it:

```
if (setuid(0) != -1) {
```

```
DIE("icky Linux kernel bug!");
```

```
}
Save and exit.
```

8.1.3 Compiling and optimizing

After compiling , a executable file "boa" (about 232K) is auto-created under the directory "boa-0.94.13": #make

Optimize the file:





#arm-linux-strip boa

The process of optimizing removes the debug information from boa, and the size changes from 232K to 62K. The transplantation of is now complete.

8.2 Transplanting cgic library

8.2.1 Building compile environment

The download web site of cgic library is: <u>http://www.boutell.com/cgic/cgic205.tar.gz</u>, and the latest version is cgic205 version

Download the file and decompress it to the directory "/opt/EmbedSky/", and the directory "cgic205" is auto-created:

#tar xvfz cgic205.tar.gz -C /opt/EmbedSky/

8.2.2 Configuring compile condition

Modify the file "Makefile" in the directory "cgic205": #cd /opt/EmbedSky/cgic205 #vi Makefile The contents after modification: CFLAGS=-g -Wall CC=arm-linux-gcc AR=arm-linux-ar RANLIB=arm-linux-ranlib LIBS=-L./ -lcgic

//Used to be CC = gcc
//Used to be AR = ar
//Used to be RANLIB = ranlib

all: libcgic.a cgictest.cgi capture

install: libcgic.a cp libcgic.a /usr/local/lib cp cgic.h /usr/local/include @echo libcgic.a is in /usr/local/lib. cgic.h is in /usr/local/include.

libcgic.a: cgic.o cgic.h rm -f libcgic.a \$(AR) rc libcgic.a cgic.o \$(RANLIB) libcgic.a

#mingw32 and cygwin users: replace .cgi with .exe

cgictest.cgi: cgictest.o libcgic.a





//Change "gcc" into: \$(CC) \$(CFLAGS)

capture: capture.o libcgic.a \$(CC) \$(CFLAGS) capture.o -o capture \${LIBS}

\$(CC) \$(CFLAGS) cgictest.o -o cgictest.cgi \${LIBS}

//Change "gcc" into: \$(CC) \$(CFLAGS)

clean: rm -f *.o *.a cgictest.cgi capture 修改后保存退出。

8.2.3 Compiling and optimizing

After compiling , the executable file "capture" and test file "cgictest.cgi" is auto-created:

#make

Optimizing:

#arm-linux-strip capture

Decrease the size of "capture" from 100K to 29K.

8.3 Configuring Web server

After the former transplantation works, now begin to configure Web server. (take NFS for example here)

8.3.1 Configuring boa

Create a directory named "web/", and create a directory named "boa/" under "etc" in file system: #cd /opt/EmbedSky/root_nfs

#mkdir web etc/boa

Copy the transplanted file "boa" to the directory "sbin/" in file system:

#cp /opt/TQ/boa-0.94.13/src/boa /opt/TQ/root_nfs/sbin

Copy the boa configuration file "boa.conf" from the directory "boa-0.94.13" to "etc/boa/" in file system: #cp /opt/EmbedSky/boa-0.94.13/boa.conf /opt/EmbedSky/root_nfs/etc/boa

Modify the file "boa.conf". (we give the contents modified and the approximate position here)

#cd /opt/EmbedSky/root_nfs/etc/boa

#vi boa.conf

The modified contents:

Port 80

//Line 25

//Listening port number, 80 is the default value.

#Listen 192.168.1.6

//Line 43

//The IP address used for binding, always be annotated. Means it is bound to INADDR_ANY, and it suit to any IP address of server.





//Line 48 User root //Line 49 Group root //Possess the authority of its group, always root. And the group exists in "/etc/group". #ServerAdmin root@localhost //Line 55 //The email address to which the server error alarm is sent to. It is annotated. //Line 62 ErrorLog /dev/console //Error log file. If it does not start from "/xxx", it starts from root path of the server. If the error log is not needed, modify the line into "ErroLog /dev/console" here. The boa printed information appears after system start-up is derived from "/dev/console". //Line 75 AccessLog /dev/null //Access log file. If it does not start from "/xxx", it starts from root path of the server. If this log is not needed, modify the line into "AccessLog /dev/null" or annotate the line. #UseLocaltime //Line 84 //Use local time or not. Use local time if this line is not annotated, otherwise use UTC time. #VerboseCGILogs //Line 90 Record CGI running information or not. Record if the line is not annotated, otherwise do not recored. ServerName yellow //Line 95 //Server name. #VirtualHost //Line 107 //Enable the virtual host or not. The device has many network interfaces, each interface is corresponding to a virtual Web server. This line is usually annotated. DocumentRoot /web //Line 112 //The main directory preserving HTML file. If it does not start from "/xxx", it starts from root path of server. #UserDir public_html //Line 117 //When receiving a customer request, a new directory is added to the main directory. //Line 124 DirectoryIndex index.html //The file name of HTML directory index. If no accessing directory is designated, this directory will be returned.

#DirectoryMaker /usr/lib/boa/boa_indexer //Line 131

//If there is no index file in HTML directory, and the user has not designated the access directory, "boa" would call the function to create an index file and return it to user. But this process is relatively a time consuming work. The user could also annotate this line and add index files to each HTML directory.

DirectoryCache /var/spool/boa/dircache //Line 140





//If "DirectoryIndex" does not exist, and "DirectoryMaker" has been annotated, the user needs to use the function carried by "boa" to create new directory index file in HTML directory (this directory needs to be readable and writable).

KeepAliveMax 1000 //Line 145 //The maximum request number of a connection supported by HTTP sustained action, Annotating this line or setting it to "0" will shut down the HTTP sustained action.

KeepAliveTimeout 10 //Line 149 //The waiting time interval between two requests of server in HTTP sustained action. The dimension is second. The connection is down if time is out.

MimeTypes /etc/mime.types //Line 156 //Designate the location of the file "mime.types". If it dose not start from "/", it starts from the root path of server. The user could choose to annotate this line and use "AddType" to designate clearly in the local file.

Default Type text/plain//Line 161//If the file has no extension name or the extension name is unkown, the default file type "MIME" is used.CGIPath /bin:/usr/bin:/usr/sbin:/sbin//Line 165//Provide the "PATH" environment parameters of CGI program.

#AddType application/x-httpd-cgi cgi //Line 174 //Associate the file extension name and the type "MIME", the same as the funtion of "mime.types" file. The user could annotate this line if use "mime.types" file. Otherwise this line should not be annotated.

#Alias /doc /usr/doc //Line 189 //Designate the redirection path of the document.

ScriptAlias /cgi-bin/ //web/cgi-bin/ //Line 194

//Designate the actual path corresponding to the virtual path of CGI script. The CGI script is always placed in actual path, and the user input "site + virtual path + CGI script" to access. In this line, "/cgi-bin/" is the virtual path and "/web/cgi-bin/" is the actual path.

Save and exit the file.

Copy the file "mime.types" to the directory "etc/". This file can always be found under "/etc" in PC. #cp /etc/mime.types /opt/EmbedSky/root_nfs/etc

8.3.2 Configuring cgic library

Create the sub directory "cgi-bin/" under "web/" in file system: #cd /opt/EmbedSky/root_nfs/web #mkdir cgi-bin Copy the transplanted cgic library and cgic test file to the directory "web/cgi-bin/": #cp /opt/EmbedSky/cgic205/capture /opt/EmbedSky/root_nfs/www/cgi-bin/ #cp /opt/EmbedSky/cgictest.cgi /opt/EmbedSky/root_nfs/www/cgi-bin/





8.4 Testing

After finishing the previous operation, the user could boot the platform by NFS (the default IP address of platform is 192.168.1.6). After the platform start-up, run boa and start web server test.

8.4.1 Static web page test

We provide a testing web page in the file system. Input the following contents in web page explorer in PC. http://192.168.1.6

The web page is open.

8.4.2 CGI script test

Use "helloweb.c" or "cgictest.cgi" for testing.

When using "cgictest.cgi", enter the following contents in web page explorer in PC: http://192.168.1.6/cgi-bin/cgictest.cgi

The web page is open.

Use "helloweb.c" for testing. The source code (under the directory "cgic205/"): #include <stdio.h> main()

{

```
printf("Content-type: text/html\n\n");
printf("<html>\n");
printf("<head><title>CGI Output</title></head>\n");
printf("<body>\n");
printf("<hl>Hello, Web Server.</hl>\n");
printf("<body>\n");
printf("</html>\n");
exit(0);
```

}

Compiling:

#arm-linux-gcc -o helloweb.cgi helloweb.c

#cp helloweb.cgi /opt/TQ/root_nfs/www/cgi-bin

Enter the following content in web page explorer in PC:

http://192.168.1.6/cgi-bin/helloweb.cgi

The web page is open





8.5 Solving the error

8.5.1 Error 1

After "boa" starts up, the error "boa.c:266.icky Linux kernel bug! :No such file or directory" occurs. The solution: When configuring "boa.conf", we set "User" and "Group" into "root". Therefore, we need to annotate the contents from line 225 to line 227 in "boa.c"; Or we could choose to set "User" and "Group" into "nobody", which needs the support of the file "etc/group" in file system. We select the first method.

8.6 Some source codes

8. 6. 1 cgictest cpp source code

//All variables need to be move to "MAIN" function when compiling.

// CGITEST.cpp : Defines the entry point for the console application.

//This program is used to test the upload character string information in WEB server.

#include "stdafx.h"

_

//This line is annotated in LINUX.

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
/*

void main() //The return value in LINUX is integer. Add "return 0" or "return 1" to the corresponding place.

if (getenv("CONTENT-LENGTH"))
{
 char *s = getenv("CONTENT-LENGTH");
 printf(s);
 }
 printf("Contenttype:text/html\n\n");
 printf("Contenttype:text/html\n\n");
 printf("<html>\n");
 printf("<html>\n");
 printf("<head><title>这是测试 POST 方法</title></head>\n");
 printf("<h2> 这是测试 POST 方法</h2>\n");
//printf(s);





printf("<hr>\n"); printf("<a>Go back to out put.html page \n"); printf("</body>\n"); printf("</html>\n"); fflush(stdout); } */ /* convert hex string to int */ //Used for converting the format of Chinese character coding. int htoi(char *s) char *digits="0123456789ABCDEF"; if (islower(s[0])) s[0]=toupper(s[0]); if (islower(s[1])) s[1]=toupper(s[1]); return 16 * (strchr(digits, s[0]) - strchr (digits, '0'))+(strchr(digits, s[1])-strchr(digits, '0')); } void main() { } printf ("Contenttype: text/plain\n\n"); printf("<html>\n"); printf("<head><title>这是测试 POST 方法</title></head>\n"); printf("<body bgcolor=#008080 text=#FFFFF>
\n"); // printf("<img border=0 src=http:</pre> //127.0.0.1:8080/winter.gif width=750 height=120>"); printf(""); printf("<hr noshade color=#FF0000>"); printf("<h2> 这是测试 POST 方法</h2>\n"); printf("<hr noshade color=#FF0000>"); //Place the gotten value in "nValue" int i,n; char c; int nSum = 1; //The number of variables char nStr[1000]; //Used for storing the upload character string, 1000 is the maximum number. //Clear the 10 variables. memset(nStr,0,1000); // char nCurrentValue[200]; //The current gotten value. // char nValueName[10][50]; //Variable name // memset(nValueName,0,500); //Clear the name of 10 variables. char nValue[10][100]; //10 varibles is the maximum number, maximum 100 characters for each variable.





memset(nValue,0,1000);	//Clear the 10 variables.
int nIndex = 0 ;	//Current variable index.
int nPosion = 0 ;	//The serial number of the variable currently operated.
int iseq=0;	//The start flag of each variable.
n=0;	
if(getenv("CONTENT_LENGTH") =	== NULL)
{	
return;	//No "CONTENT_LENGTH" environment parameter exists in
web server environment.	
n=atoi(getenv("CONTENT_LENGT	"H")); //Convert the length of character string into integer type value
printf("数据长度%d ",n);	
for (i=0;i <n;i++)< td=""><td></td></n;i++)<>	
$\{ \ (\) \}$	
c=getchar();	//Get a character from standard input.
nStr[i]=c;	
//The following contents are mainly a	bout the URL coding and decoding.
switch (c)	
{	
case '&':	
nSum += 1 ;	//The number of variables.
nIndex += 1;	//The index number of variable.
nPosion = 0;	//Clear the character position.
//c=' ';	
iseq = 0;	//Clear the start flag of variable.
break;	
case '+':	//Space key conversion.
//c=' ';	
if(iseq == 1)	
{	
nValue[nIndex][nPosion] = ' ';	
nPosion ++;	
break;	
case '%':	//No number or letter, for example, Chinese character coding
conversion.	"No number of letter, for example, chinese character country
l abar a ^[2] t	
char s[3];	
s[0] = getchar();	

s[1] = getchar();











printf(nStr);

}

//Display the POST upload character string

printf("<hr noshade color=#0000FF>");

printf("
");

printf("调用该 CGI 程序的网页的 URL: %s",getenv("HTTP_REFERER"));

printf("
");

printf("调用该 CGI 程序的 Web 浏览器的机器名和域名: %s",getenv("REMOTE_HOST")); printf("
"); printf("IP 地址和主机名: %s",getenv("REMOTE_ADDR"));

printf("IP 地址和主机

printf("
");

printf("服务器的 IP 或名字: %s",getenv("SERVER_NAME"));

printf("
");

printf("主机的端口号: %s",getenv("SERVER_PORT"));

printf("
");

printf("服务器软件的名字: %s",getenv("SERVER_SOFTWARE"));

printf("
");

printf("用户和组名: %s",getenv("REMOTE_USER"));

printf("
");

printf("Web 服务器传递数据给 CGI 程序时所采用的方法%s",getenv("REQUEST_METHOD")); printf("
"); printf("发送给服务器的完整 URL 请求: %s",getenv("REQUEST LINE"));

printf("
");

, i

printf("该 CGI 程序的名称: %s",getenv("SCRIPT_NAME"));

printf("
");

printf("QUERY-STRING: %s",getenv("QUERY_STRING"));
printf("
");

printf("<hr noshade color=#00FF00>"); printf("<a>数据上传测试! \n"); printf("</body>\n"); printf("</html>\n"); fflush(stdout);

} void GetOnePostChar()





}

{

8.6.2 Source code of home page

<html>

<head>

<meta http-equiv="Content-Language" content="zh-cn">

<meta http-equiv="Content-Type" content="text/html; charset=gb2312">

<title>CGI 数据上传测试</title>

</head>

<body background="bg.jpg" style="background-attachment: fixed">

<hr noshade color="#FF0000">

<form method="POST" action="cgi/CGITEST.CGI" name="form">

collapse" height="197">

<blink>

姓名: </blink>

<input name="NAME" size="10" value="Yellow" style="float: left">

IP 地址: <input type="text" name="T1" size="20" value="192.168.1.6">

子网掩码: <input type="text" name="T2" size="20" value="255.255.255.0">

网关地址: <input type="text" name="T3" size="20" value="192.168.1.2">

组播地址: <input type="text" name="T4" size="20" value="234.5.6.7">

用户名:

<input type="text" name="T5" size="20" value="administrator">

口令密码: <input type="password" name="T6" size="20" value="123456">





性别: <select size="1" name="SEX" style="border-style: solid; border-width: 1px; padding-left: 4px;</pre> padding-right: 4px; padding-top: 1px; padding-bottom: 1px" > <option selected value="男">男</option> <option>女</option> </select> 年龄: <input name="AGE" size="10" value="38" style="float: left"> <button name="B2" style="width: 60px; height: 40px" type="submit" value="SEND INFORMATION"> 上传数据</button> </form> <hr noshade color="#FF0000">

 </body> </html>



8.7 Environment variable

8.7.1 The environment variable relevant to the server

The CGI version obeyed by the server
The IP address or name of the server
The port number of the host
The name of the server software

8.7.2 The environment variable relevant to the customer

The customer environment is probably unknown to the server, but it is important because it is relevant to the explorer and so on

ACCEPT	List the response type that could be accepted by request.
ACCEPT_ENCODING	List the coding type that could be supported by the customer
ACCEPT_LANGUAGE AUTORIZATION	Indicate the ISO code that could be accepted by customer
AUTORIZATION	Indicate the qualified customers
FORM	List the EMAIL address of customers
IF_MODIFIED_SINGCE	Return value only when adopting "get" request method and file is older than
the designated date	
PRAGMA	Set the probably used proxy
REFFERER	Indicate the URL of the document currently connected to
USER_AGENT	Indicate the software used by the customer

8.7.3 The environment variable relevant to the request

The CGI application should be aware of all information, because requests could not be same. The information relevant to requests contains important elements, like user calling information, request type and the percentage of transmitted information. The following contents introduce detailed the information, expecially the 3 variables.

REQUEST_METHOD QUERY_STRING CONTENT_LENGTH

1

These 3 variables are very important, which represent the process of transmitting data to CGI program. The user could make good use of them, such as, get to know the opponents are calling your function, the use has registered or not, and connect to your CGI program to set path information to be contained in request. And there is no need to guess in which page of the server you are at.

AUTH_TYPE	The verification mode of the server
CONTENT_FILE	The data file containing CGI program
CONTENT_LENGTH POST	The number of bytes sent to standard input (\ensuremath{STDIN}) in request





CONTENT_TYPE PATH_INFO CGI PATH_TRANSLATED PATH_INFO The corresponding absolute path QUERY_STRING REMOTE_ADDR REMOTE_USER REQUEST_LINE REQUEST_METHOD SCRPT_NAME

The type of data that is sent The added path of the program The part after "?" of URL sent to CGI program The IP or host name of the terminal user The group name of user if the user is legal The complete URL request sent to server The data sending method as part of the HTTP request, like "get" The name of running script





Chapter 9 Embeded database transplantation (SQLite)

The basic objectives of database are data storage, search and so on. Besides the function of search, add, delete, the traditional database also provides many advanced features, like trigger, storage process, data backup, information recover and so on. However, in most situations, the frequently used functions are not the advanced ones, but the basic ones. And the traditional database is always too huge in size, especially for the embedded system. Therefore, the small-sized embedded database begins to win more attention.

Embedded database possesses the general features of database. The difference between embedded database and traditional database: Embedded database is directly driven by program, but the tranditional database is driven by engine response; Embedded database is always small in size, which can be conveniently transplanted to handheld devices.

9.1 Introduction of SQLite database

SQLite database is a kind of embedded database which is oriented to a simplified usage. This database avoids the complex features of the traditional enterprise database, and only keeps the basic database functions.

The function and performance of SQLite are both excellent, although it is some kind of a simplified database. SQLite supports the following features:

- Support ACID affair (ACID is the abbreviation of Automic, Consisten, Isolated and Durable)
- > No managerial configuration is needed
- Support SQL92 norm
- > Data are placed in different files. The maximum size of file supported by SQLite reaches 2TB
- > Database could be shared among different devices with different supporting bytes
- ➤ Small sized
- Small system overhead and high searching efficiency
- Easy-used API interface
- Could be bound with many languages, like "Tcl", "Python", "C/C++", "Java", "Ruby", "Lua", "Perl" and "PHP".
- ➢ No external support is needed
- ➢ The code is well annotated
- > Over 95% of the code have been tested
- Open source

SQLite has the advantages of powerful function, simple interface, high speed, small size and so on. And it is especially suitable for embedded system and secondary development. Here we set a chapter to introduce the SQLite transplantation in ARM-Linux and the brief test. Hoping it would be helpful.



9.2 Transplanting SQLite database

9.2.1 Get SQLite source code

Access the web site http://sqlite.org/download.html and get the latest SQLite source code.

We provide the SQLite source code of version 3.5.9, which is namely the file "SQLite.tar.bz2" under the directory "\Linux\linux-2.6.13\" in TQ2440 CD-ROM.

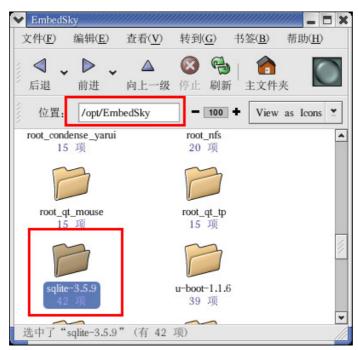
The source code of SQLite contains a simple test program, under the directory "sqlite_test" in source code package.

9.2.2 Transplanting SQLite

Step1: Decompress the source code package. Copy the package to the directory "/opt/EmbedSky" in Linux in PC, and execute the following command to decompress it:

#tar xvfj SQLite.tar.bz2 -C /

After decompression, the directory "sqlite-3.5.9" is auto-created under "/opt/EmbedSky":



Step2, configuring SQLite. Get into the directory "sqlite-3.5.9" and use the compiling script file "build" provided by us to compile SQLite. Executing the following command:

#build

_

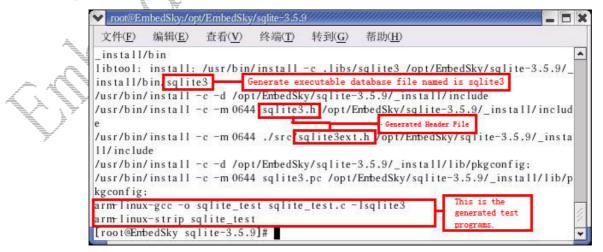
As shown in the following diagram:





文件(<u>F</u>) 练	高辑(<u>E</u>)	查看(<u>V</u>)	终端(<u>T</u>)	转到(<u>G</u>)	帮助(<u>H</u>)			
[root@Embed	Sky Enbe	edSky]#	cd sqlite	-3.5.9/				ŀ
[root@Embed	Sky sql	ite-3.5.9]# ./bui	ld				
					ebuild	type, don't u	usehost.	1
-		-				e mode will		-
checking bu	ild sys	tem type	i686-	pc-linux-	-gnu			- 1
checking ho	st syste	em type.	arm-un	known-lin	nux-gnu			- 1
checking fo					0			- 1
checking fo	r C com	piler de	fault out	put file	name a	.out		
checking wh	ether th	he C com	oiler wor	ks yes	5			
checking wh	ether w	e are cre	oss compi	ling y	ves			- 1
checking fo	r suffix	of exe	cutables.					- 1
checking fo	r suffix	c of obje	ect files	0				- 1
checking wh	ether w	e are us	ing the C	NU C com	oiler y	e s		- 1
checking wh	ether an	m-linux-	gcc acce	pts -g	. yes			. P

Compiling is complete:



Installation automatically starts after compiling. The database program "sqlite3" and the test program "sqlite_test" are under the directory "bin" in "sqlite-3.5.9/_install"; The head file of database is under the directory "include" in "sqlite-3.5.9/_install"; Library file is under the directory "lib" in "sqlite-3.5.9/_install":

×	_install							_ = >
3	文件(<u>F</u>)	编辑(<u>E</u>)	查看(<u>V</u>)	转到(<u>G</u>)	书签(<u>B</u>)	帮助(<u>H</u>)		
1000	⊲ _{后退}	▶ 、 前进	▲ 向上一级	停止 刷	~ _	夹		\bigcirc
1111	位置:	/opt/Emb	edSky/sqlite	e-3.5.9/_ir	istall –	100 +	View as	Icons 🖆
Ī	3		F	3		P		^
T	bin 2 项		inch 2			lib 6 项		



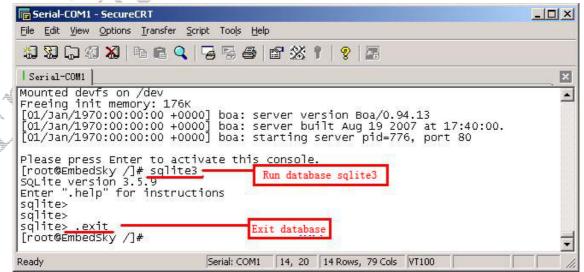
9.2.3 Using SQLite

Copy the files "sqlite3" and "sqlite_test" gained after compiling to the directory "/bin" of file system, and copy the library files under the directory "lib" to "/lib" in file system.

Copy the files under the directory "include" to "/include" of cross-compiler. And copy the files under the directory "lib" to "/lib" of cross-compiler. Therefore, the cross-compiler is able to support "sqlite3" database (the corresponding files has been added to cross-compiler by us).

9.3 Testing SQLite database

Running "sqlite" database solely:



Use "sqlite_test" program to test the database. The operation steps in platform are shown as follows:

[root@EmbedSky /]# sqlite_test test.db "create table

> tbl0(name varchar(10),number smallint);"

[root@EmbedSky /]# sqlite_test test.db "insert into

> tbl0 values('test1',1);"

[root@EmbedSky /]# sqlite_test test.db "insert into

> tbl0 values('test2',2);"

[root@EmbedSky /]# sqlite_test test.db "select *

> from tbl0;"

name = test1

number = 1

name = test2

number = 2

[root@EmbedSky /]#

The operating states:



Í



File Edit View Options	<u>Transfer Script</u> Tools <u>H</u>	lelo		_
	h R Q 356		2	
Serial-COM1				
[[01/Jan/1970:00:0	0:00 +0000] boa: s 0:00 +0000] boa: s	server built A	n Boa/0.94.13 wg 19 2007 at 17:4 er pid=776, port 80	0:00.
Please press Ente [root@EmbedSky /] SQLite version 3. Enter ".help" for sqlite> sqlite> sqlite> .exit	r to activate this # sqlite3 5.9 instructions	; console.		
[root@EmbedSky /]	# sqlite_test test ar(10).number sma	.db "create t llint):"	able Creat a list	in test.db
[root@EmbedSky /]	# sqlite_test tes st1'.1):"	st.db "insert	into Add a list o	f
<pre>> tbl0 values('te</pre>	#`sqlite_test tes st2',2);"	.c.up insert	inco	-
[root@EmbedSky /] > from tbl0;" name = test1 number = 1	# <u>sqlite_test tes</u>	st.db "select	Obtain a list of date in test.db	
name = test2 number = 2				
[root@EmbedSky /]	#			
- 100 CO 2000				