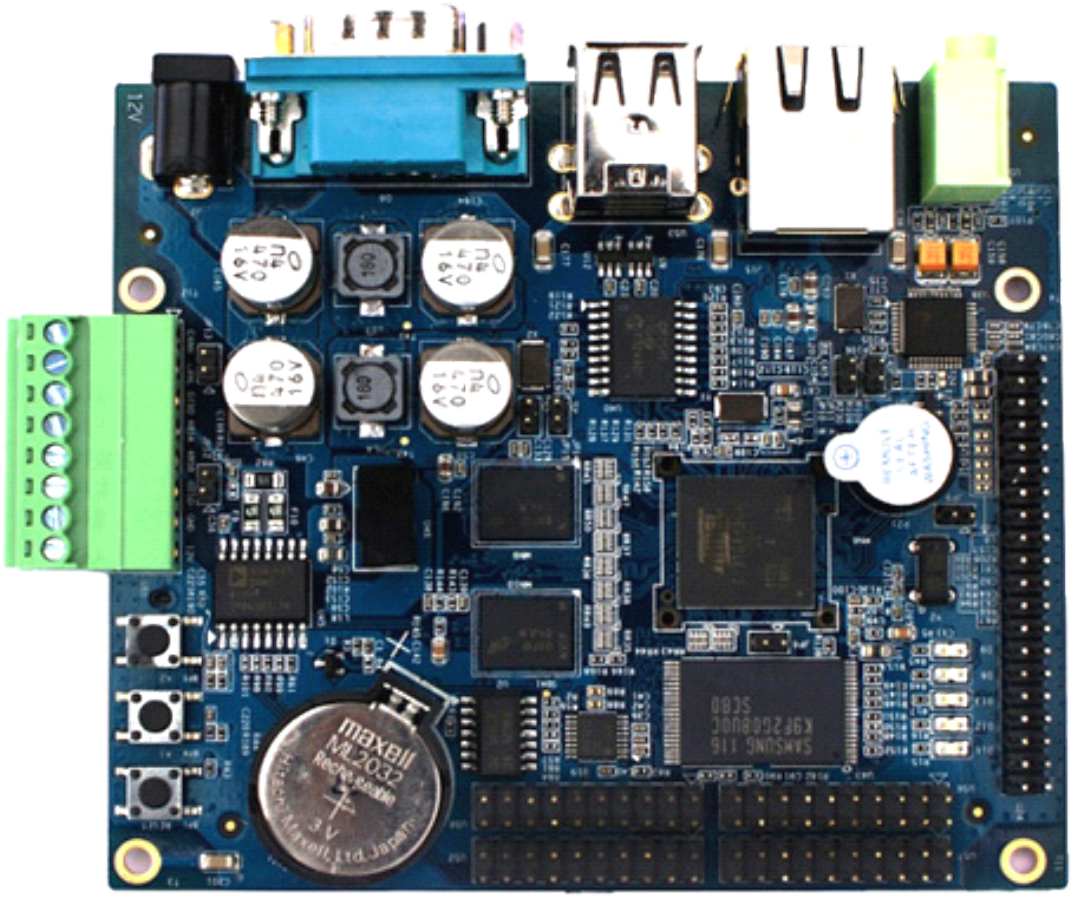
SBC6845
Single Board Computer



User Manual

（WinCE6.0）

Version 1.2 – Jan. 18, 2013

Copyright Statement:

- SBC6845 and its related intellectual property are owned by Shenzhen Embest Technology Co., Ltd.
- Shenzhen Embest Technology has the copyright of this document and reserves all rights. Any part of the document should not be modified, distributed or duplicated in any approach and form with the written permission issued by Embest Technology Co., Ltd.

Disclaimer:

- Shenzhen Embest Technology does not take warranty of any kind, either expressed or implied, as to the program source code, software and documents in the CD/DVD-ROMs provided along with the products, and including, but not limited to, warranties of fitness for a particular purpose; The entire risk as to the quality or performance of the program is with the user of products; Should the program prove defective, the user of products assumes the cost of all necessary servicing, repair or correction.

Revision History:

Version	Date	Note
1.0	2011-10-24	Original Version
1.1	2012-01-16	Added MartrixKey driver and updated display driver
1.2	2013-01-18	Revised the layout

Table of Contents

Chapter 1 Product Overview

1.1 Introduction

SBC6845 is an ARM embedded single board computer designed by Shenzhen Embest Technology Co., Ltd. It is a small-sized board based on Atmel's industrial microprocessor AT91SAM9G45 and featured with 128MB DDR2 SDRAM, 256MB NAND Flash and 4MB Data Flash, as well as abundant interfaces including 5 RS232 serial interfaces (COM2 is RS485), a CAN interface, an Ethernet interface, a high-speed USB host interface, a SD/MMC card interface and an audio output interface.

SBC6845 is designed to satisfy the different requirements of various fields such as industrial control, intelligent instrumentation, data acquisition and analysis, medical products and network equipments.

1.2 Packing List

- A SBC6845 board
- A serial cable
- A network cable
- A USB cable
- A 12V power adapter
- A LCD touch-screen (optional item, available in 4.3-inch 480*272 or 7-inch 800*480)
- A CD-ROM

1.3 Product Features

- Dimensions: 106.5mm x 94mm (6-layer PCB)
- Operation temperature: -40 ~ +85℃

- Operating Humidity: 0% ~ 90%
- Power Supply: 12V/1.25A
- Processor: Atmel AT91SAM9G45
- On-Board Memories:
  - 64MB*2 DDR2 SDRAM
  - 256MB NAND Flash
  - 4MB Data Flash
- Audio/Video Output Interfaces:
  - A 3.5mm jack
  - A 2*20-pin DIP interface for LCD touch-screen
  - A buzzer
- Data Transfer Interfaces:
  - 5 serial interfaces
  - A CAN 2.0 interface (isolated)
  - 2 USB 2.0 interfaces
  - A 10/100Mb Ethernet interface
  - A SD card slot (hot plugging)
  - Pins for SPI, I2C, PWM, ADC, Keypad and GPIO

1.4 Components on SBC6845

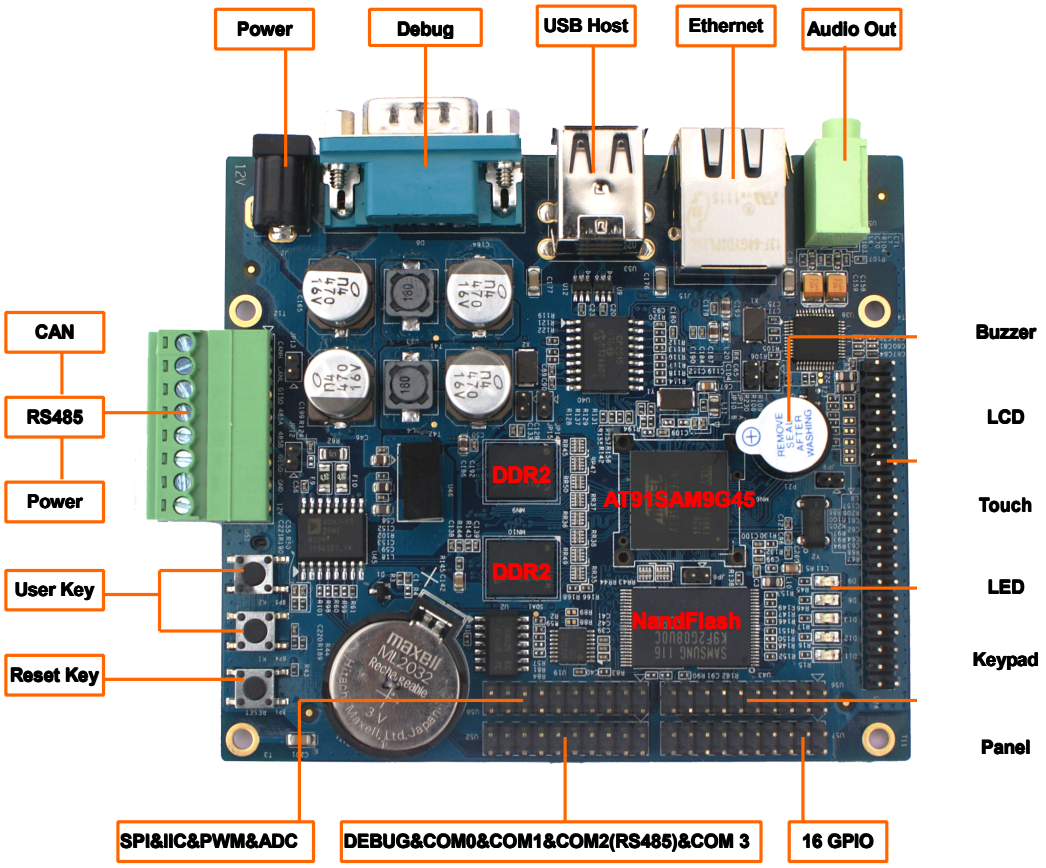


Figure 1-1 Components on SBC6845

1.5 Hardware Dimensions

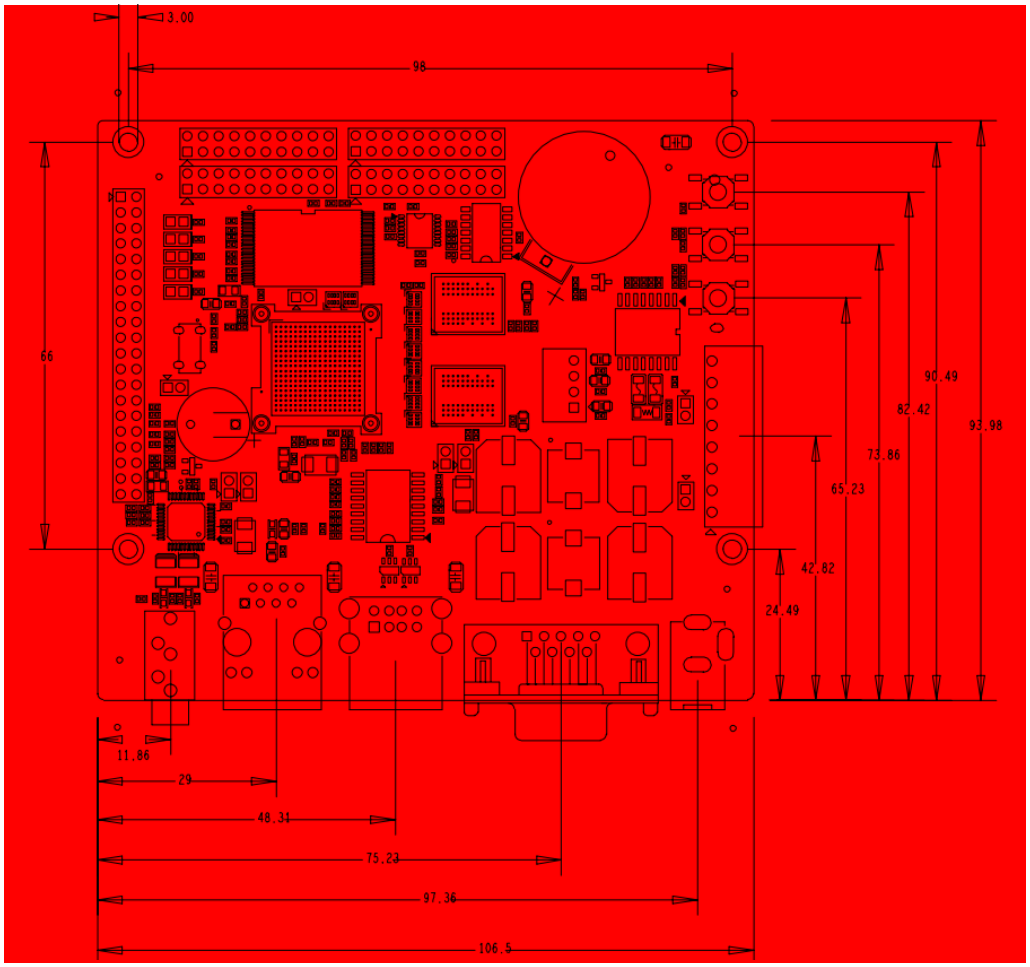
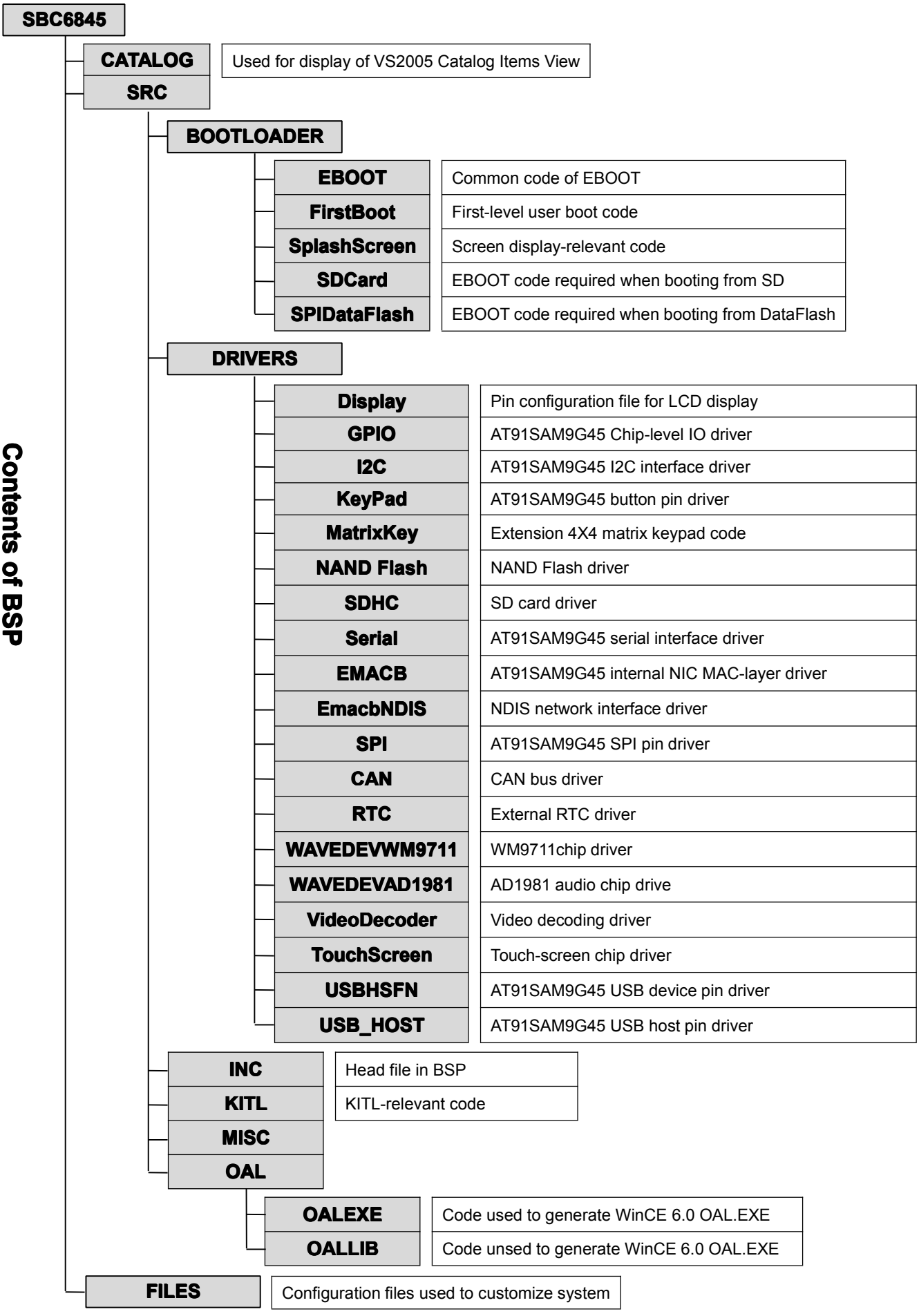


Figure 1-2 Hardware Dimensions

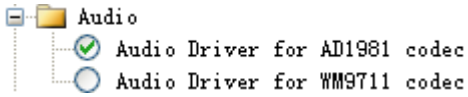
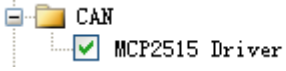
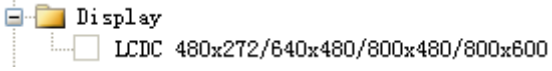
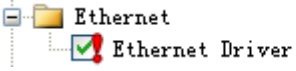
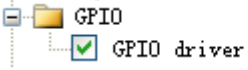
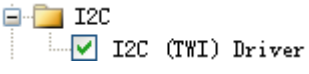
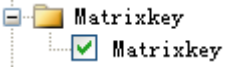
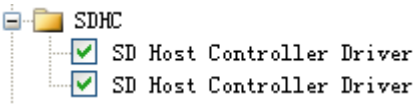
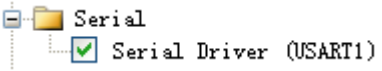
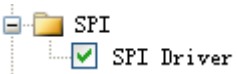
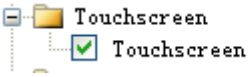
1.6 BSP Package in CD-ROM

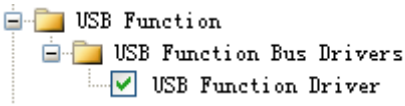
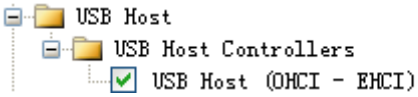
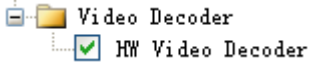
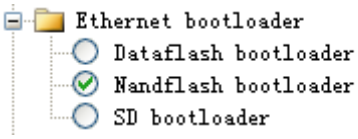
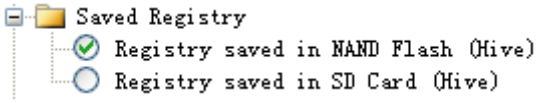
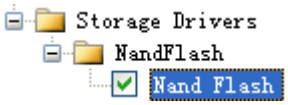
The CD-ROM provided along with SBC6845 contains a BSP package that helps development of WinCE system. Users can make a custom system running on SBC6845 platform by utilizing the package. The table shown below lists out the contents of the BSP attached with the corresponding descriptions.

Contents of BSP



The table shown below lists the module driver options that are available in BSP when viewing them in Visual Studio 2005, as well as their corresponding descriptions.

Modules	Descriptions
	Check the option if AD1981 external audio codec is required
	Check the option and SPI interface driver as well if CAN bus is required.
	By default, LCD driver is enabled to support 4.3-inch, 5.6-inch and 7-inch screen provided by Embest. The adoption of an appropriate display configuration will be determined automatically according to states of jumpers JP10 and JP11.
	Check the option if Ethernet is required.
	Check the option if GPIO is required.
	Check the option if I2C is required.
	Check the option if matrix keypad is required. The corresponding pins of U56 connector on SBC6845 include PC16, PC17, PC18, PC19, PC22, PC23, PC24 and PC25.
	Check the option if SD card is required.
	Check the option if USART serial interface extended from AT91SAM9G45 on SBC6845 is required.
	Check the option if SPI is required.
	Check the option if touch-screen is required.

Modules	Descriptions
	Check the option if USB device is required.
	Check the option if USB host is required.
	Check the option if hardware decoding is required.
	To generate FIRSTBOOT and EBOOT that boot system from DataFlash, please check DataFlash bootloader; To generate FIRSTBOOT and EBOOT that boot system from NAND Flash, please check NandFlash bootloader.
	Check the option if Hive registry is required.
	Check the option if the spare space is required to save data after WinCE is written in NAND flash.

Chapter 2 System Boot-up and Testing

2.1 System Boot-up

Currently SBC6845 WinCE 6.0 system can boot up from NAND Flash or serial interface. This section will briefly introduce the boot-up principle of SBC6845 WinCE 6.0 system and give you a detailed explanation of the booting process from NAND Flash.

2.1.2 Brief Introduction to Booting Process

When SBC6845 is powered up, the ROMBOOT program embedded in the CPU AT91SAM9G45S will be running automatically to check if there is a BootLoader in NAND Flash. If so, the BootLoader would be copied to SRAM of AT91SAM9G45S and executed. If not, the program will continue searching in DataFlash. If eventually there is no any BootLoader found in both flash memories, the program will shift to downloading mode waiting for input from serial interface. The following figure shows the system booting process.

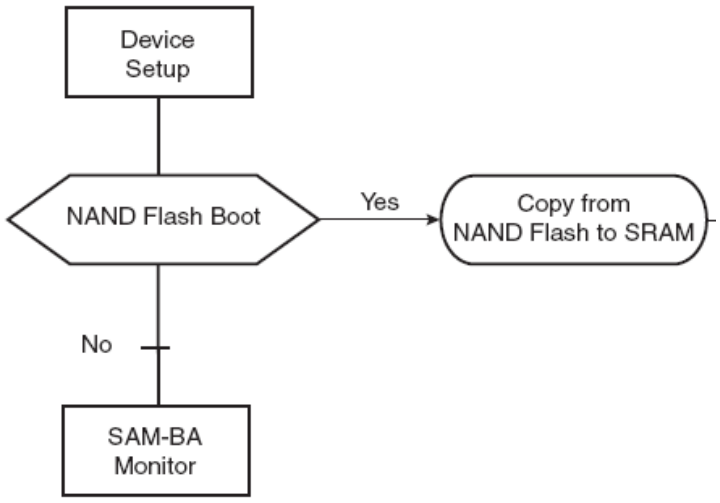


Figure 2-1 System Booting Process

2.1.3 Booting from NAND Flash

When system boots up from NAND Flash, the first-level boot code FIRSTBOOT at 0x00000000 in NAND Flash is copied to the SRAM of AT91SAM9G45 and executed. FIRSTBOOT will initialize AT91SAM9G45, SDRAM and NAND Flash, and copy the second-level boot code EBOOT at 0x00080000 in NAND Flash to the SDRAM on SBC6845 to execute EBOOT, which by default will be proceeding to copy WinCE image named NK.bin from the address 0x00080000 in NAND Flash to SDRAM and let operating system take over. Furthermore, EBOOT is used to manage bottom-layer hardware and configure settings related to data sharing with operating system.

The figure shown below is the storage structure of NAND Flash;

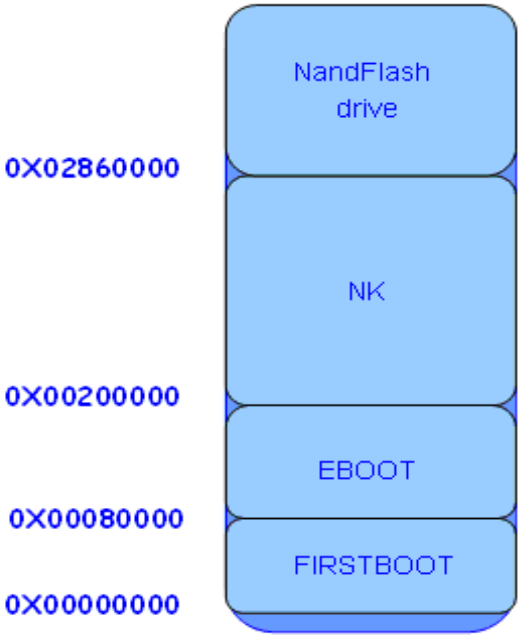


Figure 2-2 Storage of NAND Flash

The detailed booting process of WinCe is shown below;

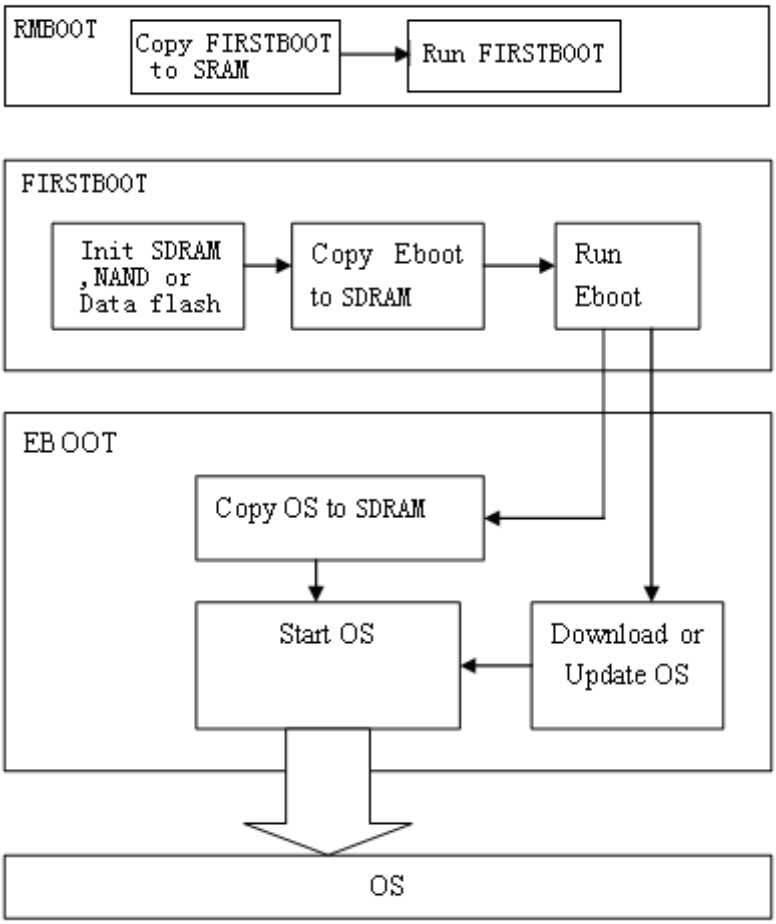


Figure 2-3 Detailed WinCE Booting Process

2.2 System Burning

By default, SBC6845 is installed with a Linux system, and therefore users have to burn the NAND Flash on the board with a WinCE system by themselves. The following contents will introduce how to burn the flash with a whole system step by step by using the WinCE6.0 provided in the CD-ROM and a burning tool SAM-BA from Atmel, or using VS2005 WinCE6.0 development environment or other TFTP server (e.g. CEDownload.exe) to download a WinCE system image NK.bin to the board through EBOOT and network. (The EBOOT generated by current BSP version can only support downloading WinCE system image file NK.bin, not FIRSTBOOT and EBOOT.)

1.1.1 Hardware Connections and Configurations

1） Use a cross-over serial cable and a USB cable to connect SBC6845 to your PC.

2） Configure HyperTerminal on your PC according to the settings shown in the following figure;

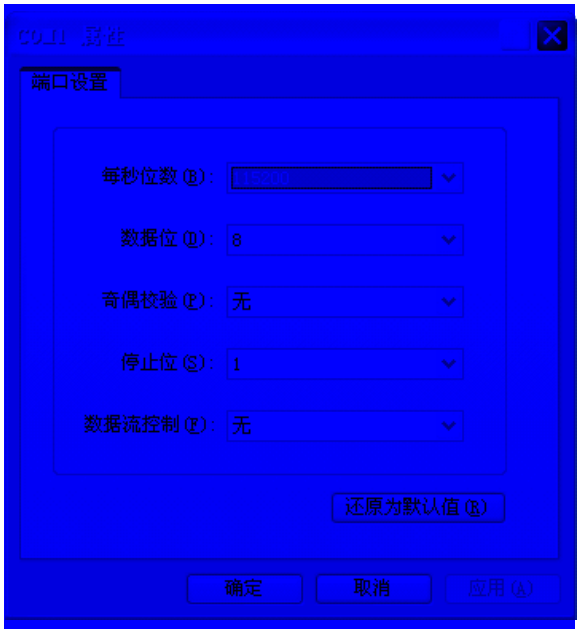


Figure 1-1 Configure HyperTerminal

3） Connect the power adapter provided along with the product to SBC6845 and press Reset button on the board;

Note:

- Please ensure the jumpers JP8 and JP14 on SBC6845 are disconnected before you power up the board in case there is any boot code in NAND Flash or Data Flash which will lead to malfunction of the burning tool SAM-BA.
- If a window pops up and prompts you to install USB device after you power up the board, please select "**Install Automatically**" and click **Next** to finish installation process.

1.1.2 Erasing NAND Flash

1） Start SAM-BA to open the following window;

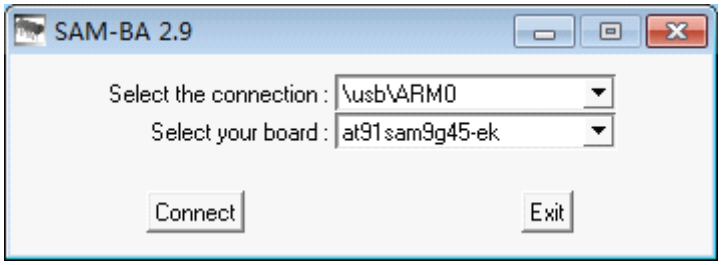


Figure 1-2 SAM-BA Initial Window

If the hardware is connected properly, **\usb\ARM0** will appear in **Select the connection** text box. Now select **at91sam9g45-ek** in the **Select your board** drop-down menu and click **Connect**.

Note:

Please short JP8 on SBC6845 before you click **Connect**.

2） Then the main window of SAM-BA will be shown as below. Click **NandFlash** tab, and select **Enable NandFlash** in **Scripts** drop-down menu and then click **Execute**;

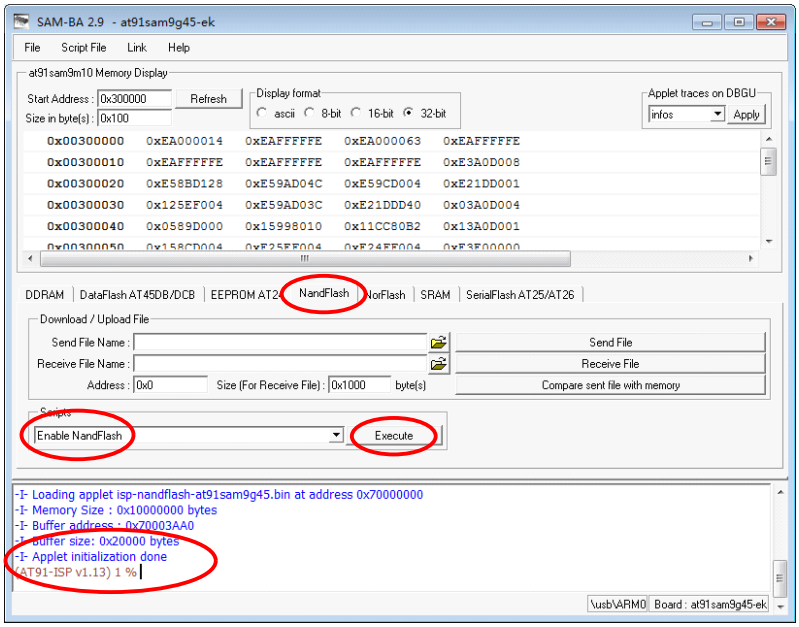


Figure 1-3 Enable NAND Flash

The information box at the bottom of the window will show the details when you operate.

3） Select **Erase All** in the **Scripts** drop-down menu and click **Execute** to erase the contents of NAND Flash as the figure shown below;

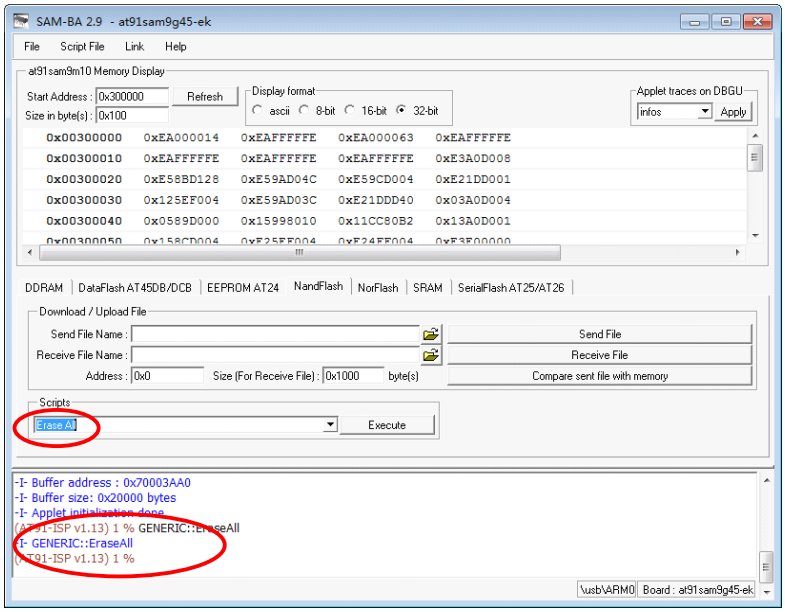


Figure 1-4 Erase NAND Flash

1.1.3 Writing Bootloader and System Image

4） Select **Send Boot File** in the **Scripts** drop-down menu and click **Execute** to open the window shown below;

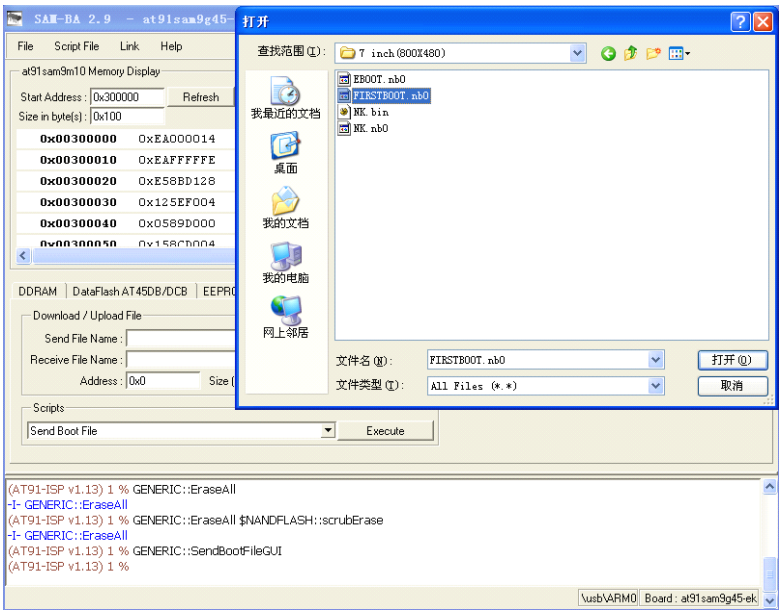


Figure 1-5 Select FIRSTBOOT.nb0

Find FIRSTBOOT.nb0 on your PC and click **Open**. SAM-BA will burn the NAND Flash starting from 0x00000000 to write FIRSTBOOT.nb0.

Note:

🕮 If you can not find FRSTBOOT.nb0 in the above window, please click **File Type** drop-down menu and select **All Files (*.*)** to display all the files.

5） Type **0x80000** in **Address** text box within **Download/Upload File** block of the window and click  on the right of **Send File Name** text box to open the following window;

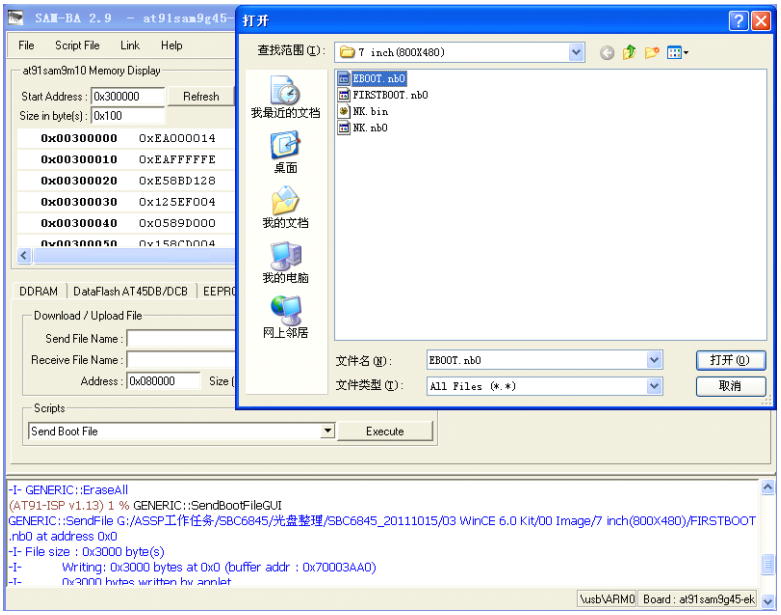


Figure 1-6 Select EBOOT.nb0

Find EBOOT.nb0 on your PC and click **Open**, and then click **Send File** on the right of **Send File Name** text box to write EBOOT.nb0 into NAND Flash.

6） Type **0x200000** in **Address** text box within **Download/Upload File** block of the window and click  on the right of **Send File Name** text box to open the following window;

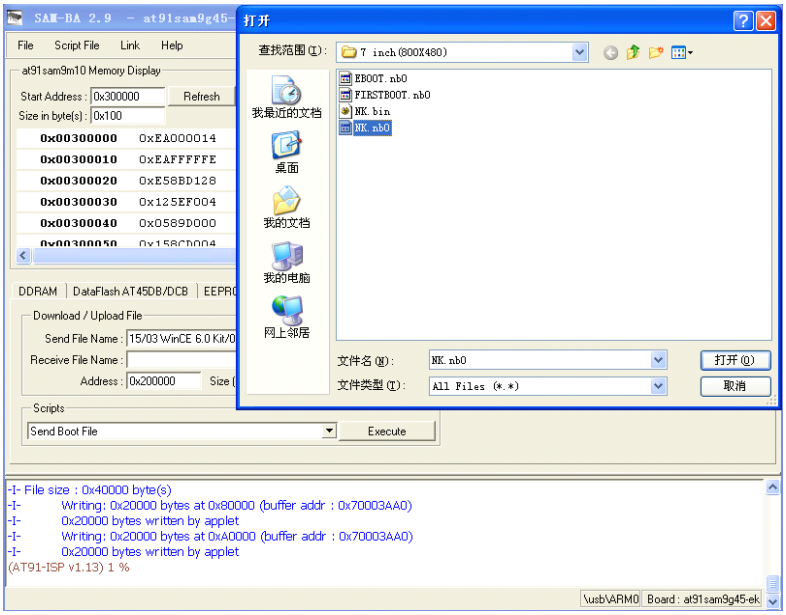


Figure 1-7 Select NK.nb0

Find NK.nb0 on your PC and click Open, and then click **Send File** on the right of **Send File Name** text box to write NK.nb0 into NAND Flash.

WinCE 6.0 image has been written to SBC6845. You can reboot the board and enter the system now.

1.2 Updating the System

Updating system refers to implementation of replacing kernel file NK.nb0 only when there is already a complete system in NAND Flash where FIRSTBOOT.nb0, EBOOT.nb0 and NK.nb0 exist.

This section will show you how to implement system kernel updating by using CEDownload.exe software and network interface on SBC6845.

Note:

Although kernel file can be updated separately by using SAM-BA and USB interface, there is possibility that the system might be unstable if writing it without erasing NAND

Flash in advance. Therefore, this method is not recommended. While in the process of downloading kernel by using CEDownload.exe, EBOOT will automatically erase the data at the corresponding address.

1） Use a cross-over serial cable and a network cable to connect SBC6845 to your PC;

2） Run **CEDownload.exe** under **\03 WinCE 6.0 Kit\04 Tools** in the CD-ROM on your PC to open **WinCE Download Server** window as shown below;

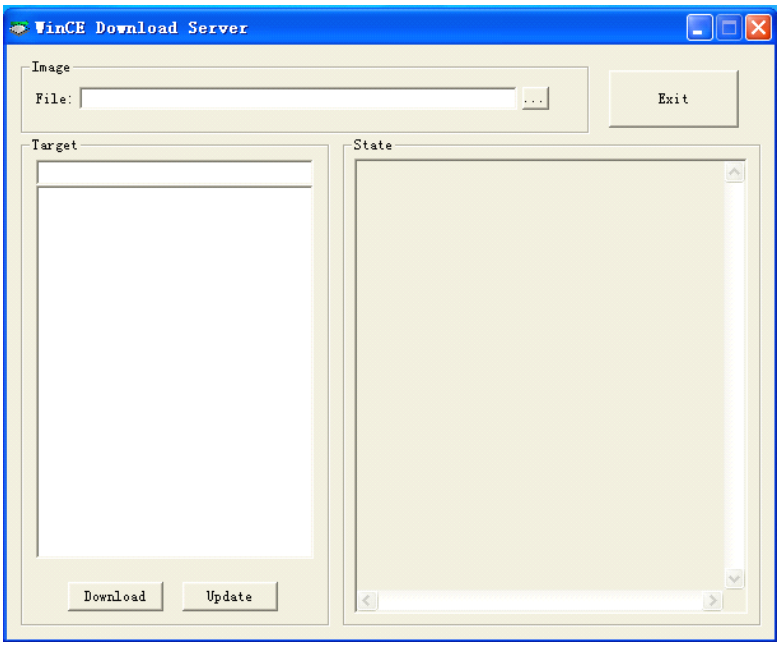


Figure 1-8 WinCE Download Server Window

Click ... on the right of **File** text box to find the file **NK.bin** on your PC.

Note:

The kernel file to be downloaded from WinCE Download Server is named NK.bin, but not NK.nb0. The file will be eventually converted to NK.nb0 by EBOOT and written to NAND Flash.

3） Connect the power adapter to the board and press **SPACE** key on your PC's keyboard to enter EBOOT menu as shown in the following HyperTerminal window;

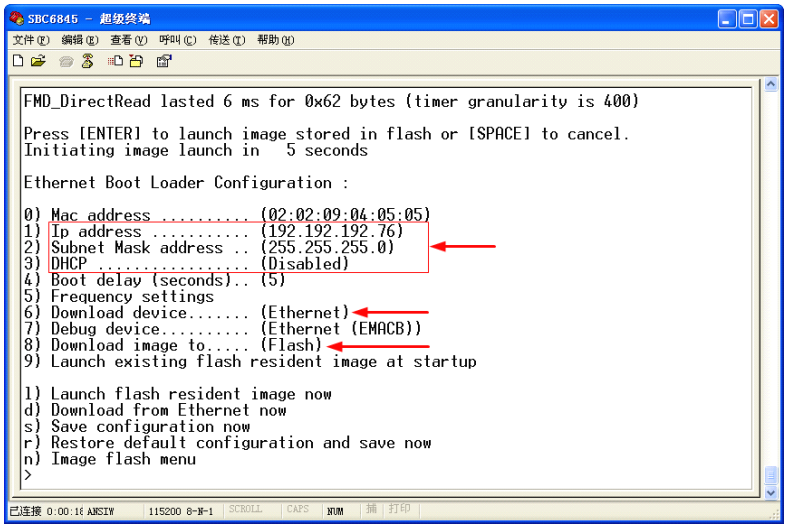


Figure 1-9 EBOOT Menu

Configure the entries marked with red arrows in the above window. Press **Enter** on your keyboard to confirm the change in each entry.

Note:

- The IP addresses in the window should be set as the same as your PC's network segment in order to ensure a proper communication.
- The EBOOT program generated from BSP contained in the CD-ROM can only support downloading NK.bin through network cables, not FIRSTBOOT and EBOOT.

4） Type **d** when configuration is finished. The HyperTerminal will show the following information;

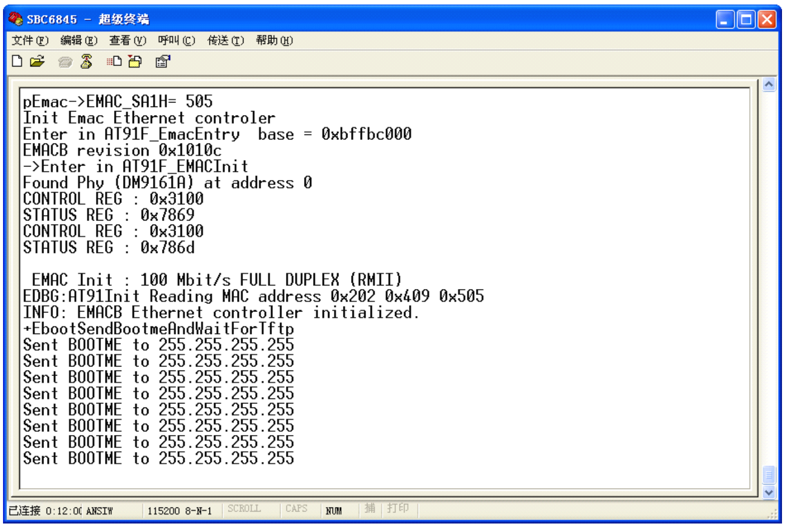


Figure 1-10 Reading to Download

When you see **Sent BOOTME to 255.255.255.255** in the HyperTerminal, the network is ready to download kernel files.

5） Click **Download** in the **WinCE Download Server** window. The HyperTerminal will show information as the follows;

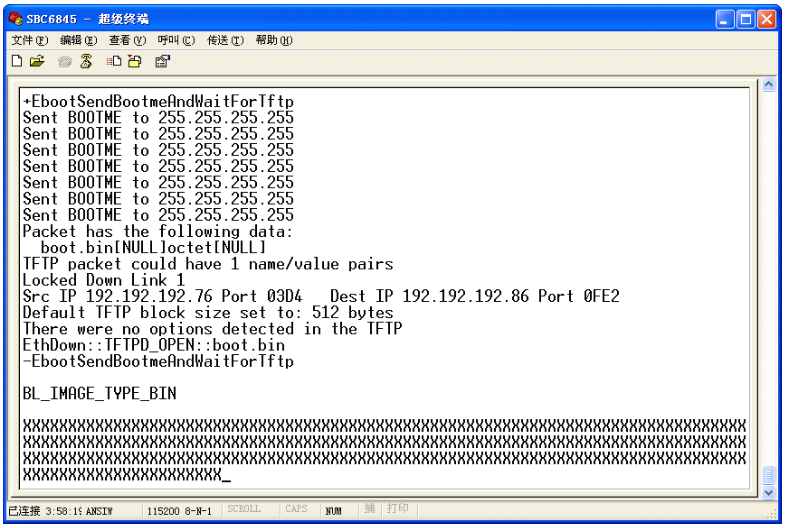


Figure 1-11 Downloading NK.bin

After downloading is completed, the system will reboot and load WinCE system automatically.

Chapter 2 Generating WinCE System Image

The CD-ROM provided along with SBC6845 contains a BSP package that includes hardware drivers and operating system source code. You can generate a WinCE image by directly using the BSP package, or make a customized system by modifying the source code in the BSP. This chapter will introduce in detail how to generate a new WinCE system based on the BSP package.

2.1 Creating WinCE Development Environment

Before you can customize a system, you need to create an appropriate environment on your PC first, for example, install tools such as Windows Embedded CE 6.0 Development Platform. The table shown below lists the software required to be install on your PC. Please visit Microsoft's website to download and follow the order in the table to install them.

Table 2-1 Development Platform Software

No.	Names
1	Visual Studio 2005
2	Visual Studio 2005 SP1
3	Visual Studio 2005 SP1 Update for Vista (vista system require)
4	Windows Embedded CE 6.0 Platform Builder
5	Windows Embedded CE 6.0 SP1
6	Windows Embedded CE 6.0 R2
7	Windows Embedded CE 6.0 Product Update Rollup 12/31/2008
8	Windows Embedded CE 6.0 R3
9	Windows Embedded CE 6.0 Product Update Rollup 12/31/2009
10	ActiveSync 4.5
11	Windows Mobile 6 Professional SDK

Note:

- You can specify the path where you want the software to be installed, but please ensure other options remain unchanged.
- Hereafter the default installation path of all the development platform software is D:\WINCE600\.

2.2 Generating Image without Changes

If you don't need to change anything in the BSP from CD-ROM, you can generate a system image fast by follow the steps listed below.

1） Create a folder named **OSDesigns** under the directory **D:\WINCE600** where Visual Studio 2005 is installed, and decompress **SBC6845.rar** saved under **\03 WinCE 6.0 Kit\02 Project** of the CD-ROM to the new folder.

2） Decompress SBC6845.rar saved under **\03 WinCE 6.0 Kit\01 BSP** of the CD-ROM to **D:\WINCE600\PLATFORM** (The directory PLATFORM is created automatically after installation of Windows Embedded CE 6.0);

3） Decompress ATMEL.rar saved under **\03 WinCE 6.0 Kit\01 BSP** of the CD-ROM to **D:\WINCE600\PLATFORM\COMMON\SRC\SOC**;

4） Open the project file **SBC6845.sln** saved under the directory by Visual Studio 2005, and select 生成 > 生成解决方案 on the menu bar to generate system image (The 生成 menu is available only when a project is opened).

2.3 Making a Customized Image

The following contents will take an example of handheld device operating system to introduce the customization process based on the BSP in the CD-ROM.

Note:

- The functions selected in the following example are only for your reference. You can make a different selection according to your requirement.

1） Please do the step 2 and 3 in section 3.2 (If you already did before, you may proceed with the following steps);

2） Start Visual Studio 2005 and select 文件 > 新建 > 项目 on the menu bar to open new project window as shown below (If it is the first time you run Visual Studio 2005, the program will prompt you to select a default setting for the software. Please select 常规开发设置);

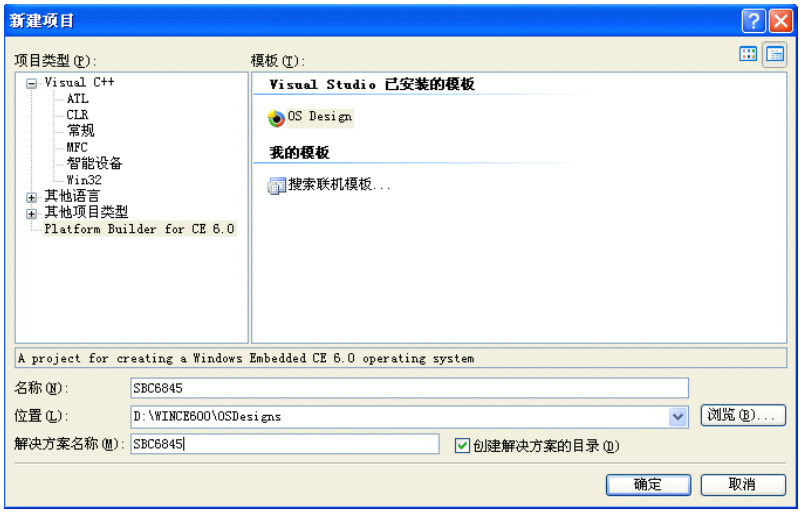


Figure 2-2 Create New Project

Click **Platform Builder for CE 6.0** in the tree-view on the left, and enter a project name and a path where project files will be saved at the bottom part of the window, and then click **OK**.

3） Click **Next** in the pop-up Design Wizard window as shown below;

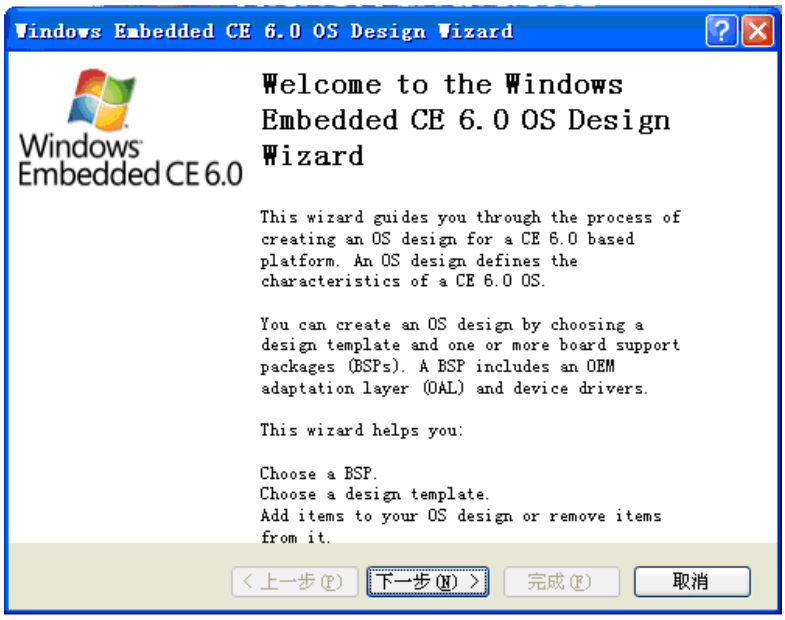


Figure 2-3 New Project Design Wizzard

4） Check **SBC6845:ARMV4I** in the following window and click **Next**;

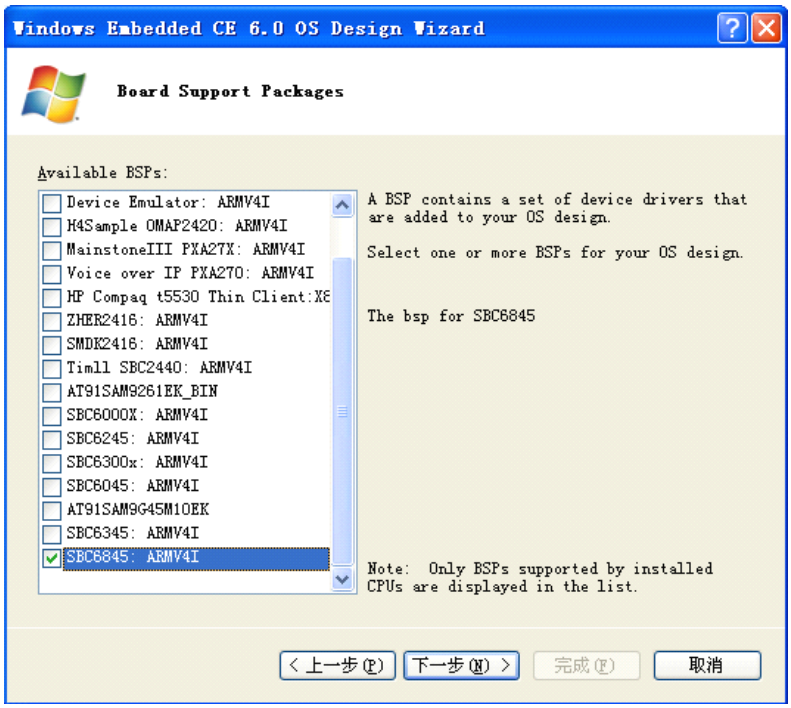


Figure 2-4 Select a BSP

5） Select **PDA Device** in the list of design templates and click **Next**;

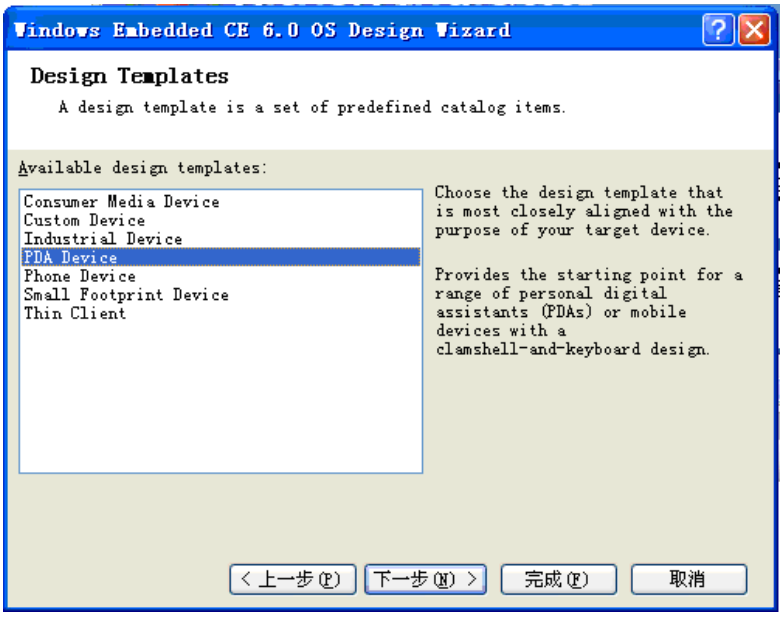


Figure 2-5 Select a Design Template

6） Select **Mobile Handheld** in the list of variants and click **Next**;

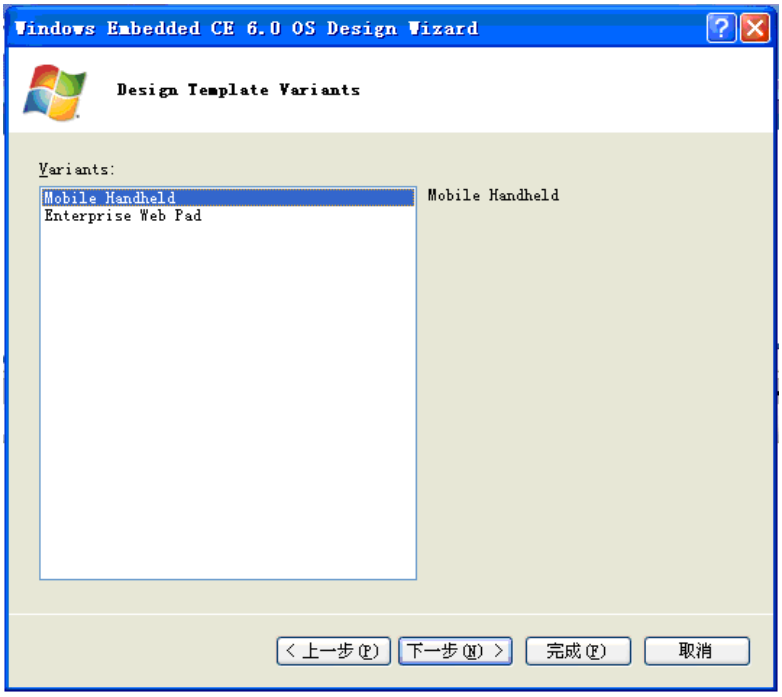


Figure 2-6 Select a Variant

7） Check the functions you need in the application list and click **Next**;

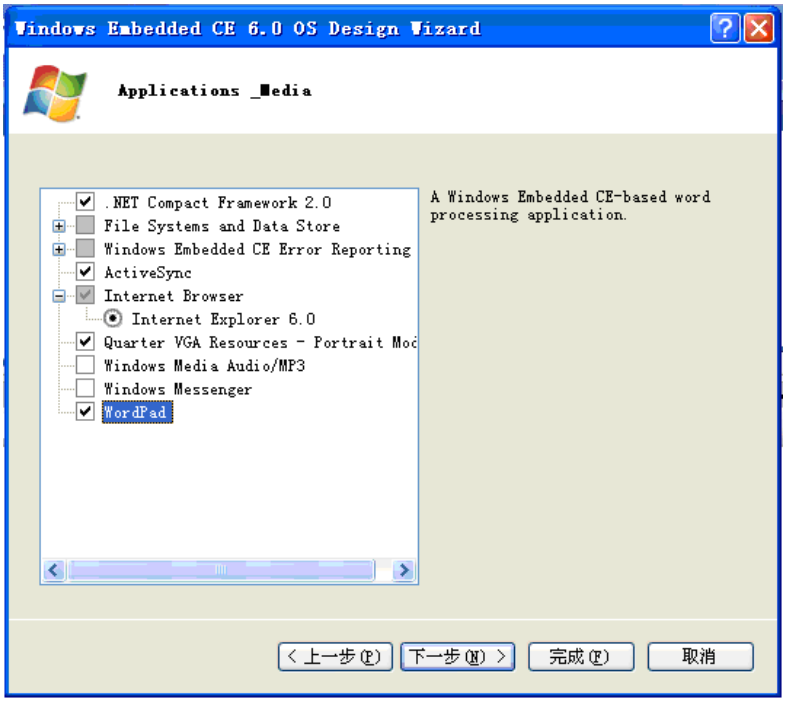


Figure 2-7 能 Select Applications and Media

8） Select the functions required in the list of networking and communications, and uncheck the options **Bluetooth** and **IrDA** (SBC6845 does not support Bluetooth and infrared communication), and then click **Next**;

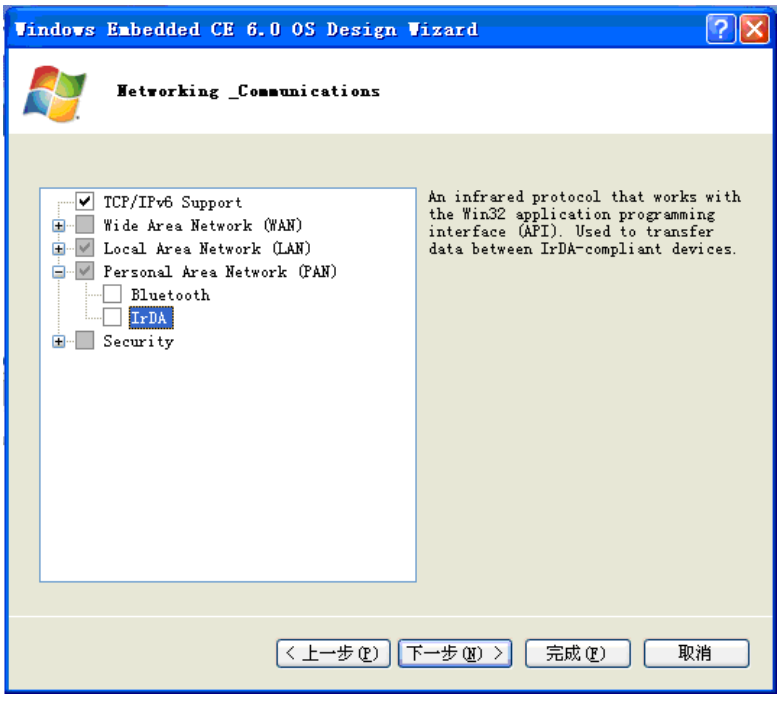


Figure 2-8 Select Networking and Communications

9） Click **Finish** in the window as shown below to finish the customization process;

Figure 2-9 Click **Finish**

10） Click **Acknowledge** in the window shown below;

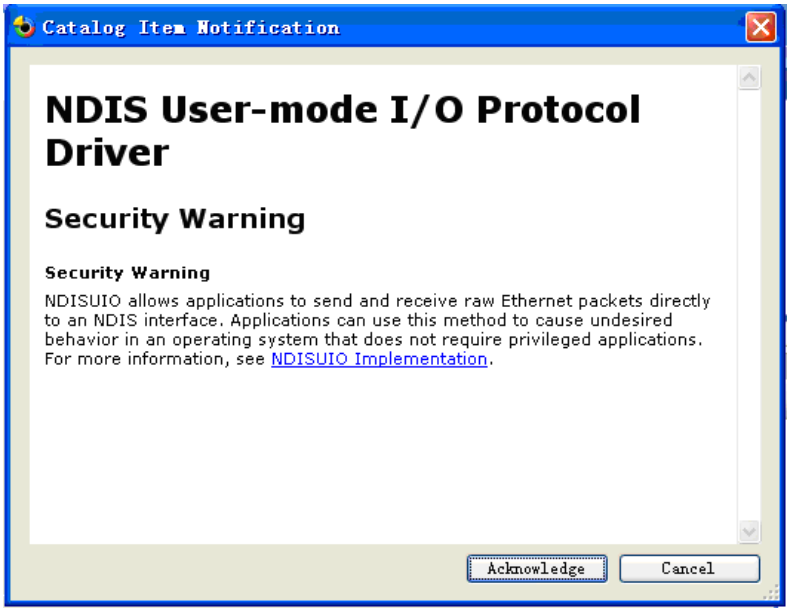


Figure 2-10 Click **Acknowledge**

11） You can view and change configurations in **Catalog Items View**;

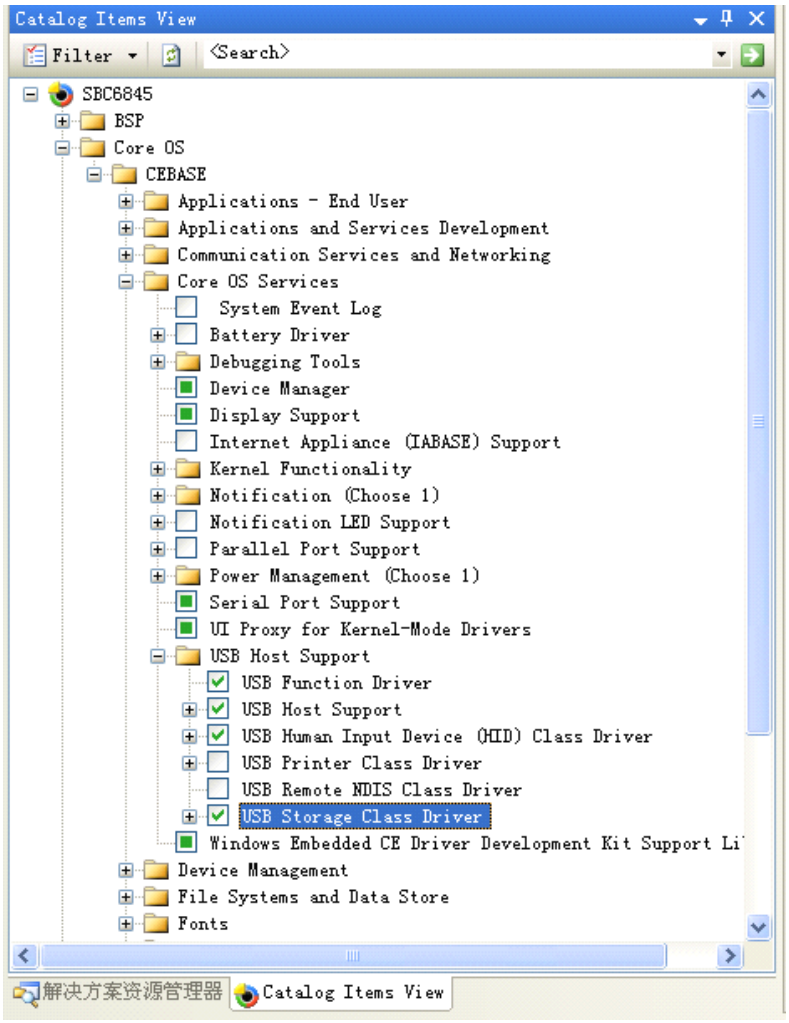


Figure 2-11 Customization Configurations

Here you can modify the configurations for functional components and module drivers of SBC6845 based on your requirement (Please refer to section 1.6 for detailed information on drivers), for example, you can check **USB Function Driver** and **USB Storage Class Driver** under the branch USB Host Support, and check module drivers under **Third Party > SBC6845: ARMV4I**);

12） After you finish configuration, click the drop-down menu next to  on the tool bar of Visual Studio 2005 and select **SBC6845 ARMV4I Release** as the compilation type as shown below;

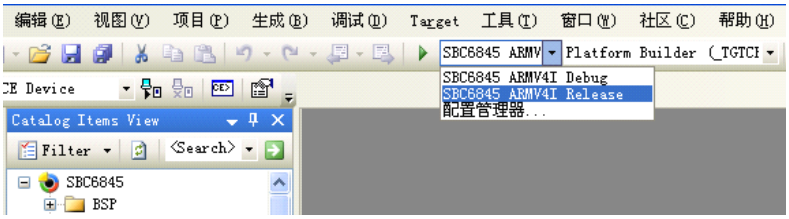


Figure 2-12 Select Compilation Type

13） Click **项目 > SBC6845 属性** on the menu bar to open SBC6845 Properties window as shown below;

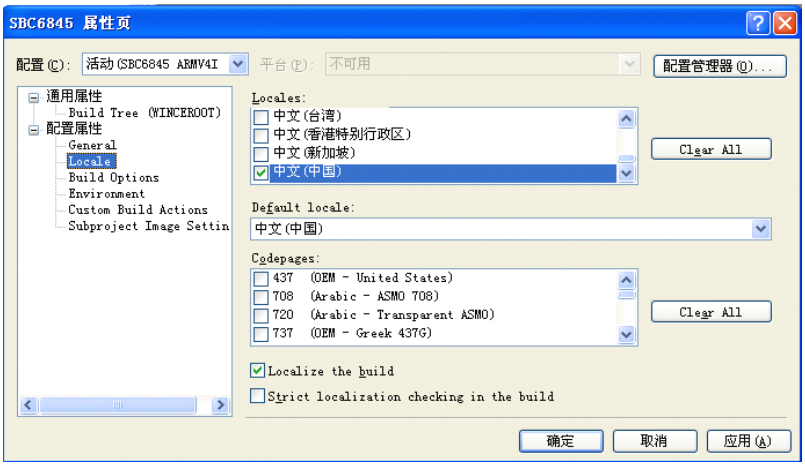


Figure 2-13 SBC6845 Properties

Click **Locale** in the tree-view on the left and check the languages for the operating system, and then click **OK**.

14） Click **Build Options** in the tree-view of SBC6845 Properties window and check options on the right for system compilation.

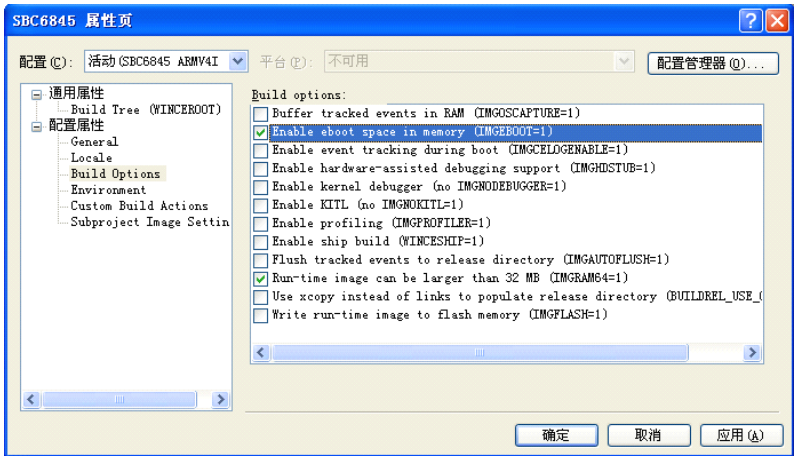


Figure 2-14 Compilation Options

Note:

📖 The option **Enable eboot space in memory** should be checked under any circumstance.

15） Click 生成 > 生成解决方案 on the menu bar of Visual Studio 2005 to start compilation process;

Visual Studio 2005 will generate 6 image files - FIRSTBOOT.nb0, EBOOT.nb0, NK.nb0, FIRSTBOOT.bin, EBOOT.bin and NK.bin and save them under D:\WINCE600\OSDesigns\SBC6845\SBC6845\RelDir\SBC6845_ARMV4I_Release\ when compilation process is completed.

Appendix

Brief Introduction to EBOOT Menu

EBOOT is a program saved in the NAND Flash on SBC6845. When the board is powered up, the system will conduct a 5-second countdown. By pressing **SPACE** on your PC's keyboard within the 5 seconds, you can enter EBOOT menu and see it in HyperTerminal as shown below;

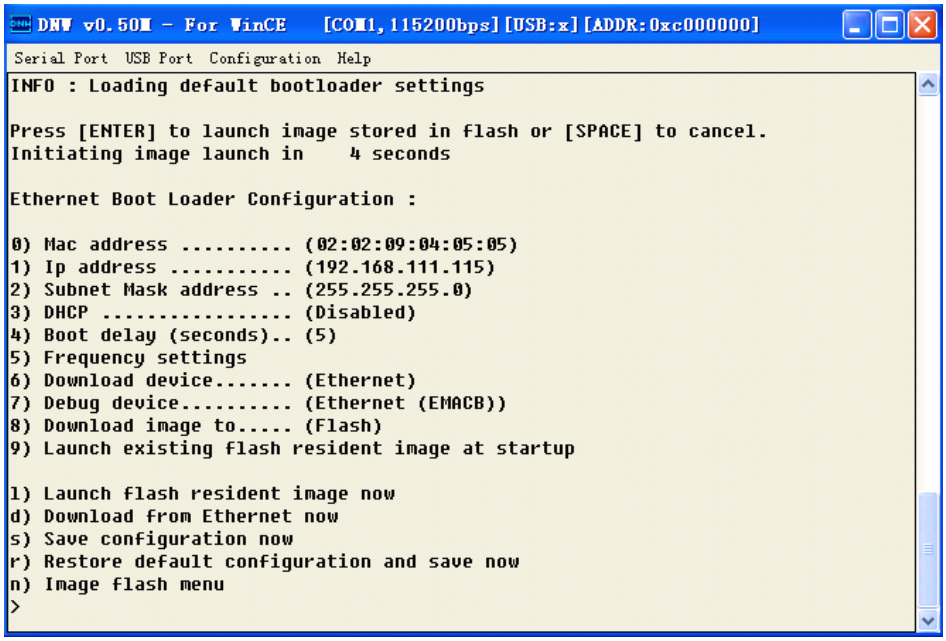


Figure 1 EBOOT Menu

Through EBOOT menu you may implement many operations including setting MAC and IP addresses, configuring for writing images, and erasing NAND Flash. The table shown below lists all the menu entries and their respective brief descriptions.

Table 1 EBOOT Menu Entries

Keys	Menu Entries	Descriptions
0	Mac address	Set MAC address for SBC6845
1	Ip address	Set IP address for SBC6845
2	Subnet Mask address	Set subnet mask for SBC6845
3	DHCP	Enable or disable automatic IP allocation
4	Boot delay	Set delay before booting the system

Keys		Menu Entries	Descriptions
5		Frequency settings	View frequency of AT91SAM9G45S
6		Download device	Select device to download system
7		Debug device	Select device to debug under EBOOT
8		Download image to Flash	Specify downloading destination of system files
		Download image to SDRAM	
9		Launch existing Flash resident image at startup	Select loading the existing system or downloading a system when booting up
		Download new image at startup	
l		Launch Flash resident image now	Loading the existing system immediately
d		Download from Ethernet now	Downloading a system from Ethernet immediately
s		Save configuration now	Save EBOOT configurations immediately
r		Restore default configuration and save now	Reset EBOOT configurations to default settings
n	1	Erase all sectors	Erase all the contents in NANF Flash
	2	Enter manually the image parameters	Enter image parameters manually
	3	Quit	Exit current menu

Note:

- Changing parameters of system image is normally not recommended in order to avoid unexpected errors.
- The system image parameters Physical Start Address, Starting ip andTotal ROM size 三 can be viewed by clicking 生成 **> Open Release Directory in Build Window** on the menu bar of Visual Studio 2005.

Making Boot-Up Logo

The section will show you how to make a logo with a software tool Image2Lcd and display it when SBC6856 is booting up.

1） Find Image2Lcd under\ **03 WinCE 6.0 Kit\04 Tools** of the CD-ROM and install it on your PC;

2） Start Image2Lcd and click **Open** on the menu bard of the software window to load the image you need；

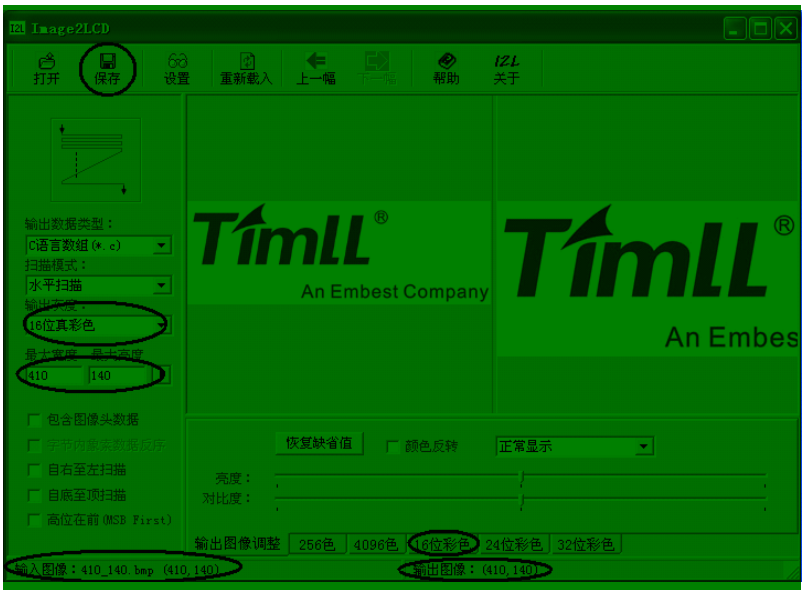


Figure 2-15 打开图片

In the three drop-down menus on the left of the window, select **C 语言数据（*.c）**，**水平扫描** and **16 位真彩色** respectively. Enter **410** and **140** into 最大宽度 and 最大高度 text box under the drop-down menus, and then click **16 位彩色** tab at the bottom of the window and set color as **R:G:B=5:6:5**.

Note:

Please ensure the image you selected is a BMP bit image with a 410X140 resolution and 24-bit color bits. If the original image is not consistent with these properties, please modify it with other graphics software first.

3） Click **Save** on the menu bar to save the image as *.h, for example, Logo.h;

4） Open Logo.h with notepad and copy the entire data array as shown below;

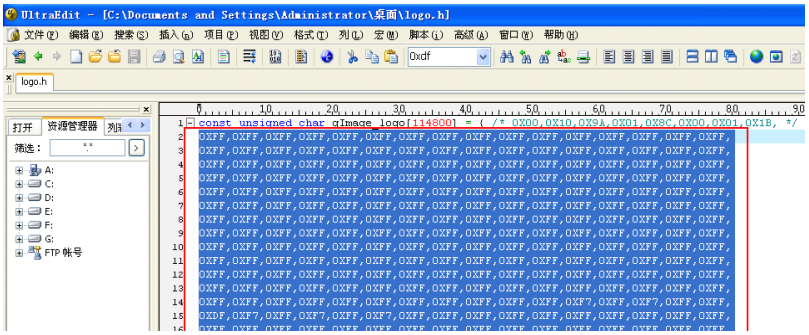


Figure 2-16 Copy Data Array

5） Find BitMap.h under D:\WINCE600\PLATFORM\SBC6845\SRC\BOOTLOADER\SplashScreen and open it with notepad, and then replace the contents between #else and #endif of header_data[] data array. After compiling and generating a new eboot.nb0 and writing it into NAND Flash, you can see the logo when the board is booting up.

Technical Support and Warranty

Technical Support

Embest Technology provides its product with one-year free technical support including:

- Providing software and hardware resources related to the embedded products of Embest Technology;
- Helping customers properly compile and run the source code provided by Embest Technology;
- Providing technical support service if the embedded hardware products do not function properly under the circumstances that customers operate according to the instructions in the documents provided by Embest Technology;
- Helping customers troubleshoot the products.

The following conditions will not be covered by our technical support service. We will take appropriate measures accordingly:

- Customers encounter issues related to software or hardware during their development process;
- Customers encounter issues caused by any unauthorized alter to the embedded operating system;
- Customers encounter issues related to their own applications;
- Customers encounter issues caused by any unauthorized alter to the source code provided by Embest Technology;

Warranty Conditions

1） 12-month free warranty on the PCB under normal conditions of use since

the sales of the product;

2） The following conditions are not covered by free services; Embest Technology will charge accordingly:

A. Customers fail to provide valid purchase vouchers or the product identification tag is damaged, unreadable, altered or inconsistent with the products.

B. Products are damaged caused by operations inconsistent with the user manual;

C. Products are damaged in appearance or function caused by natural disasters (flood, fire, earthquake, lightning strike or typhoon) or natural aging of components or other force majeure;

D. Products are damaged in appearance or function caused by power failure, external forces, water, animals or foreign materials;

E. Products malfunction caused by disassembly or alter of components by customers or, products disassembled or repaired by persons or organizations unauthorized by Embest Technology, or altered in factory specifications, or configured or expanded with the components that are not provided or recognized by Embest Technology and the resulted damage in appearance or function;

F. Product failures caused by the software or system installed by customers or inappropriate settings of software or computer viruses;

G. Products purchased from unauthorized sales;

H. Warranty (including verbal and written) that is not made by Embest Technology and not included in the scope of our warranty should be fulfilled by the party who committed. Embest Technology has no any responsibility;

3） Within the period of warranty, the freight for sending products from customers to Embest Technology should be paid by customers; the freight from Embest to customers should be paid by us. The freight in any direction occurs after warranty period should be paid by customers.

4） Please contact technical support if there is any repair request.

注意：

Embest Technology will not take any responsibility on the products sent back without the permission of the company.

Contact Information

Hotline: +86-755-25635626-872/875

Fax: +86-755-25635626-666

Pre-sales: sales@embedinfo.com

After-sales: support@embedinfo.com

Website: http://www.armkits.com or http://www.embest-tech.com

Address: Tower B 4/F, Shanshui Building, Nanshan Yungu Innovation Industry Park, Liuxian Ave. No. 1183, Taoyuan St., Nanshan District, Shenzhen, China (518055)