

JMY980 Core Board and JMY901W/R Reader English Manual

(Revision 1.00)

Jinmuyu Electronics Co. LTD

2012/7/24



Please read this manual carefully before using. If any problem, please mail to: Jinmuyu@vip.sina.com



Contents

1 Product Introduction.....	2
2 Interface and Address Allocation Instructions.....	2
2.1 Pin Description.....	2
2.2 Address Space Allocation, And Chip Select Signals Defined.....	5
3 Procedural Programming And Systems Download.....	6
3.1 Bootloader Programming.....	6
3.1.1 Programming nor Flash Software Installation.....	6
3.1.2 ARM9 NOR Flash Programming Process.....	8
3.2 Download the Operating System.....	12
3.2.1 Preparatory Work before the Download System.....	12
3.2.2 Download Linux system.....	13
3.2.3 Download WindowsCE System.....	20
4 WindowsCE 6.0 Developer's Guide.....	21
4.1 Building WindowsCE 6.0 Development Environment.....	21
4.1.1 Installing Visual Studio 2005 and patch.....	22
4.1.2 Installing the Windows CE 6.0 and patch.....	30
4.1.3 Installing the BSP and core engineering sample.....	38
4.1.4 The location of each driver source code.....	41
4.2 Compiling and configuring for Windows CE 6.0 kernel and Bootloader.....	42
4.2.1 Compiler default kernel engineering example.....	42
4.2.2 Compiling and programing NBOOT of the Bootloader.....	45
4.2.3 Modifying the LCD type and serial output function in the BSP.....	47
4.2.4 Creating and editing Windows CE for startupping Logo.....	48
4.2.5 Creating SDK.....	50
4.2.6 Installing SDK.....	52
4.3 With PC synchronization.....	55
4.3.1 Installation of synchronous drive and software.....	55
4.4 Create an application, compile and download to development board running through VS2005.....	61
4.4.1 Creating a Project.....	61



1 Product Introduction

JMY980 is a minimal system board, with the basic system configuration:

CPU: Samsung S3C2440, Frequency: 400MHz;

NOR FLASH: 4MByte, Nonvolatile while power off;

NAND FLASH: 256MByte, Nonvolatile while power off;

SDRAM: 64MByte, Composed of two 16bit width of 32MByte SDRAM, the clock frequency up to 100MHz;

System clock source: 12M Passive crystal;

Real-time clock: Internal real time clock (not included then the backup with a battery);

System power supply: +5V;

Support system: Linux 2.6.32/Windows CE 6.0

UCos2/2440test (Bare metal test program);

Dimension: 63 × 52mm;

56Pin 2.0mm Spacing; GPIO interface PA;

50Pin 2.0mm Spacing; LCD; CMOS CAMERA interface PB;

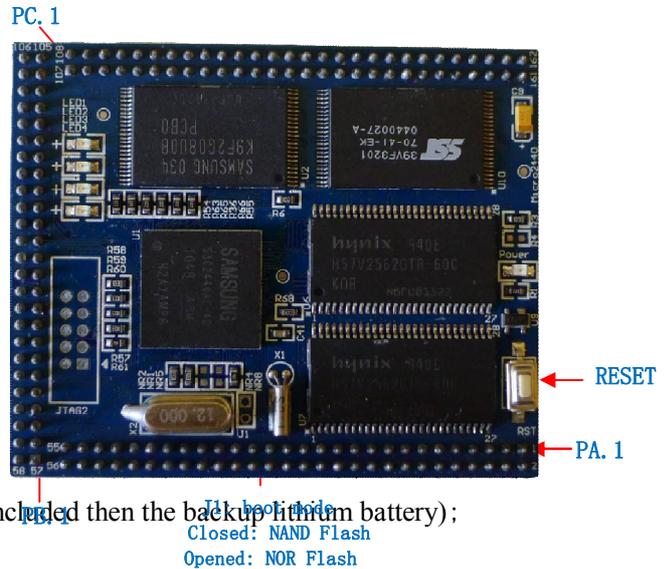
56Pin 2.0mm Spacing; system bus PC;

10Pin 2.0mm Spacing; JTAG interface;

A power indicator and four user LEDs;

Board the JTAG, professional voltage regulator chip, users connected to a 5V power supply can be developed to do simple debugging.

Note: Information on CD-ROM provided in the core board schematics and PCB packages, development tools and Samsung original data.



2 Interface and Address Allocation Instructions

2.1 Pin Description

Interface PA	Network Name	Directions (Some ports can be multiplexed)	Interface PA	Network Name	Directions (Some ports can be multiplexed)
PA1	VDD5V	5V power	PA2	GND	GND
PA3	EINT19	EINT19/GPG11	PA4	EINT18	EINT18/GPG10/nCTS1



PA5	EINT17	EINT17/GPG9/nRST1	PA6	EINT16	EINT16/GPG8
PA7	EINT15	EINT15/GPG7/SPICLK1	PA8	EINT14	EINT14/GPG6/SPIMOSI1
PA9	EINT13	EINT13/GPG5/SPIMISO1	PA10	EINT11	EINT11/GPG3/nSS1
PA11	EINT8	EINT8/GPG0	PA12	EINT6	EINT6/GPF6
PA13	EINT5	EINT5/GPF5	PA14	EINT4	EINT4/GPF4
PA15	EINT3	EINT3/GPF3	PA16	EINT2	EINT2/GPF2
PA17	EINT1	EINT1/GPF1	PA18	EINT0	EINT0/GPF0
PA19	WP_SD	WP_SD/GPH8	PA20	SDCLK	SDCLK/GPE5
PA21	SDCMD	SDCMD/GPE6	PA22	SDDATA2	SDDATA2/GPE9
PA23	SDDATA3	SDDATA3/GPE10	PA24	SDDATA0	SDDATA0/GPE7
PA25	SDDATA1	SDDATA1/GPE8	PA26	LCDVF2	OM0 (NOR-NAND Select)
PA27	LCDVF0	LCDVF0/GPC5, Used for USB_EN	PA28	M_nRESET	Manual reset signal (Active-low)
PA29	DN1	DN1/PDN0, USB Slave' s D-	PA30	DP1	DP1/PDP0, USB Slave' s D+
PA31	DN0	DN0, USB Host' s D-	PA32	DP0	DP0, USB Host' s D+
PA33	AIN2	AIN2	PA34	VDDRTC	RTC power input (1.8V)
PA35	AIN0	AIN0	PA36	AIN1	AIN1
PA37	L3MODE	L3MODE/TOUT2/GPB2	PA38	L3DATA	L3DATA/TOUT3/GPB3
PA39	L3CLOCK	L3LOCK/TCLK0/GPB4	PA40	I2SLRCK	I2SLRCK/GPE0
PA41	I2SSCLK	I2SSCLK/GPE1	PA42	CDCLK	CDCLK/GPE2
PA43	I2SSDI	I2SSDI/GPE3	PA44	I2SSD0	I2SSD0/GPE4
PA45	GPB0	TOUT0/GPB0	PA46	GPB1	TOUT1/GPB1
PA47	TXD2	TXD2/nRTS1/GPH6	PA48	RXD2	RXD2/nCTS1/GPH7
PA49	TXD1	TXD1/GPH4	PA50	RXD1	RXD1/GPH5
PA51	TXD0	TXD0/GPH2	PA52	RXD0	RXD0/GPH3
PA53	nCTS0	nCTS0/GPH0	PA54	nRTS0	nRTS0/GPH1
PA55	I2CSDA	I2CSDA/GPE15	PA56	I2CSCL	I2CSCL/GPE14

Interface PB	Network Name	Directions (Some ports can be multiplexed)	Interface PB	Network Name	Directions (Some ports can be multiplexed)
PB1	TSYM		PB2	TSYP	
PB3	TSXM		PB4	TSYM	
PB5	VD22	VD22/GPD14	PB6	VD23	VD23/GPD15
PB7	VD20	VD20/GPD12	PB8	VD21	VD21/GPD13
PB9	VD18	VD18/GPD10	PB10	VD19	VD19/GPD11
PB11	VD16	VD16/GPD8	PB12	VD17	VD17/GPD9
PB13	VD14	VD14/GPD6	PB14	VD15	VD15/GPD7
PB15	VD12	VD12/GPD4	PB16	VD13	VD13/GPD5
PB17	VD10	VD10/GPD2	PB18	VD11	VD11/GPD3
PB19	VD8	VD8/GPD0	PB20	VD9	VD9/GPD1
PB21	VD6	VD6/GPC14	PB22	VD7	VD7/GPC15
PB23	VD4	VD4/GPC12	PB24	VD5	VD5/GPC13



PB25	VD2	VD2/GPC10	PB26	VD3	VD3/GPC11
PB27	VD0	VD0/GPC8	PB28	VD1	VD1/GPC9
PB29	LCD_PWR	LCD_PWR/EINT12/GPG4	PB30	VM	VM/VDEN/GPC4
PB31	VFRAME	VFRAME/VSYNC/GPC3	PB32	VLINE	VLINE/HSYNC/GPC2
PB33	VCLK	VCLK/GPC1	PB34	LEND	LEND/GPC0
PB35	CAMDATA7	CAMDATA7/GPJ7	PB36	CAMDATA6	CAMDATA6/GPJ6
PB37	CAMDATA5	CAMDATA5/GPJ5	PB38	CAMDATA4	CAMDATA4/GPJ4
PB39	CAMDATA3	CAMDATA3/GPJ3	PB40	CAMDATA2	CAMDATA2/GPJ2
PB41	CAMDATA1	CAMDATA1/GPJ1	PB42	CAMDATA0	CAMDATA0/GPJ0
PB43	CAMCLK	CAMCLK/GPJ11	PB44	CAM_PCLK	CAM_PCLK/GPJ8
PB45	CAM_VSYNC	CAM_VSYNC/GPJ9	PB46	CAM_HREF	CAM_HREF/GPJ10
PB47	EINT20	EINT20/GPG12	PB48	CAMRST	CAMRESET/GPJ12
PB49	VDD5V	VDD5V	PB50	GND	GND

Interface PC	Network Name	Directions (Some ports can be multiplexed)	Interface PC	Network Name	Directions (Some ports can be multiplexed)
PC1	EINT7	EINT7/GPF7	PC2	EINT9	EINT9/GPG1
PC3	LnGCS1	Chip Select LnGCS1	PC4	LnGCS3	Chip Select LnGCS3
PC5	LnGCS2	Chip Select LnGCS2	PC6	LnWBE1	LnWBE1
PC7	LnGCS4	Chip Select LnGCS4	PC8	LnWE	LnWE
PC9	LnOE	LnOE	PC10	nRESET	nRESET
PC11	nWAIT	nWAIT	PC12	nXDACK0	nXDACK0
PC13	LADDR0	LADDR 0	PC14	nXDREQ0	nXDREQ0
PC15	LADDR1	LADDR 1	PC16	LADDR2	LADDR 2
PC17	LADDR3	LADDR 3	PC18	LADDR4	LADDR 4
PC19	LADDR5	LADDR 5	PC20	LADDR6	LADDR 6
PC21	LADDR7	LADDR 7	PC22	LADDR8	LADDR 8
PC23	LADDR9	LADDR 9	PC24	LADDR10	LADDR 10
PC25	LADDR11	LADDR 11	PC26	LADDR12	LADDR 12
PC27	LADDR13	LADDR 13	PC28	LADDR14	LADDR 14
PC29	LADDR15	LADDR 15	PC30	LADDR16	LADDR 16
PC31	LADDR17	LADDR 17	PC32	LADDR18	LADDR 18
PC33	LADDR19	LADDR 19	PC34	LADDR20	LADDR 20
PC35	LADDR21	LADDR 21	PC36	LADDR22	LADDR 22
PC37	LADDR23	LADDR 23	PC38	LADDR24	LADDR 24
PC39	LDATA0	LDATA 0	PC40	LDATA1	LDATA 1
PC41	LDATA2	LDATA 2	PC42	LDATA3	LDATA 3
PC43	LDATA4	LDATA 4	PC44	LDATA5	LDATA 5
PC45	LDATA6	LDATA 6	PC46	LDATA7	LDATA 7
PC47	LDATA8	LDATA 8	PC48	LDATA9	LDATA 9
PC49	LDATA10	LDATA 10	PC50	LDATA11	LDATA 11
PC51	LDATA12	LDATA 12	PC52	LDATA13	LDATA 13
PC53	LDATA14	LDATA 14	PC54	LDATA15	LDATA 15

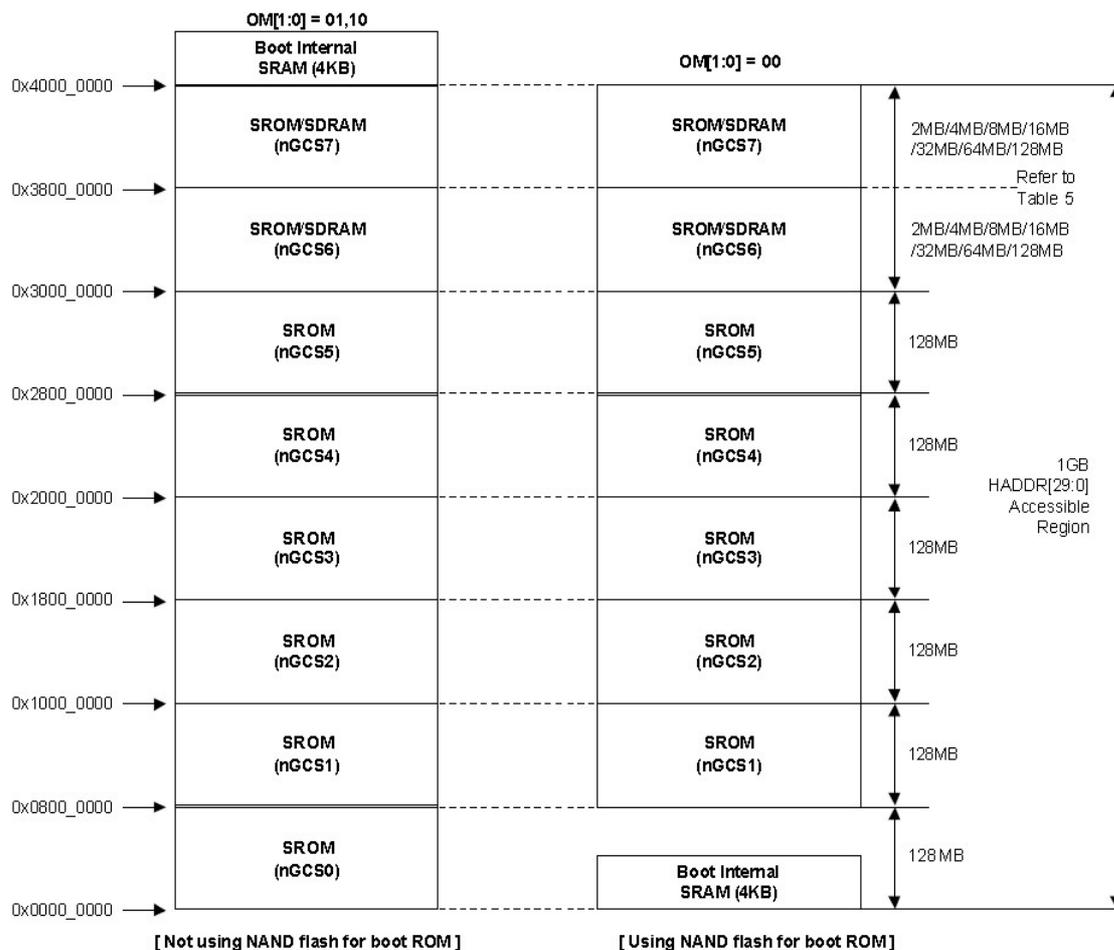


PC55	VDD5V	POWER 5V	PC56	GND	GND
------	-------	----------	------	-----	-----

2.2 Address Space Allocation, And Chip Select Signals Defined

S3C2440 supports two startup modes: Nand Flash and Nor Flash

Both start modes, the storage space allocation for each chip select is different, as shown below:



In the above picture,

Nor Flash nGCS0 piece of selected startup mode of the memory allocation map on the left;

The right is the Nand Flash startup mode of the memory allocation map;

The following defines the device address space allocation, and chip select;

Before conducting device address, one thing is important to note that the device nGCS0 chip select space in the start mode, the map is not the same. You can know from the above Fig.;

NAND Flash boot mode, the internal 4K Bytes Boot SRAM is mapped to nGCS0 chip select space;

Nor Flash boot mode (non-Nand Flash startup mode), connected with nGCS0, Nor Flash, external memory is mapped to nGCS0 chip select space;

SDRAM Address space: 0x30000000 ~ 0x34000000.



3 Procedural Programming And Systems Download

3.1 Bootloader Programming

For the new Board, there is no any program; we need program the first program via JTAG interface that is Supervivi. With the Supperviv, we can download more complex system program via USB interface.

3.1.1 Programming nor Flash Software Installation

H-JTAG software installation requirements: Computer must have a parallel port. (The software is installed only in the first case, if installed, this step is omitted).

1、Installation H-JTAG

For H-JTAG, the installation file is located in the CD-ROM “JMY980TOOLS\H-JTAG” catalog, double-click to run, and can be installed in accordance with its prompt.

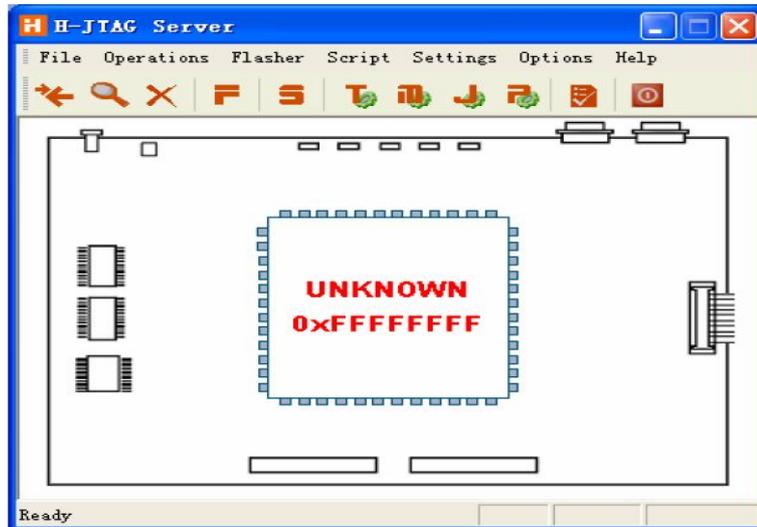


After installing, the generated H-JTAG and H-Flasher shortcut will show on the desktop; double-click to run the H-JTAG, the program will automatically detect whether to connect the JTAG equipment, because we have not done any of the settings, it will pop up a prompt window:



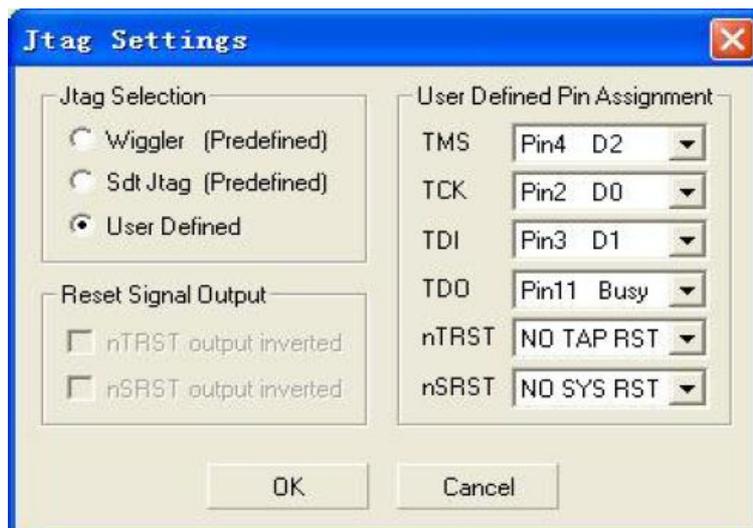


To click OK to enter the main program, not connected to any target device, so displaying as following:



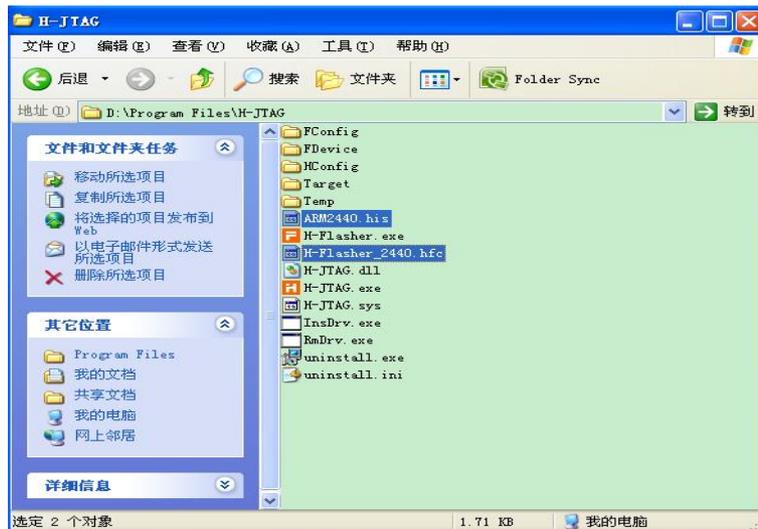
2、 JTAG interface set

In the H-JTAG interface menu to hit Setting-> Jtag Setting, doing as shown below to set up,click OK to return to the main interface.

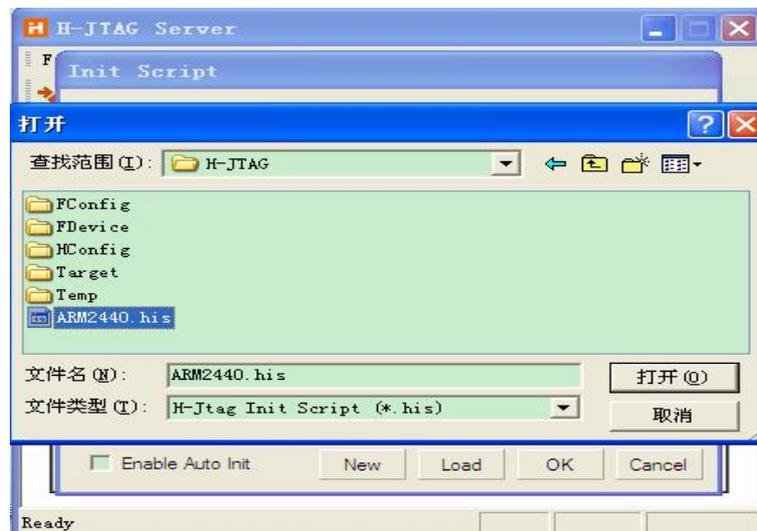


3、 Set the init script

To copy H-JTAG” ARM2440.his and H-Flasher_2440.hfc file in the directory “JMY980TOOLS \ H-JTAG” for the CD-ROM to the installation directory of the H-JTAG, as shown:



In the main interface of the H-JTAG, to hit Script->Init Script, then will Pop-up Init Script window, to hit the Load button in the window below, to locate and select the open just copied ARM2440.his file, as shown below:

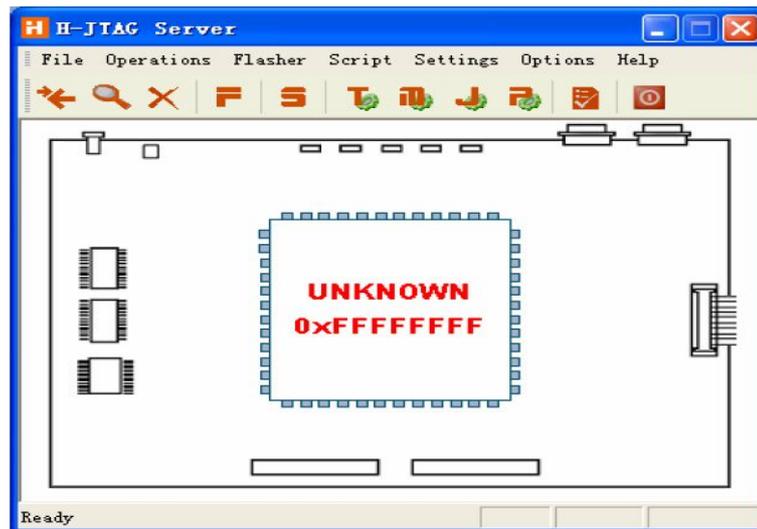


At this time, the Init Script window will be loaded into the script fill, as shown, be careful not to click the “Enable Auto Init”, click OK to return the H-JTAG interface.

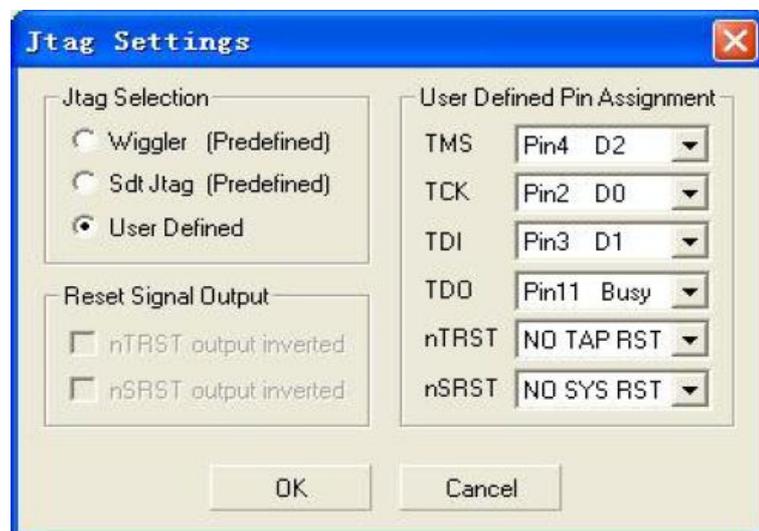
3.1.2 ARM9 NOR Flash Programming Process

- 1、 Check the programming tools
 - (1) To prepare a Computer with a parallel port, and installed the software of the H-JTAG.
 - (2) One of parallel lines for NOR Flash.
 - (3) JMY901 contact less RF reader board or your company's own development board.
- 2、 Configuration of H-JTAG software:

Opening the software as following:



In the H-JTAG interface menu to hit Setting->Jtag Settings, to make the following diagram configuration:

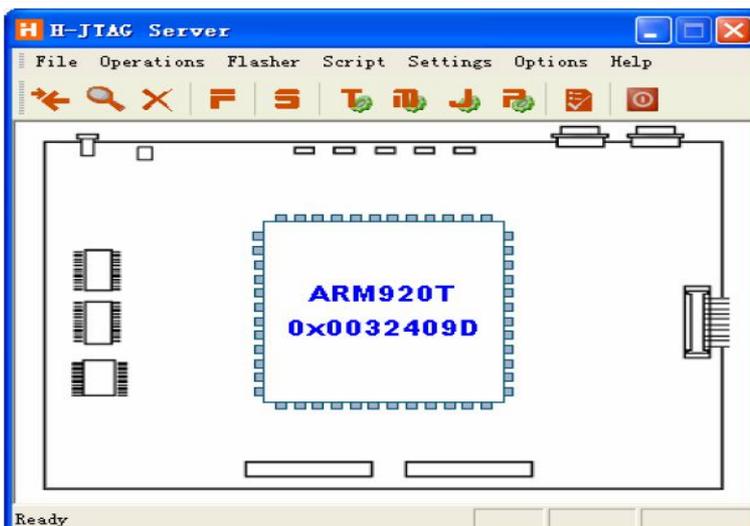


3、 To connect the device

- (1) Using the +5V supply line, to connect the core board but without powering.
- (2) Using the parallel port cable to connect the core board and computer.
- (3) Confirm JMY901 toggle switch S2 in the NOR side
- (4) Turn on the power.

4、 Check the device connection is normal or not

To hit Operations->Detect Target, if the showing as below, that is meaning the interface has been connected:

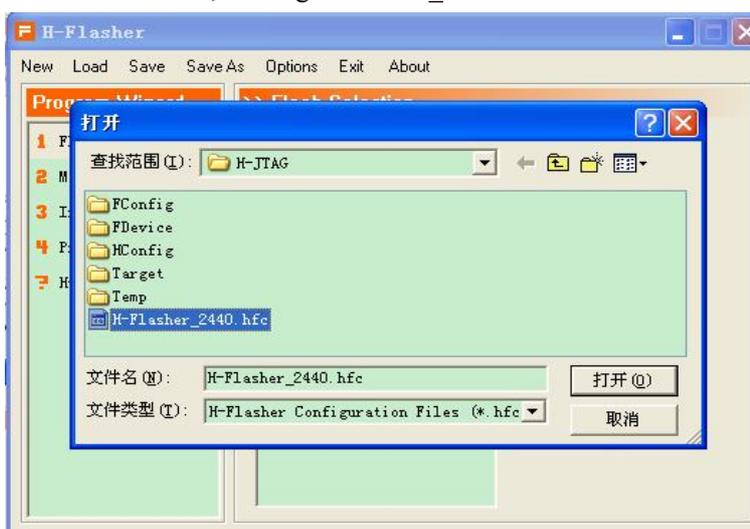


5、 To load H-Flasher_2440.hfc

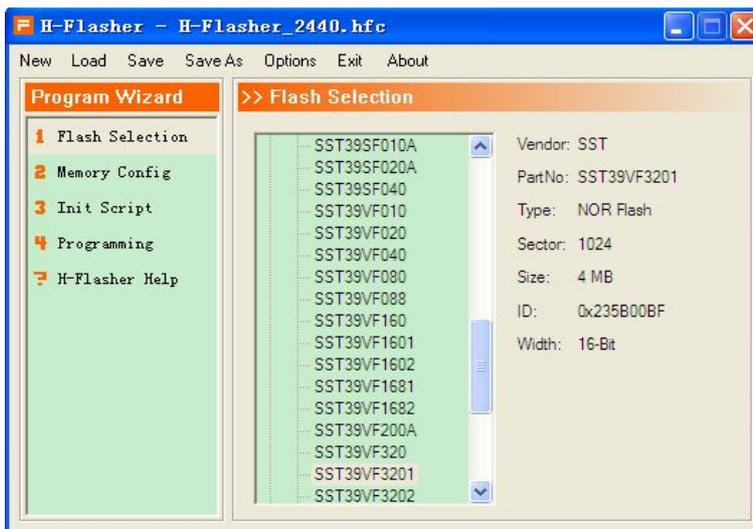
Hitting Flasher->Start H-Flasher, to show the following H-Flash interface:



To hit Load in the H-Flash interface, loading H-Flasher_2440.hfc:

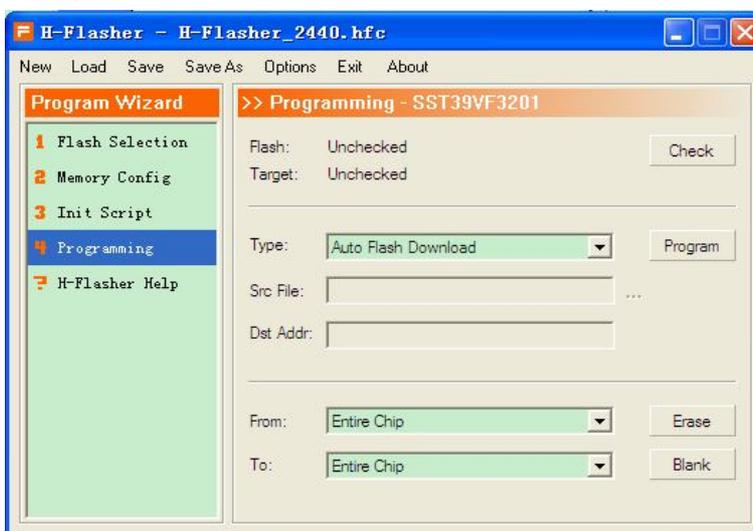


After loaded, the following interface will appear, to choose SST39VF3201:

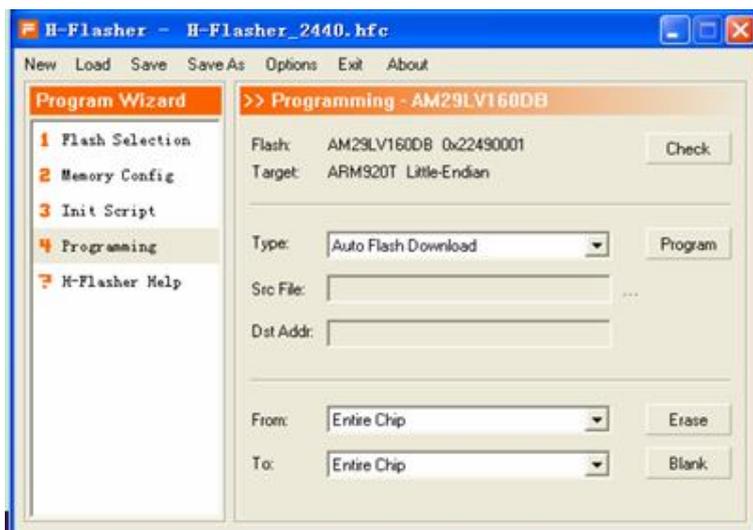


6、 Set programming parameters

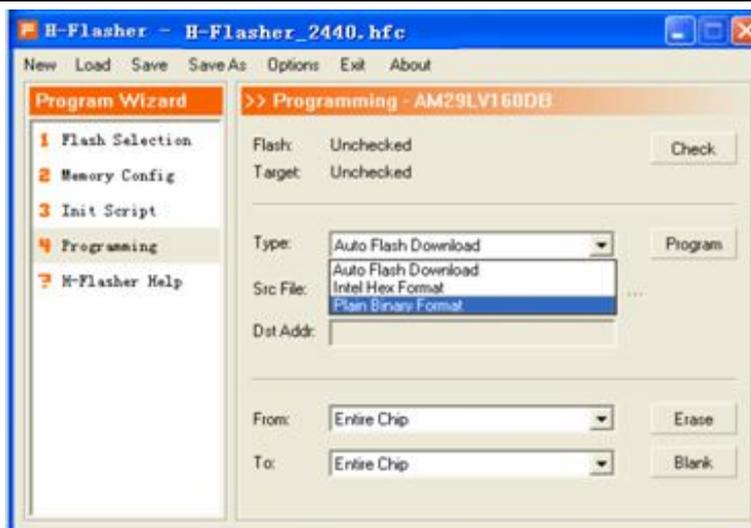
(1) Hitting 4 Programming



(2) To hit Check button, if the core board is ok, it displays the following interface:



(3) Hitting Type drop-down list, to choose "Plain Binary Format":

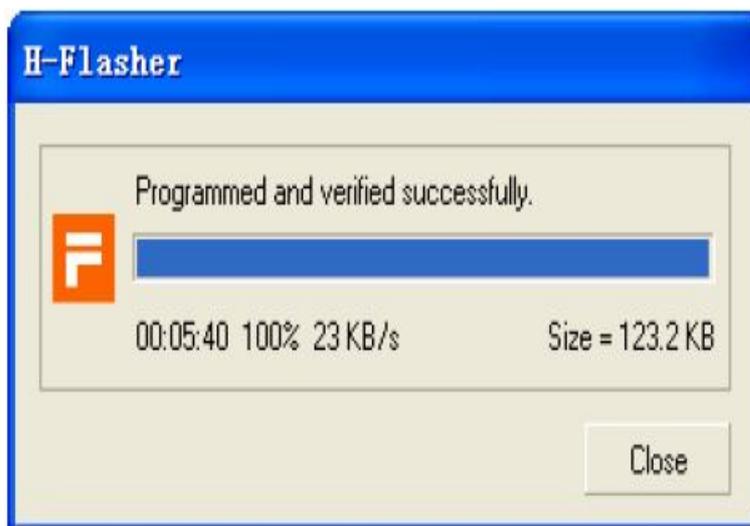


(4) And then to hit the Browse button to the right side of the Src File..., Select the file(supervivi-128M) to be programmed.

(5) Input 0 in the Dst Addr column.

7、 Programming

To hit Program, If the programming is successful, it will be shown as following:



3.2 Download the Operating System

3.2.1 Preparatory Work before the Download System

1、 Check the programming tools

(1) To prepare a Computer with USB and Serial ports

(2) Each one for USB and Serial cable

(3) A JMY980 Core board.

(4) JMY901 contact less RF reader board or your company's own development board.

(5) DNW software and 115200.ht HyperTerminal software (for this two software no need to install,



directly copy to the hard disk to run) .

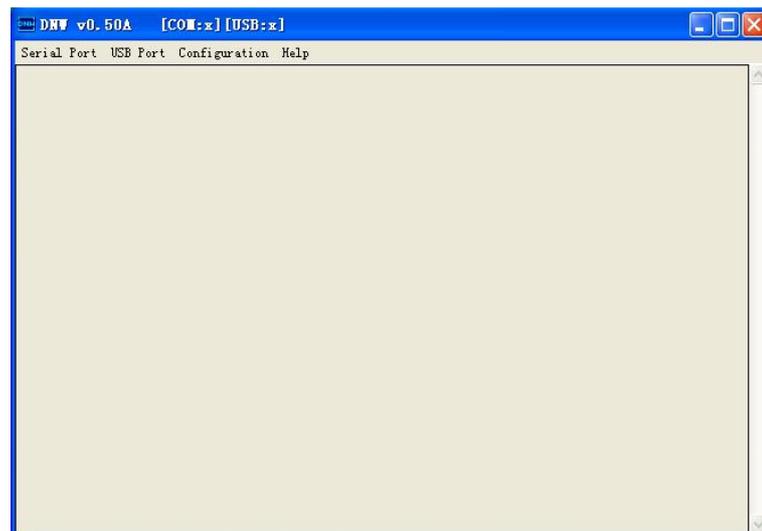
- (6) Installation USB driver (JMY980TOOLS\usb driver) .
- (7) To switch S2 of JMY901 in NOR-side(NOR Flash staring mode)

2、Opening software

- (1) To open 115200.ht HyperTerminal software, as following:

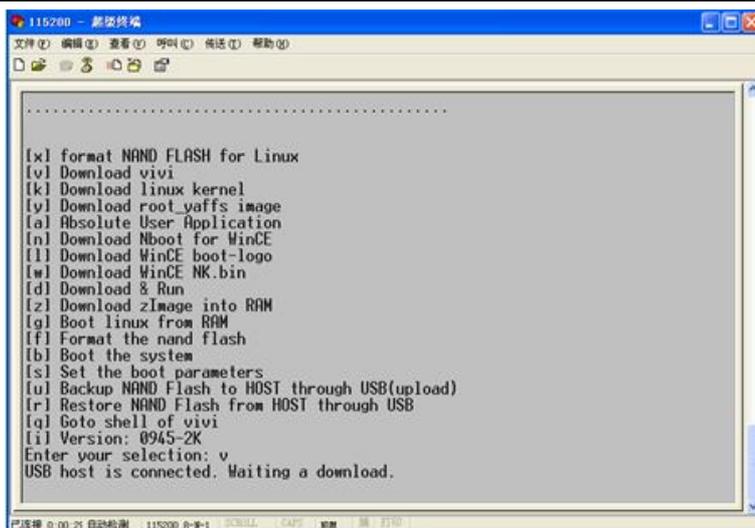


- (2) To open DNW software, as following:

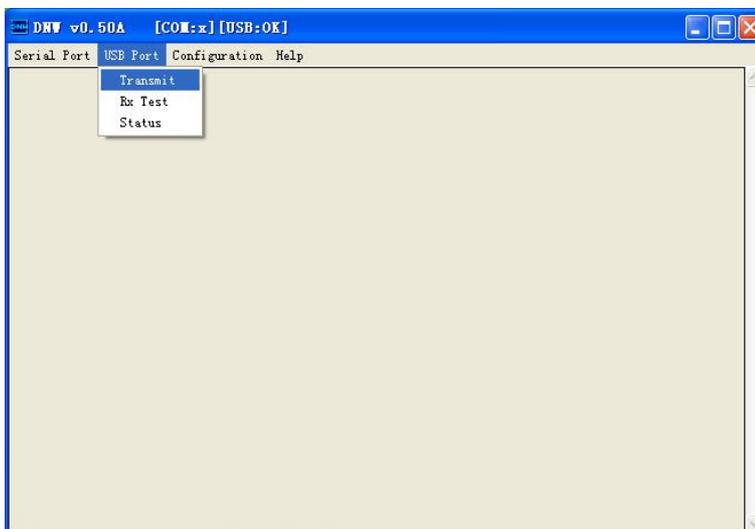


3.2.2Download Linux system

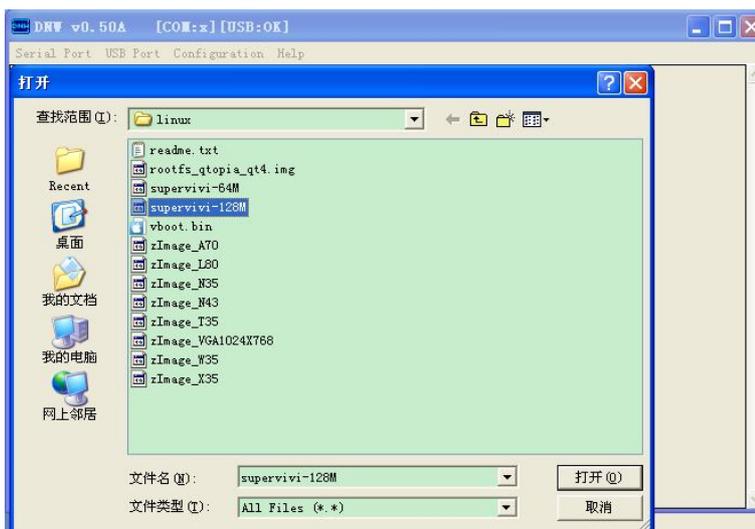
- 1、 To confirm the Serial ports and USB Slave interface are connected, after power on, the 115200.sh software will be showing as the following:



To hit “USB Port->Transmit->Transmit” in the DNW software, as following:

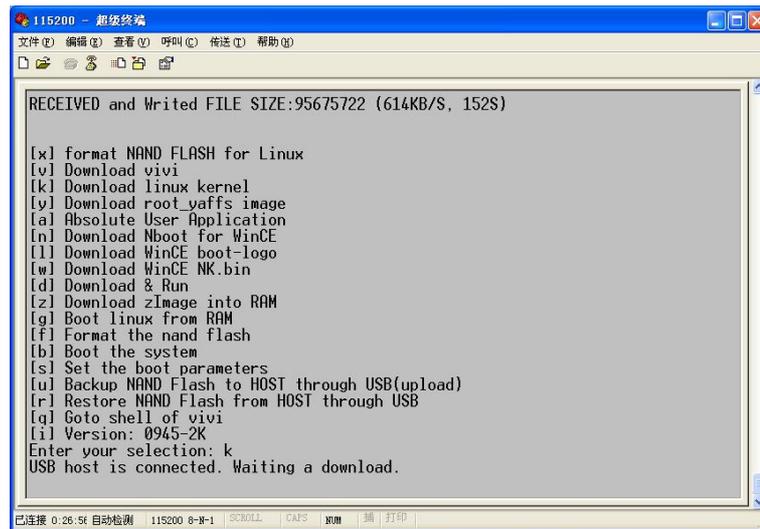


To select supervivi-128M, then hit open, as following:

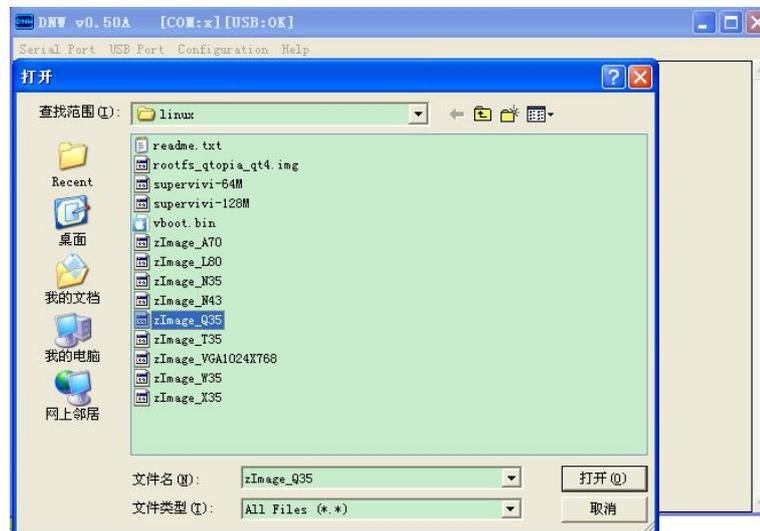


5、Installation Linux Kernel

To select the function key [k], as following:

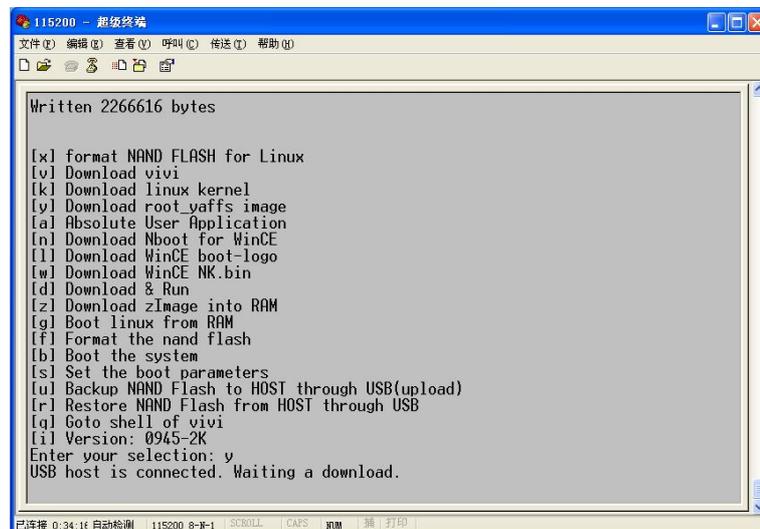


To hit “USB Port->Transmit->Transmit” in the DNW software, to select zImage_Q35, as following:



6、Installation the root file system

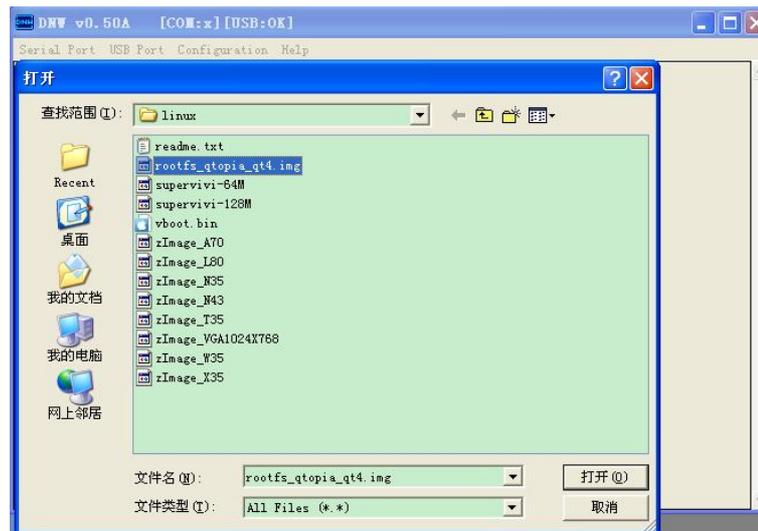
To select the function key [y], as following:



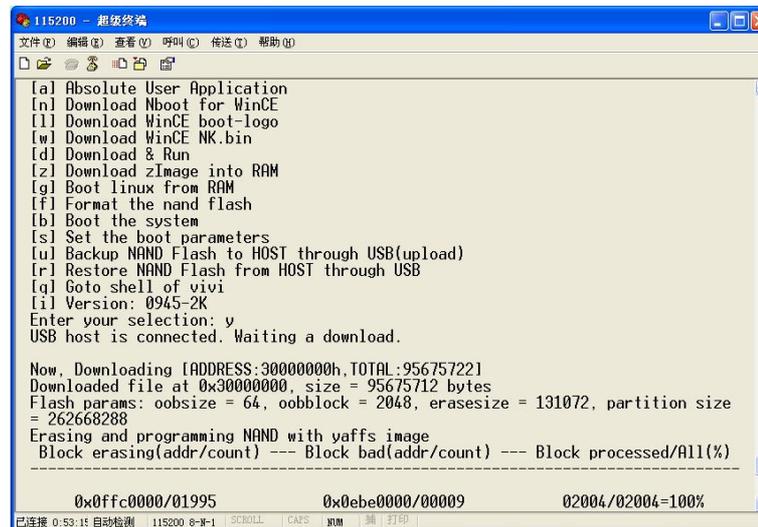
To hit “USB Port->Transmit->Transmit” in the DNW software, to select rootfs_qtopia_qt4.img,



as following:



To start sending the file system, a little longer, please wait for the process of transmission as shown below:



After the transmission, it will be showing Load ysffs OK.

7、Enter into Linux system

After power off, switch S2 of JMY901 to NAND side, and last to re-power on, the system will be started from the NAND Flash, as shown below:



```
115200 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
yaffs: dev is 32505859 name is "mtdblock3"
yaffs: passed flags ....
yaffs: Attempting MTD mount on 31.3, "mtdblock3"
yaffs: auto selecting yaffs2
block 783 is bad
block 943 is bad
block 1090 is bad
block 1110 is bad
block 1111 is bad
block 1136 is bad
block 1171 is bad
block 1189 is bad
block 1845 is bad
yaffs_read_super: isCheckpointed 0
VFS: Mounted root (yaffs filesystem) on device 31:3.
Freeing init memory: 156K
[29/Nov/1999:16:00:01 +0000] boa: server version Boa/0.94.13
[29/Nov/1999:16:00:01 +0000] boa: server built Jul 26 2010 at 15:58:29.
[29/Nov/1999:16:00:01 +0000] boa: starting server pid=679, port 80

Try to bring eth0 interface up.....eth0: link down
Done

Please press Enter to activate this console. _
```

Press Enter to enter the Linux file system to operate, as shown below:

```
115200 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
block 943 is bad
block 1090 is bad
block 1110 is bad
block 1111 is bad
block 1136 is bad
block 1171 is bad
block 1189 is bad
block 1845 is bad
yaffs_read_super: isCheckpointed 0
VFS: Mounted root (yaffs filesystem) on device 31:3.
Freeing init memory: 156K
[29/Nov/1999:16:00:01 +0000] boa: server version Boa/0.94.13
[29/Nov/1999:16:00:01 +0000] boa: server built Jul 26 2010 at 15:58:29.
[29/Nov/1999:16:00:01 +0000] boa: starting server pid=679, port 80

Try to bring eth0 interface up.....eth0: link down
Done

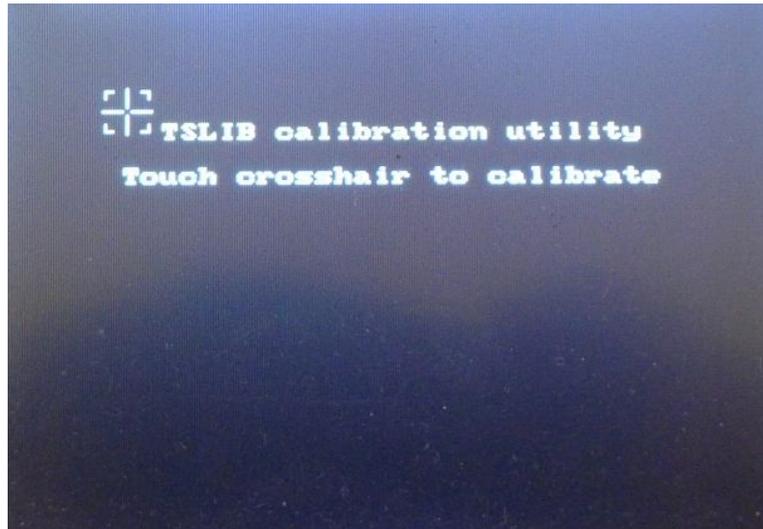
Please press Enter to activate this console.
[root@FriendlyARM /]# ls
bin          home         lost+found  proc        sys         var
dev          lib          mnt        root        tmp         www
etc          linuxrc     opt         sbin       usr
[root@FriendlyARM /]#
```

8、 The operation with touch screen

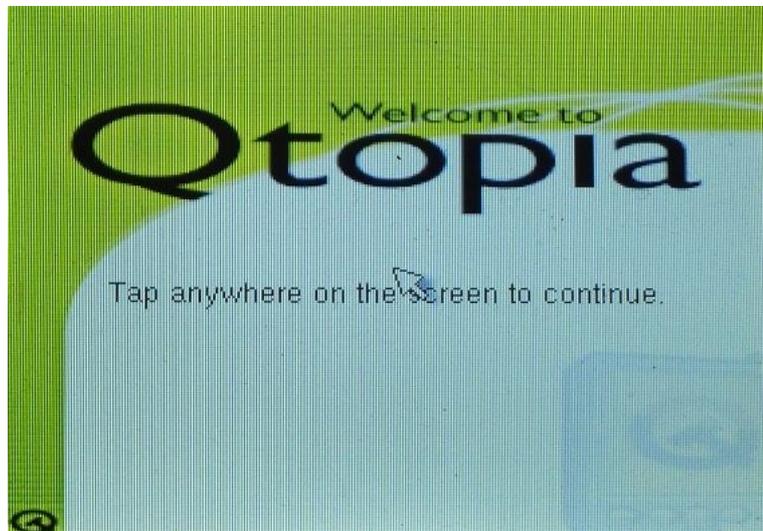
If you connect the touch screen, you can see the following Linux boot screen:



Touch calibration, click on the cross cursor to calibrate the screen, as shown below:



After calibration, then enter into Qtopia interface, as following:



Click on the screen into the system interface, as shown below:



Linux system has been installed!



3.2.3 Download WindowsCE System

Download WindowsCE System is the same to Download Linux System. The difference is function selection and programming file, function selected programming file storage in “JMY980TOOLS\images\wince6.0” catalog.

Programming steps:

- 1、 To select the function key [n], to program file nboot_Q35.bin;
- 2、 To select the function key [l], to program file bootlogo.bmp;
- 3、 To select the function key [w], to program file NK_Q35.bin;
- 4、 To install WinCE and the synchronization software ActiveSync for WindowsXP, then to storage in “JMY980TOOLS\windows platform tools\ActiveSync” catalog;

After programming the system then to switch to the NAND Flash start, touch screen will appear the following interface:



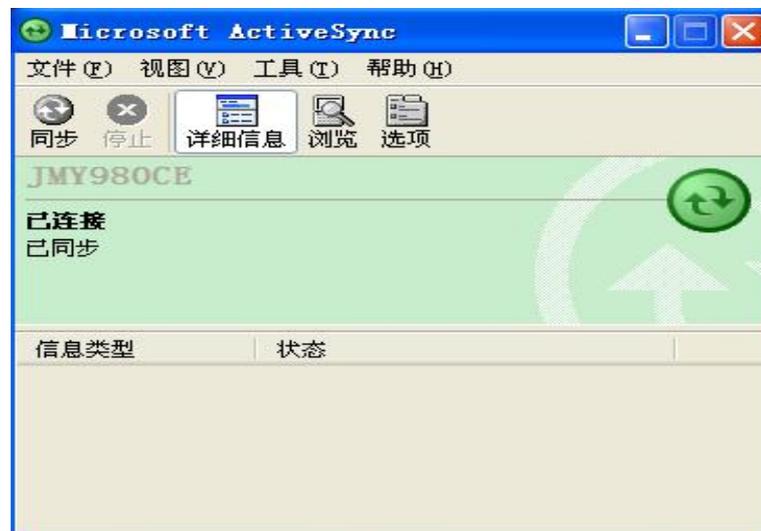
To start WinCE interface, as following:



Enter into the WinCE operation system, as following:



Synchronous software-ActiveSync will pop up, as following:



WinCE system has been installed!

4 WindowsCE 6.0 Developer's Guide

4.1 Building WindowsCE 6.0 Development Environment

Note: The following software and steps are based on Microsoft Windows XP SP3 system and other Windows system without testing.

Windows CE 6.0 installation process is very cumbersome, and for the development of host is relatively high (otherwise it will be very slow), we recommend that users, especially beginners should follow the steps described to install the development environment.

Here is the configuration of the development host, for reference only:

CPU: Pentium(R) Dual-Core E6700 @3.20GHZ

Memory: DDR2 4GB

Hard disk: 500GB



Install the required software listed below (partial):

Visual Studio 2005(Does not provide)

Download:

http://download.microsoft.com/download/e/1/4/e1405d9e-47e3-404c-8b09-489437b27fb0/En_vs_2005_Pro_90_Trial.img

Visual Studio 2005 Service Pack 1(File name:VS80sp1-KB926601-X86-ENU.exe)

Download:

<http://www.microsoft.com/en-us/download/details.aspx?id=5553>

Visual Studio 2005 Service Pack 1 Update for Windows Visat
(File name:VS80sp1-KB932232-X86-ENU.exe)

Download:

<http://www.microsoft.com/en-us/download/details.aspx?id=7524>

Visual Studio 2005 Service Pack 1 ATL Security Update
(File name:VS80sp1-KB971090-X86-INTL.exe)

Download:

<http://www.microsoft.com/en-us/download/details.aspx?id=25287>

Windows Embedded CE 6.0

Download:

<http://www.microsoft.com/en-us/download/details.aspx?id=20083>

Windows Embedded CE 6.0 Platform Builder Service Pack 1

Download:

<http://www.microsoft.com/en-us/download/details.aspx?id=4097>

Windows Embedded CE 6.0 R2

Download:

<http://www.microsoft.com/en-us/download/details.aspx?id=18111>

Windows Embedded CE 6.0 R3

Download:

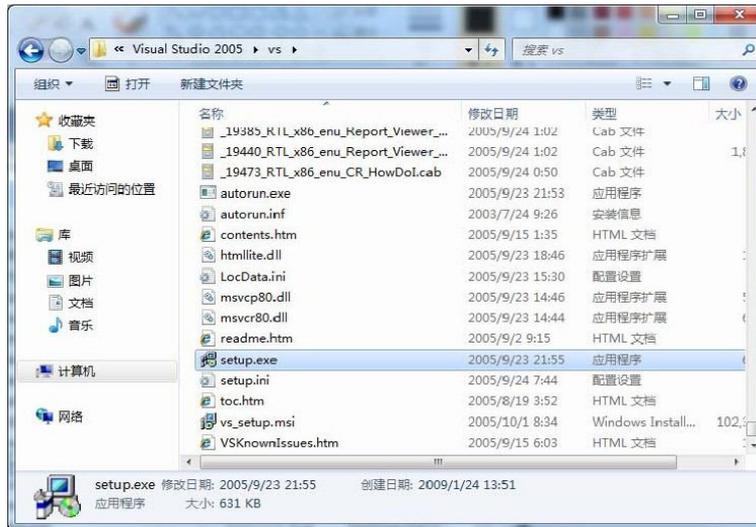
<http://www.microsoft.com/downloads/details.aspx?familyid=bc247d88-ddb6-4d4a-a595-8eee3556fe46&displaylang=ja&displaylang=en>

The order of the list above also shows that the software installation sequence: first install Visual Studio 2005 and the patch, and then install the Windows CE 6.0 and patch.

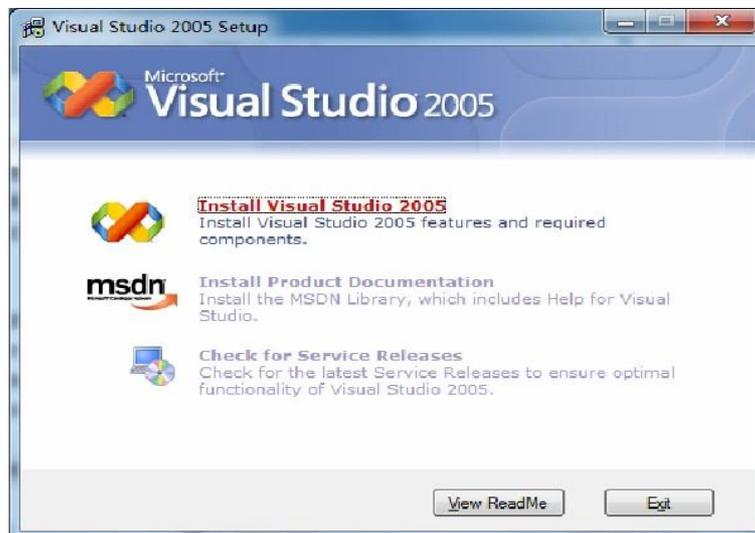
Note: The Platform Builder for Windows CE 6.0 is different with the previous Windows CE 5.0/4.2, etc. It is not an independent software development platform, but as of VS2005 plug-in installed, you must first install the VS2005, and after all kernel configuration compiler and development are based on VS2005.

4.1.1 Installing Visual Studio 2005 and patch

Step1: Opening Visual Studio 2005 folder, find the setup.exe, double-click to start the installation.



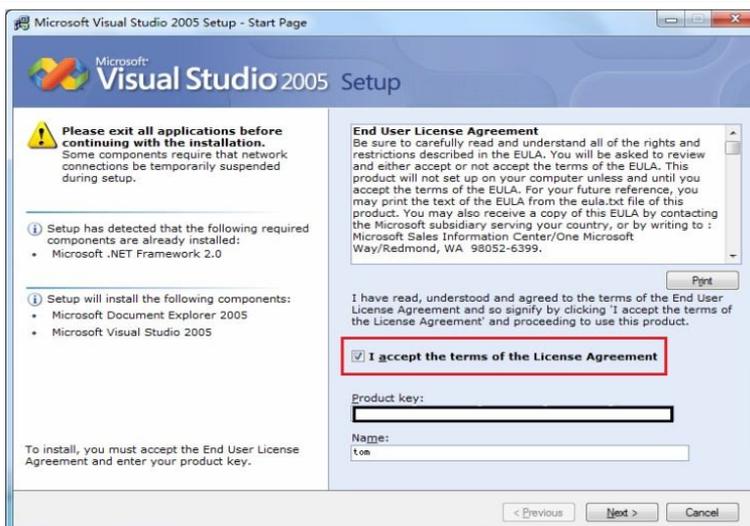
Step2: Appear in Figure interface, point the “Install the Visual Studio 2005”



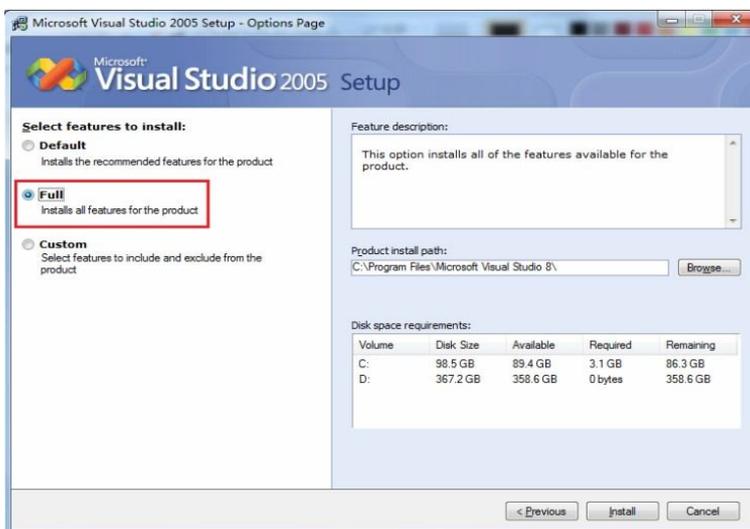
Step3: Figure interface, wait a moment, point “Next”



Step4: Figure interface, pay attention to click on the red box, and enter the serial number, point “Next”



Step5: Figure interface, select the type of installation “Full”, click “Next”



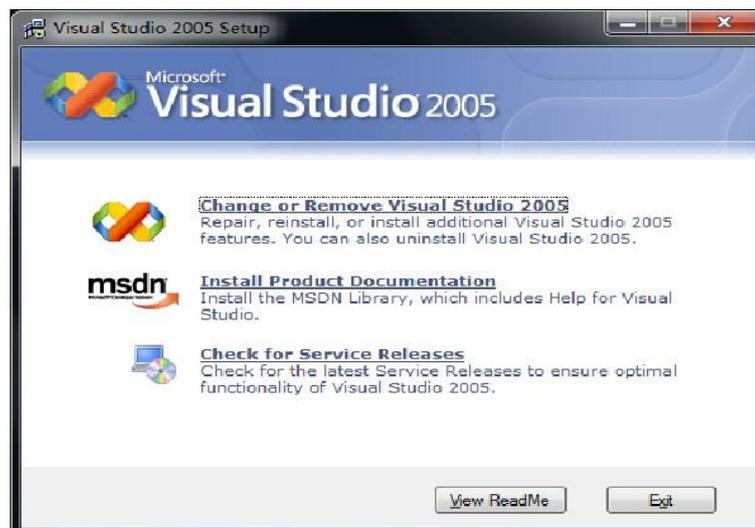
Step6: Figure interface, and started to install Visual Studio 2005, this long process, please be patient.



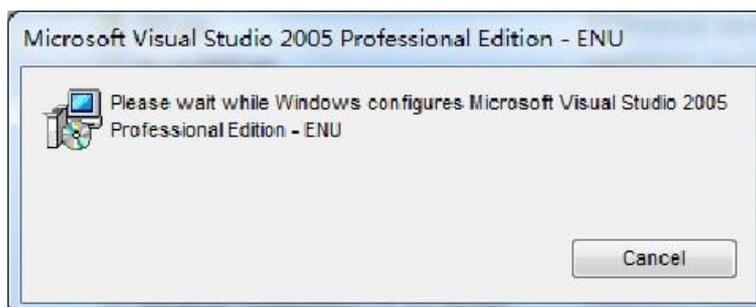
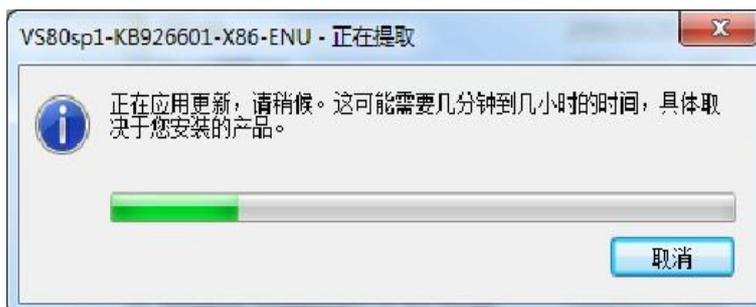
Step7: Visual Studio 2005 installed, the following screen appears, point the “Finish” to end the installation.



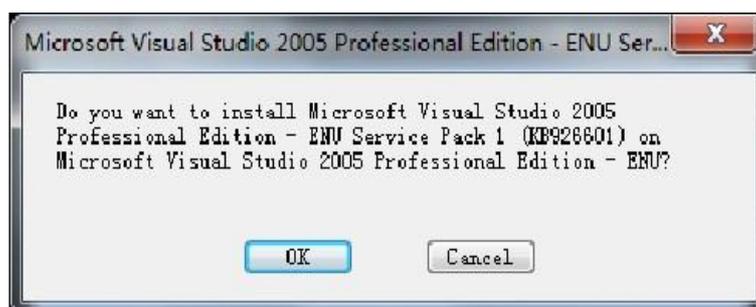
Figure interface, just click “Exit”.



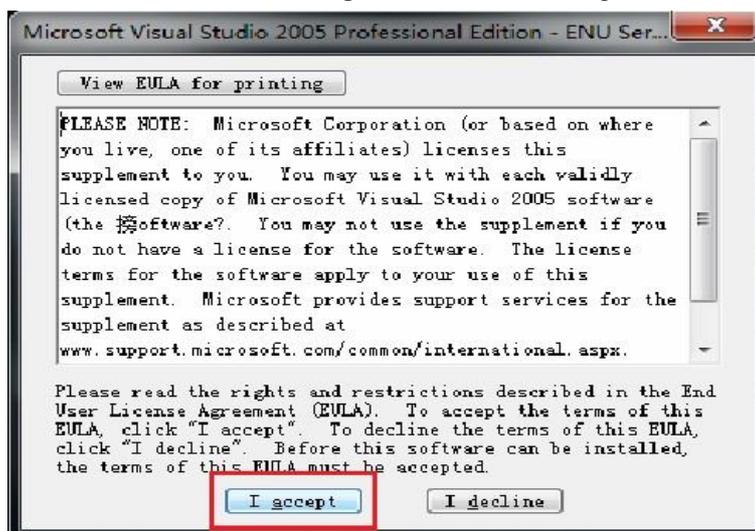
Step8: Now begin to install the first patch file “ the Visual Studio 2005 Service Pack 1”, double-click to run “VS80sp1-KB926601-X86-ENU.exe” to start the installation, appears in Figure interface



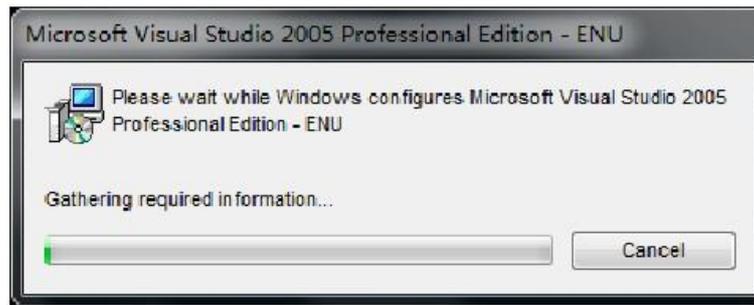
Step9: Have to wait a moment, appears in Figure picture, click “OK” officially installed



Step10: Accept the installation of the license agreement, click “I accept” to continue



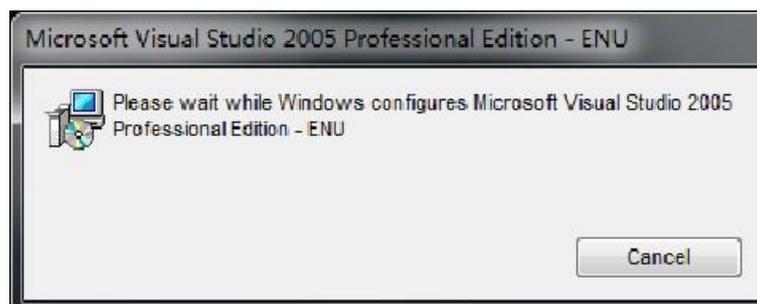
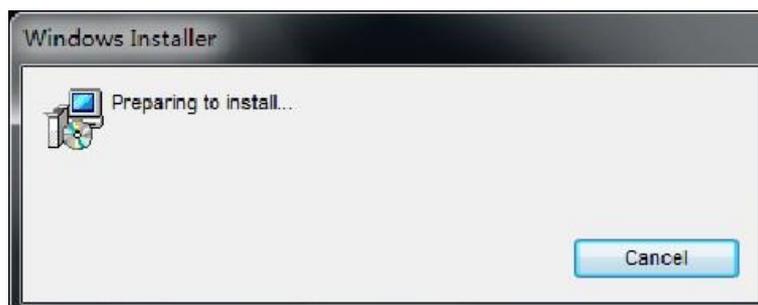
Step11: Appear during the installation interface, the longer this process, please be patient



Step12: Installed, the end of the following interface appears, click “OK” The installation of this patch



Step13: install the second patch “Visual Studio 2005 Service Pack 1 Update for Windows Vista”, double-click “ VS80sp1-Kb932232-x86-ENU.exe “,in turn appear as shown in Figure interface

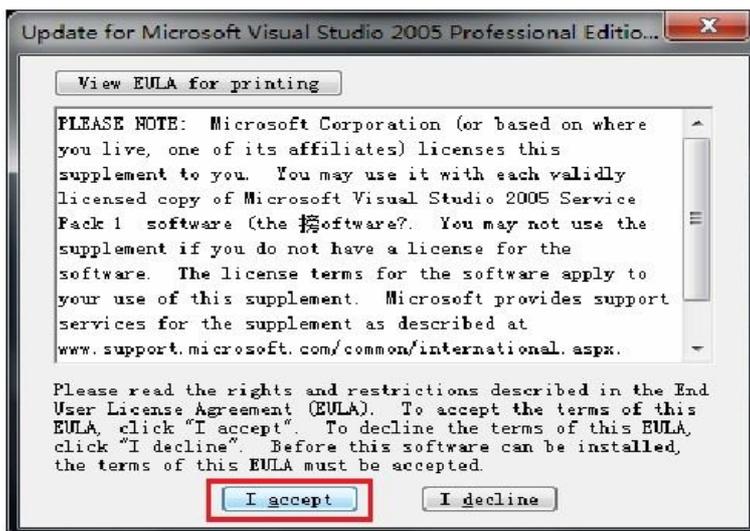


Step14: Wait a moment, appears in Figure interface, click “OK” to continue

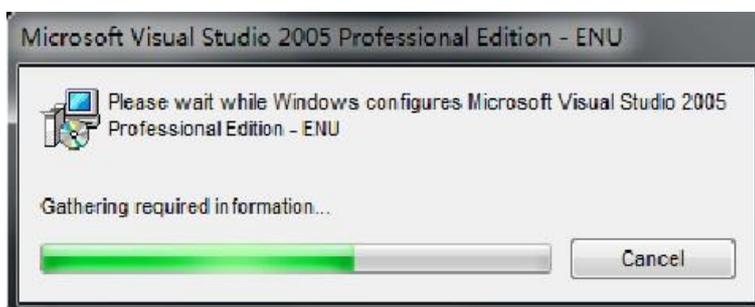




Step15: The installation license agreement screen, click “I accept” to continue



Step16: Appear during the installation interface, the longer this process, please be patient

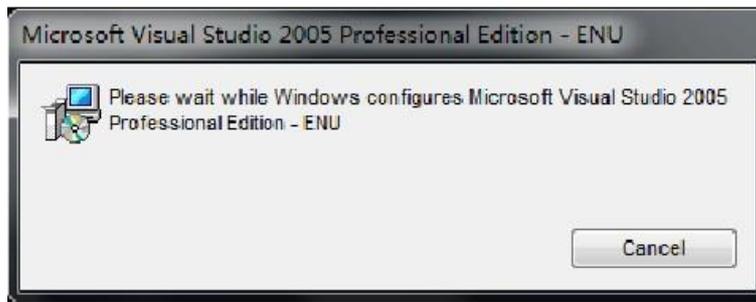
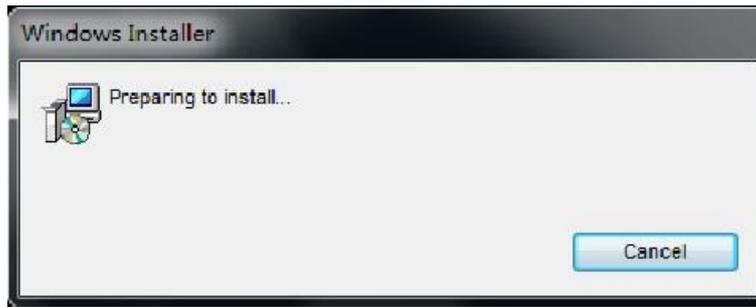


Step17: Installed, the end of the following interface appears, click “OK”



Step18: Next, install the third patch” Visual Studio 2005 Service Pack 1 ATL Security Update”, double-click to run” VS80sp1-KB971090-x86-INTL.exe”, in turn appear as shown in Figure interface

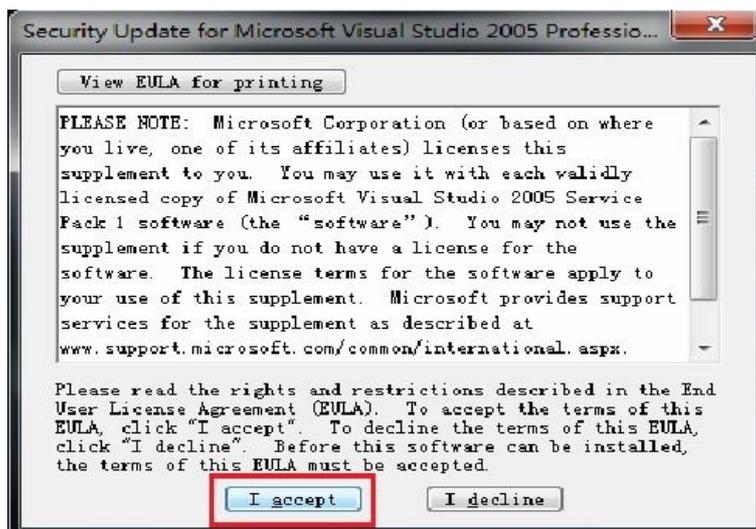




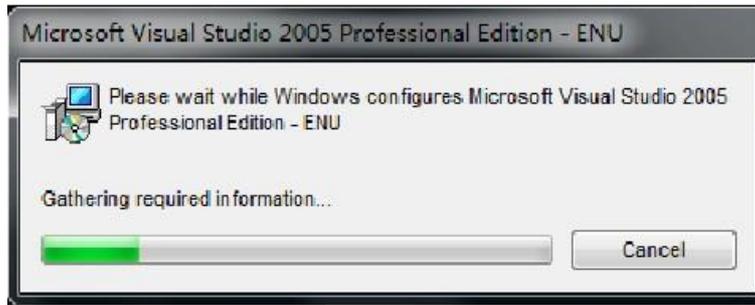
Step19: Wait a moment, appears in Figure interface, click “OK” to continue



Step20: The installation license agreement screen, click “I accept” to continue



Step21: Appear during the installation interface, the longer this process, please be patient



Step22: Installed, the end of the following interface appears, click “OK”



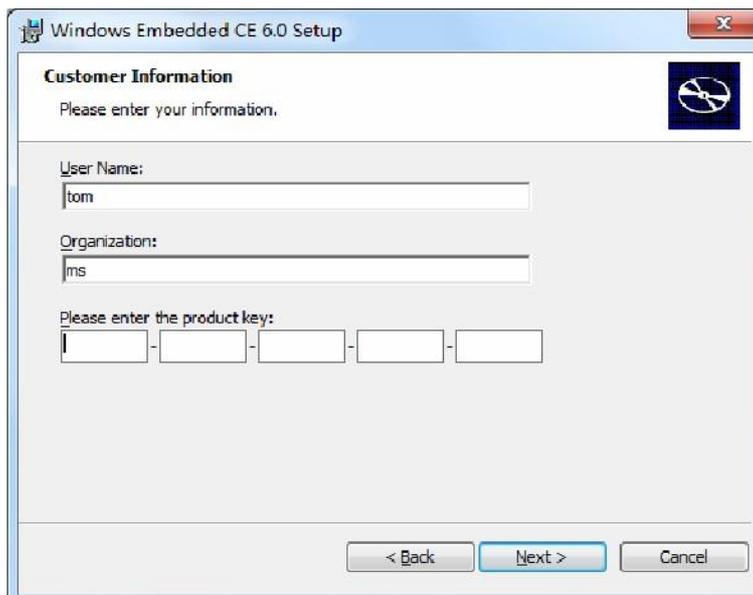
At this point, based on the Windows XP platform, Visual Studio 2005 and its patch has been installed.

4.1.2 Installing the Windows CE 6.0 and patch

Step1: Click on “Windows Embedded CE 6.0.msi” to begin the installation, as shown, click “Next” to continue



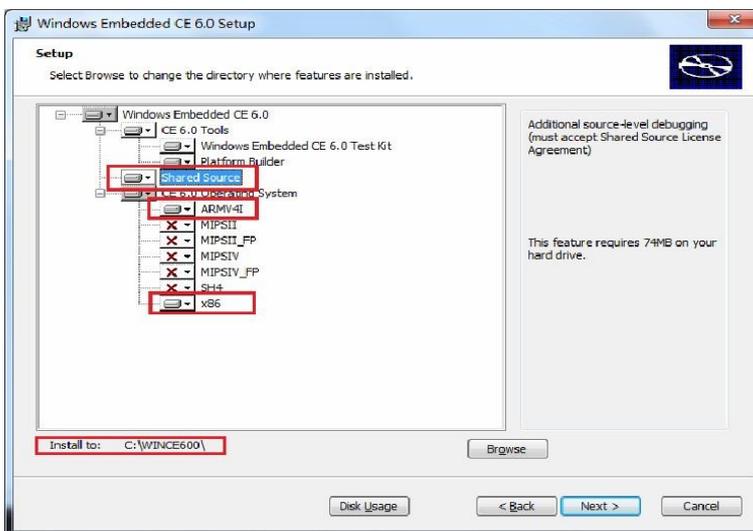
Step2: Enter the product key point , click “Next” to continue



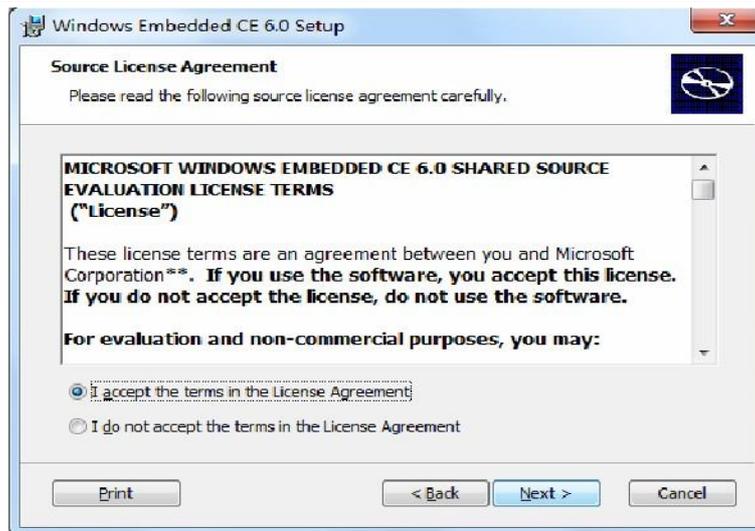
Step3: Appear to install the license agreement screen, select “I accept”, point the “Next” to continue



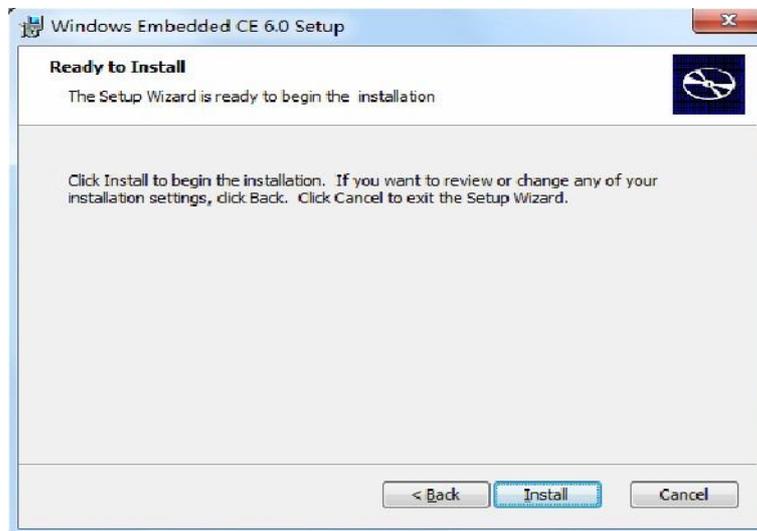
Step4: Select and set like the picture, click “Next” to continue



Step5: Figure interface, select as the picture, click “Next” to continue



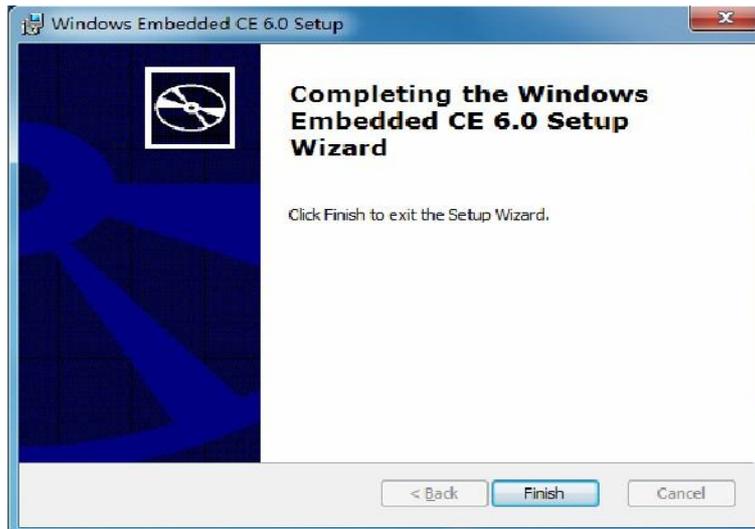
Step6: Figure interface appears, click “Install” to continue



Step7: Started to install, as shown, this process is a long time, please be patient



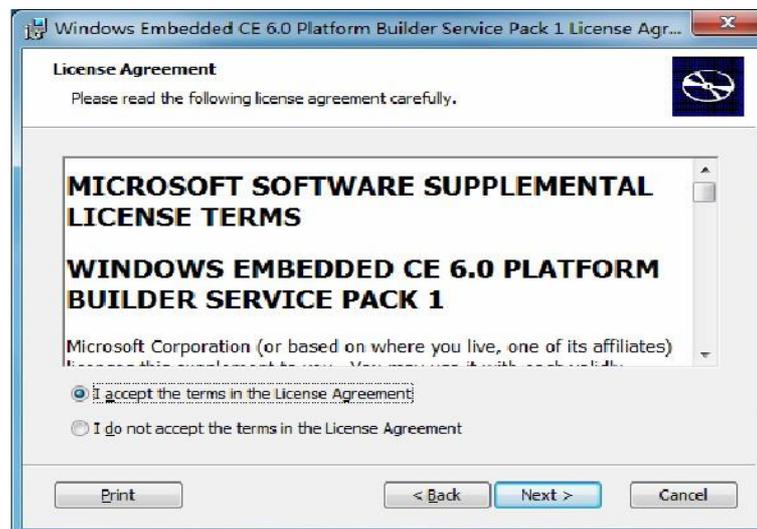
Step8: End of the installation, appears in Figure interface ,point “Finish”



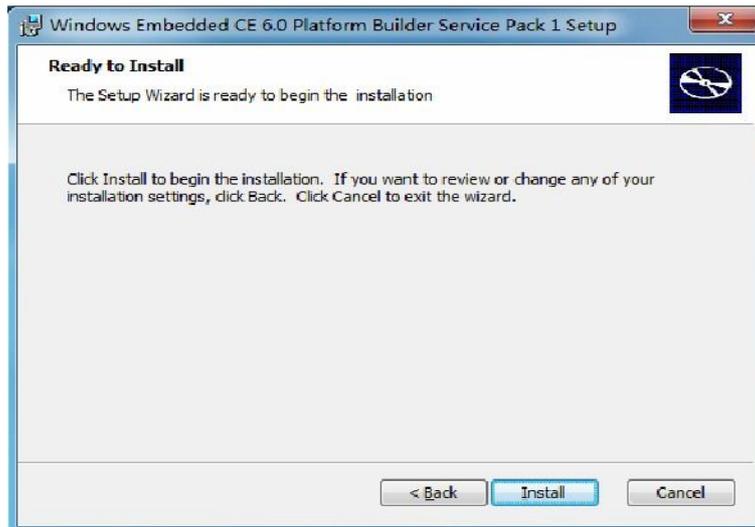
Step9: Next, install the Windows CE 6.0's the first patch for "Windows Embedded CE 6.0 Platform Builder Service Pack 1.msi", click the installation file, appears in Figure interface, point the" Next "to continue



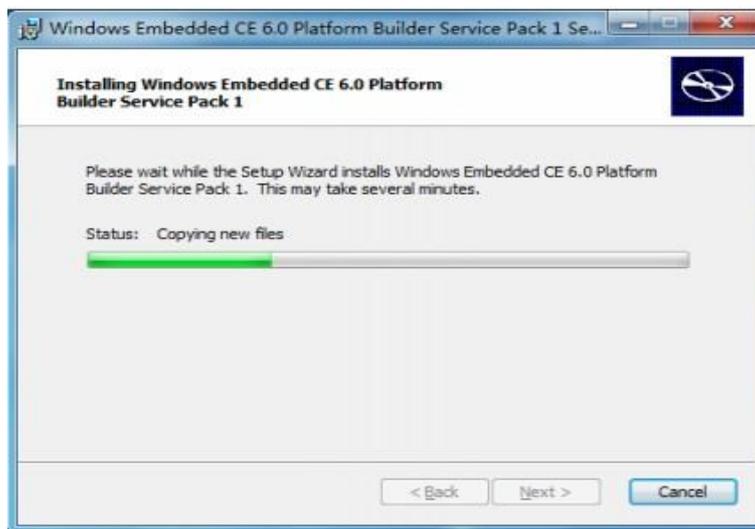
Step10: Figure interface shown, choose "I accept", and "Next" to continue



Step11: Figure interface shown, click "Next" to continue



Step12: Started to install, as shown, this process is a long time, please be patient



Step13: End of the installation, appears in Figure interface ,point “Finish” .



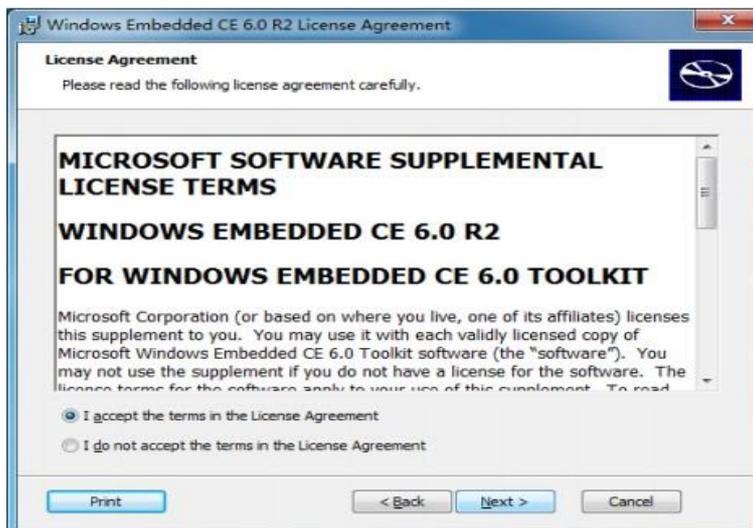
Step14: Next, install the Windows CE 6.0's the second patch “Windows Embedded CE 6.0 R2.msi



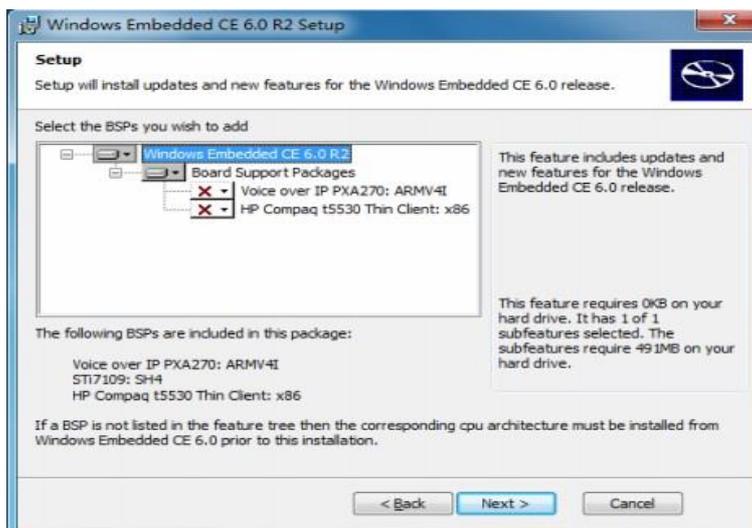
“click “Next” to continue



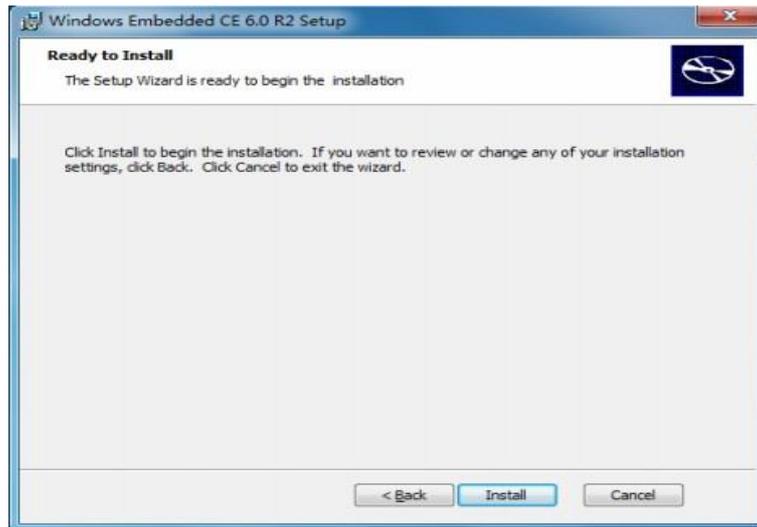
Step15: Figure interface shown, select “I accept”, and point the “Next” to continue



Step16: Figure interface shown, do not make any changes, and just point the “Next” to continue



Step17: Figure interface appears, click “Next” to continue



Step18: Started installation, this process is a long time, please be patient



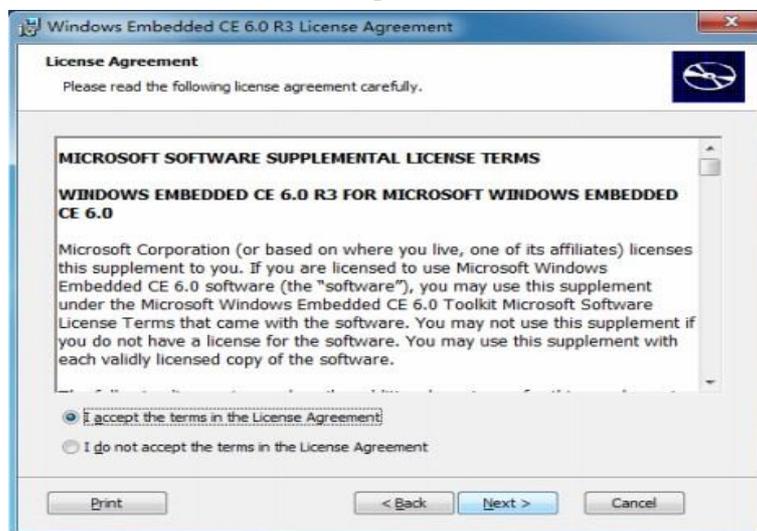
Step19: End of the installation, appears in Figure interface, point "Finish"



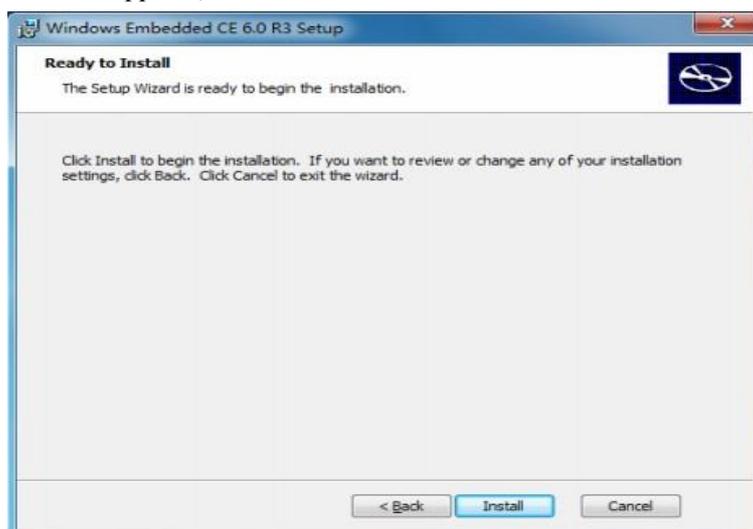
Step20: Now begin to install the third patch "R3" of Windows CE 6.0, start the installation in "Windows Embedded CE 6.0 R3.msi", as shown in Figure



Step21: Figure interface shown, choose “I accept”, and “Next” to continue



Step22: Figure interface appears, click “Next” to continue



Step23: Started installation, this process is a long time, please be patient



Step24: End of the installation, appears in Figure interface, point “Finish”

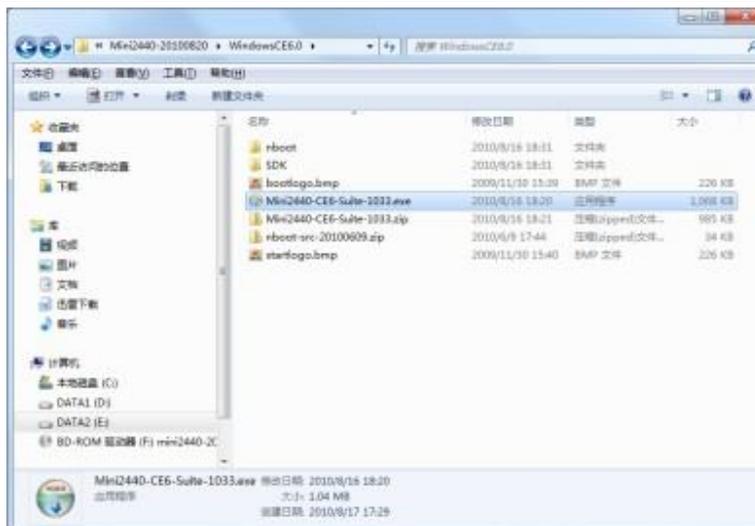


4.1.3 Installing the BSP and core engineering sample

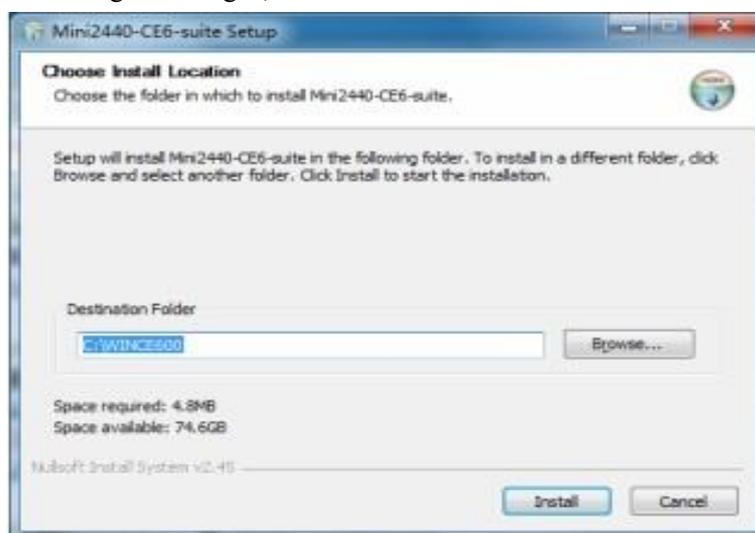
JMY980 (mini2440) BSP and sample project files have only one installation file “mini2440-ce6-suite-1033.exe”, which contains all BSP source code and core engineering sample.

Note: the following steps to install the BSP, it is recommended not to change the installation path, or they may not compile.

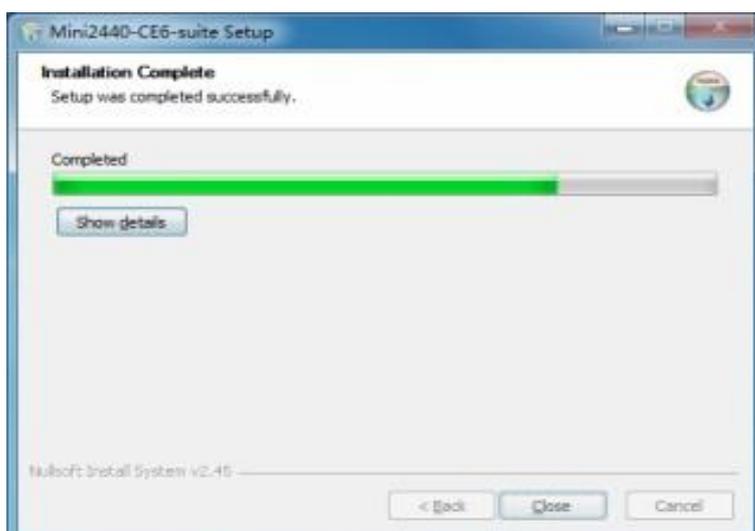
Step1: Find” mini2440-ce6-suite-1033.exe “executable installation file and double-click



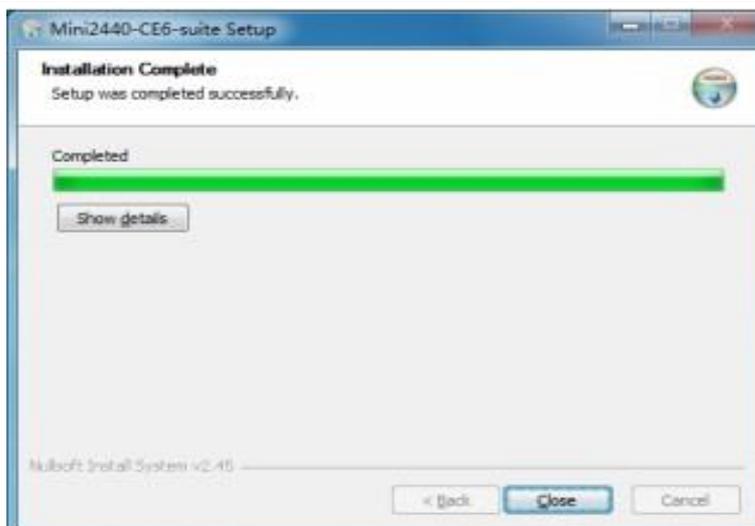
Step2: To keep the settings unchanged, “Install” to continue



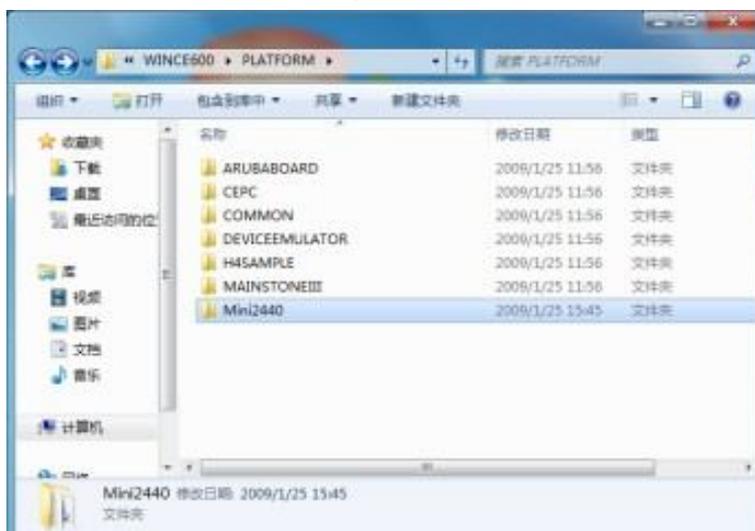
Step3: Appear during the installation interface, because the installation of a small file, the installation will be over soon



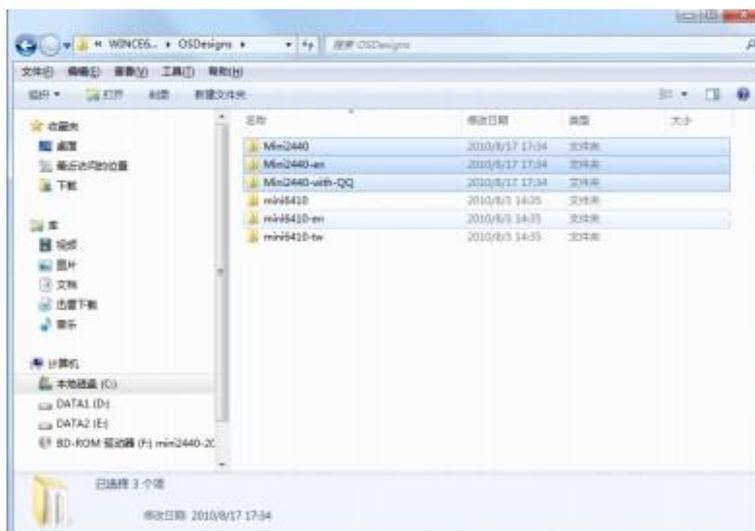
Step4: End of the installation, appears in Figure interface, point “Close”



Installed, will create “mini2440 BSP” directory under “WinCE600 \ PLATFORM” directory, as shown



And create three core sample project file directory in “WinCE600 \ OSDesigns” directory, as shown



Of which:

The project file under” Mini2440 “directory can be used to compile into corresponding to the Chinese version's WinCE kernel image in CD



The project file in “Mini440 - with-QQ” directory can be used to compile into WinCE kernel image with Tencent QQ

The project file in “Mini440-en” directory can be used to generate the English version of the WinCE kernel image

At this point, Windows CE 6.0 development environment has been completely created.

4.1.4 The location of each driver source code

The “mini440” currently has the most complete BSP that means the driver program, and each driver has basic graphical interface test program, location of each driver source code as follows:

- (1) LED Drive
 \Mini2440\SRC\DRIVERS\LEDdriver
- (2) Key Drive
 \Mini2440\SRC\DRIVERS\Userkey
- (3) PWM Control Buzzer Drive
 \Mini2440\SRC\DRIVERS\PWM
- (4) ADC Conversion Drive
 \Mini2440\SRC\DRIVERS\Touch
 Description: ADC driver and the touch screen driver in the same file
- (5) I2C Drive
 \Mini2440\SRC\DRIVERS\IIC
- (6) RTC Drive
 \Mini2440\SRC\DRIVERS\Rtc
- (7) Serial Port Drive
 \Mini2440\SRC\DRIVERS\Serial
- (8) Touch Screen Drive
 \Mini2440\SRC\DRIVERS\Touch
- (9) USB Drive
 \Mini2440\SRC\DRIVERS\Usb
- (10) SD Card Drive
 \Mini2440\SRC\DRIVERS\SDHC
 Description: Support for high-speed large-capacity SD card up to 32GB
- (11) DM9000 Network Card Drive
 \Mini2440\SRC\DRIVERS\dm9000
- (12) Audio Drive
 \Mini2440\SRC\DRIVERS\Wavedev
- (13) LCD Drive
 \Mini2440\SRC\DRIVERS\Display
- (14) Backlight Drive
 \Mini2440\SRC\DRIVERS\Backlight
- (15) CMOS Camera Drive
 \Mini2440\SRC\DRIVERS\Camera



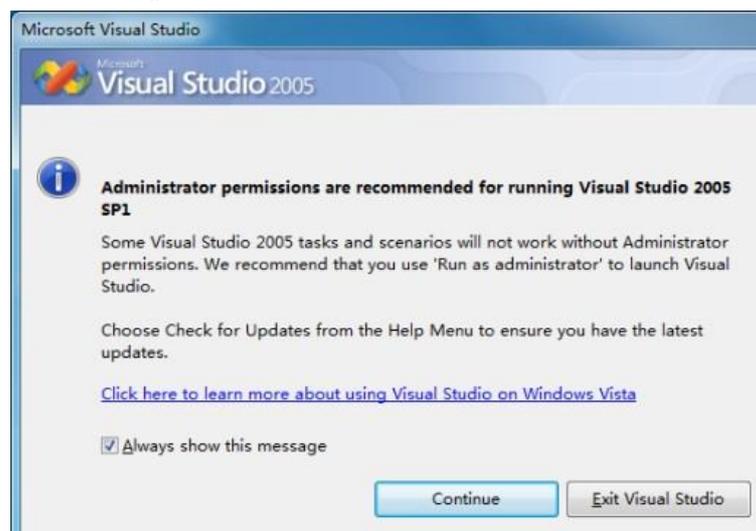
4.2 Compiling and configuring for Windows CE 6.0 kernel and Bootloader

Due to kernel configuration of Windows CE6 is more complex, it is easy to configure wrong so that it can't be compiled successfully. It's well-known that compilation of the "Windows CE" platform is very time-consuming, so the user directly follow the steps below to open the compiler, and "images \ wince6.0" directory in CD contains the corresponding compiled kernel image file.

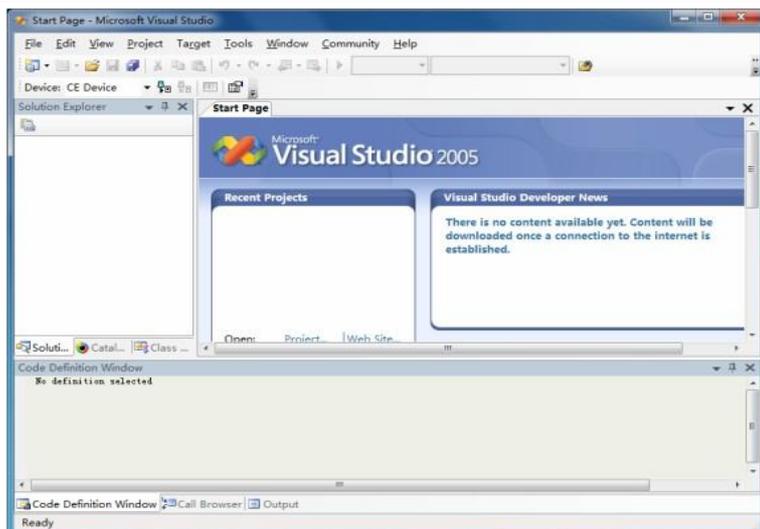
4.2.1 Compiler default kernel engineering example

Now, we start "VS2005" to compile just installed "mini2440 BSP", when you first start ,some matters need to be noted, as the following steps:

Step1: "Start" -> "Programs -> Microsoft Visual Studio 2005 -> Microsoft Visual Studio 2005", appears in Figure interface, point "Continue" to continue



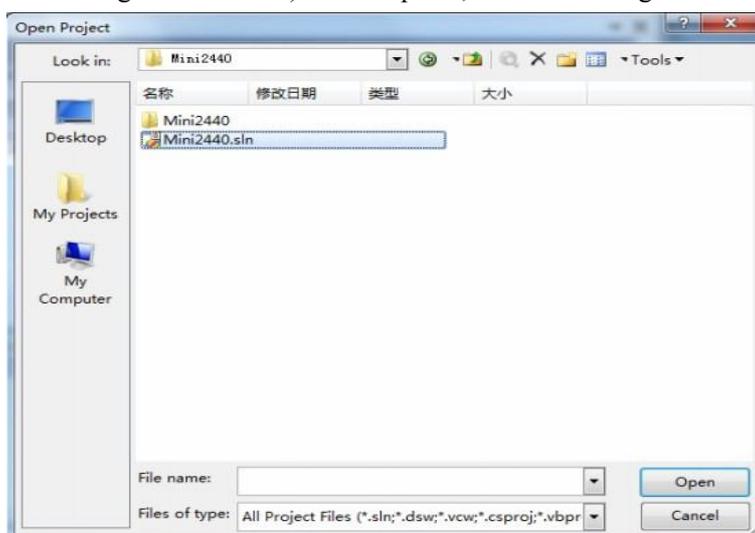
Step2: Figure interface shown, which is a working interface in VS2005, please just refer the VS2005 information



Step3: Point File-> Open-> Project / the Solution ..., as shown in Figure



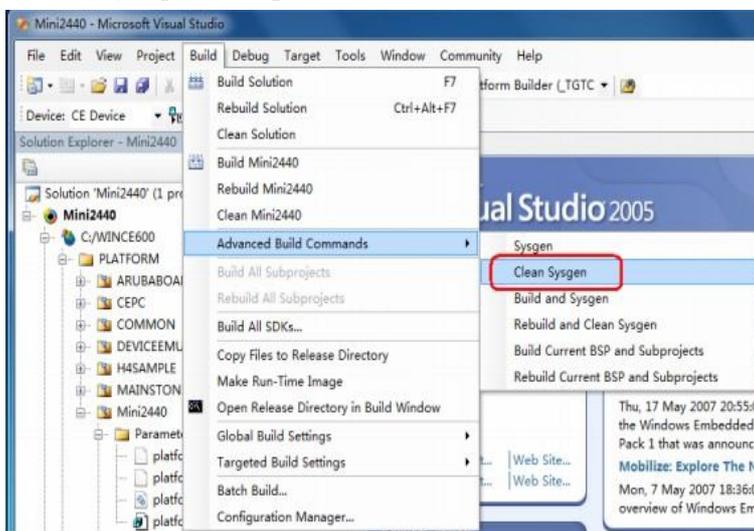
Step4: The file selection window shown, find the default kernel project file of a “mini2440”(path: C: \ WINCE600 \ OSDesigns \ Mini2440) click “Open “, as shown in Figure



Step5: Wait a moment, the default kernel project of a “mini2440” is loaded into workspace,Figure interface appears.

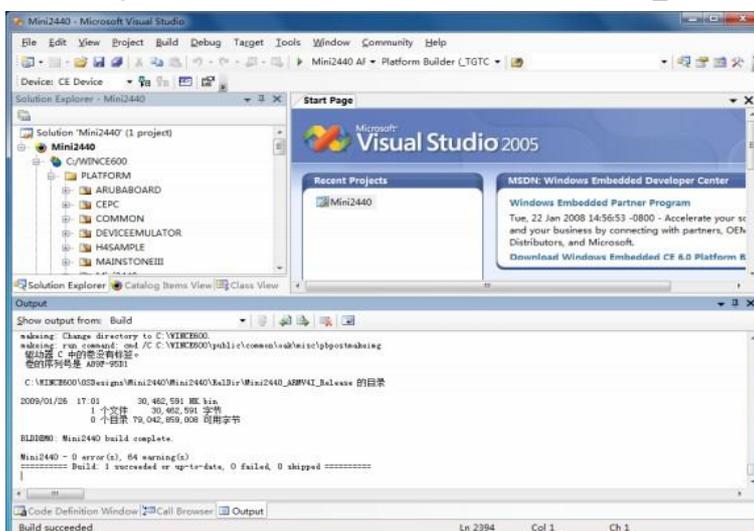


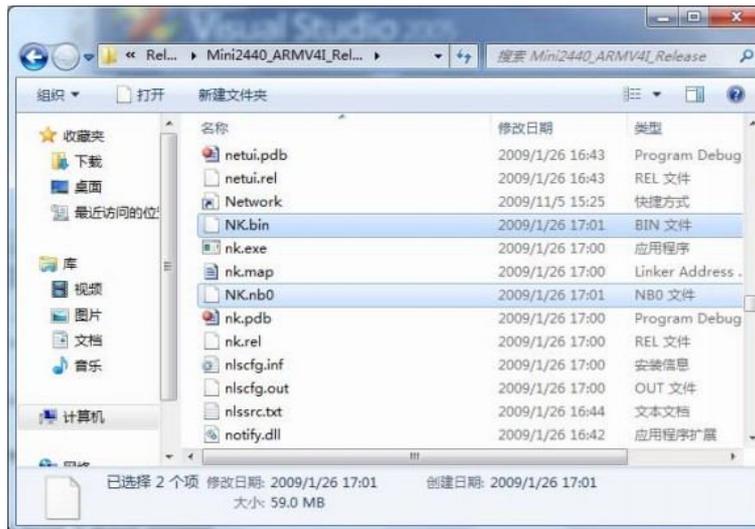
Step6: Point “Build-> Advanced Build Commands-the Clean Sysgen”,begin to compile the kernel, as shown, this process is longer, please be patient



Step7: Compilation is completed, the results shown in the figure, and it will generate kernel image the file “NK.bin “and” NK.nb0”, as the following path:

C:\WINCE600\OSDesigns\Mini2440\Mini2440\RelDir\Mini2440_ARMV4I_Release





4.2.2 Compiling and programing NBOOT of the Bootloader

Description: compiling Nboot needs to use the ADS integrated development environment, see Chapter 5.

The Nboot is a very simple bootloader, its size is less than 4K, generally programmed to 0 position Block of Nand Flash to start the WinCE kernel. Nboot is originally supplied by Samsung, which we have done a lot of improvements, currently has the following features:

- Adaptive support 64M/128M/256M/1G JMY980
- Support the start-up screen to quickly display
- Support the dynamic progress bar to load the WinCE kernel
- Start WinCE only 5-10 seconds, depending on the kernel size

Should be noted that Nboot do not have the programming function, it can only be read the file has been programmed: boot screen (BootLogo) and WinCE kernel.

Treated Nboot has a very convenient custom, you can modify the display position of the start-up screen ,background and progress bar color, position, length and width, etc. These definitions are in the option.h file, as follows:

// Select the appropriate LCD models by changing the definition, here is default selected by Q35, means Chi Mei horizontal screen LCD

```
##define LCD_N35
##define LCD_L80
##define LCD_Q35
##define LCD_X35
##define LCD_W35
##define LCD_A70
##define LCD_VGA1024768
```

```
// Set the background color
#define BACKGROUND_R 0x00
#define BACKGROUND_G 0x00
#define BACKGROUND_B 0x00
```



```
// Set the progress bar's color
#define PROGRESS_BAR_R 0xFF
#define PROGRESS_BAR_G 0xFF
#define PROGRESS_BAR_B 0x00

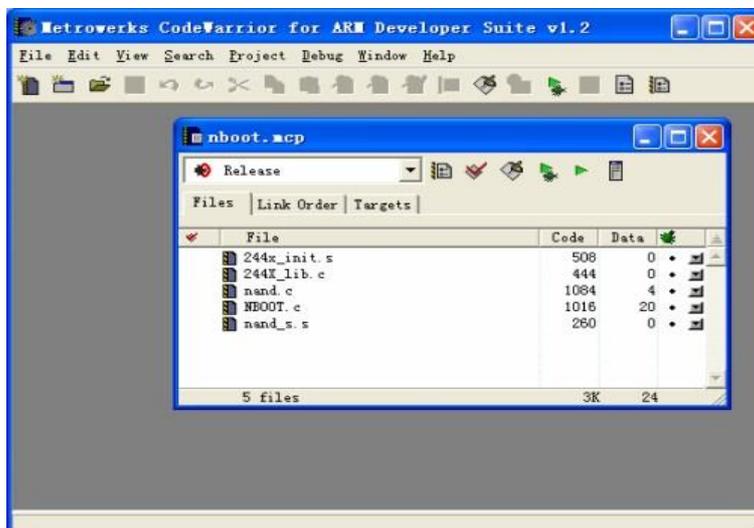
// Set the location of the boot picture
#define LOGO_POS_TOP 0
#define LOGO_POS_LEFT 0

// Set the start position and aspect
#define PROGRESS_BAR_TOP 260
#define PROGRESS_BAR_LEFT 20
#define PROGRESS_BAR_WIDTH 200
#define PROGRESS_BAR_HEIGHT 12
```

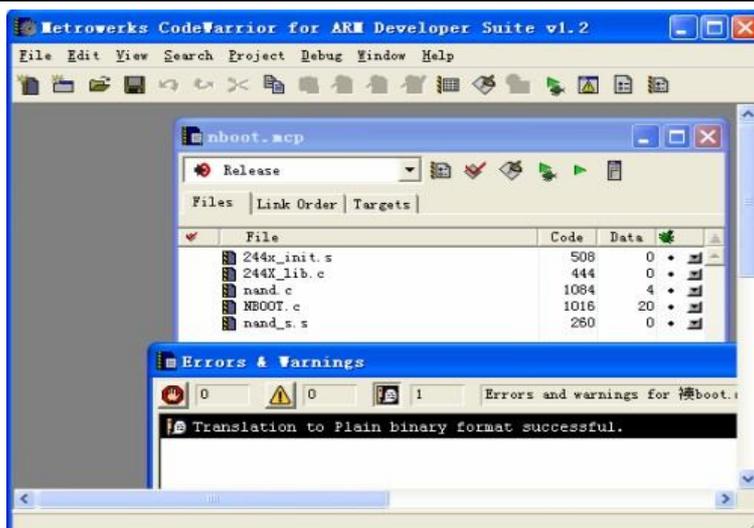
Here are the compilation methods and procedures for Nboot:

Compiling Nboot

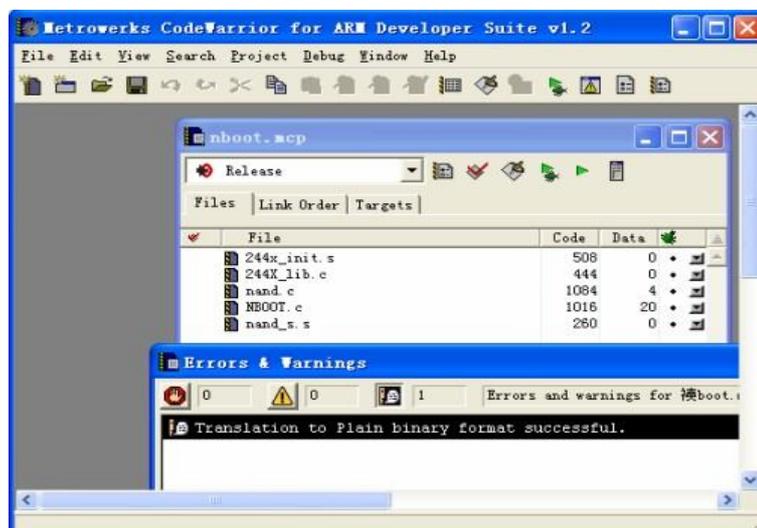
“WindowsCE6.0” NBOOT “folder under CD –ROM should be copied to a directory in hard drive(here is D:\work), remove the read-only attribute, run ADS1.2 integrated development environment, point file-> Open ... the open nboot.mcp file, as shown in Figure.



Click menu Project-> Make or simply press the F7 key, start to compile nboot project, completed as shown in Figure:



It will generate “nboot.bin” executable file under D: \ work \ NBOOT \ nboot_Data \ a DebugRel directory, as shown in Figure.



NBOOT need to be programmed to Nand Flash in the development board.

4.2.3 Modifying the LCD type and serial output function in the BSP

Description: We provide the BSP currently that supports the following types of LCD:

- Chi Mei 3.5-inch LCD with touch
- NEC3.5 inch screen with touch
- TPO 3.5 inch LCD with touch
- Sharp 8-inch LCD with touch
- 7-inch screen with touch

By modifying definition of “LCD TYPE” under mini2440 \ Src \ LCD_TYPE Inc. \ options.h header file, you can select the LCD type:

```
//#define LCD_Q35 fit to QIMEI 3.5-inch LCD
```



```
//#define LCD_L80 fit to Sharp 8-inch LCD
```

```
//#define LCD_X35 fit to Sony3.5-inch LCD
```

Tip: The default LCD model in the CD-ROM is LCD_Q35.

In “Options.h” file, the user can modify the serial output function: as an ordinary serial port functionality or debugging output (limited to serial ports 1 and 2), as the following definition:

```
#define KITL_NONE  
#define KITL_SERIAL_UART0  
#define KITL_SERIAL_UART1  
#define KITL_USBSERIAL  
#define KITL_ETHERNET
```

Here the default is defined as a normal serial port function, if we want to serial port 1 as output debugging information, should be defined as:

```
//#define KITL_NONE  
//#define KITL_SERIAL_UART0  
//#define KITL_SERIAL_UART1  
//#define KITL_USBSERIAL  
//#define KITL_ETHERNET
```

4.2.4 Creating and editing Windows CE for startuapping Logo

In the previous chapter, we mentioned:

Windows CE system’s boot process has two Logo: BootLogo and StarLogo. BootLogo has the display of Nboot load, users can modify Nboot source to adjust BootLogo display location and background color; StartLogo are part of the BSP, it is an array of files (StartLogo.c) in mini2440 \ Src \ Kernel \ Oal “directory, can achieve load display by the directory” init.c” file, “StartLogo.c” file can be generated through” CD StartLogoMaker.exe” tools.

StartLogoMaker is transplanted from “LogoMaker” that is one of “Linux Logo” creation tools, is a “green software”, it does not require installation, directly copied to the WindowsXP / Vista platforms for running, and use it to convert bmp, jpg, png format images for” StartLogo.c”array file that “mini2440 BSP” needs, with the newly generated file to replace the file with the same name in the BSP, you can replace the splash screen of “WindowsCE” ,”StartLogo.c” array head content is the followings:

```
//Automatic generated by StartLogo.exe
```

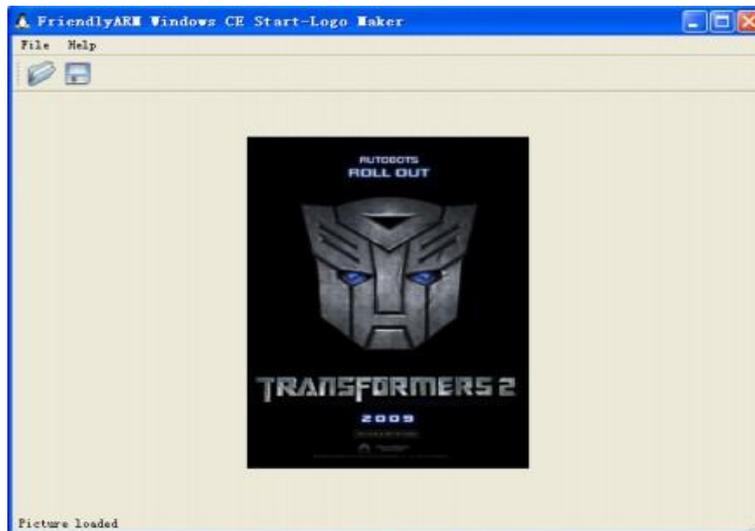
```
Static const unsigned short StartLogoData[] = {  
240,320  
0x965, 0x945, 0x164, 0x9C4, 0x1246, 0x22CA, 0x22A8, 0x2AA7,
```

Here is steps that use” StartLogoMaker.exe “to make “StartLogo.c”:

Step1: Double-click to run “StartLogoMaker.exe” program in “windows 平台工具\ StartLogoMaker”, as the Figure interface shown:



Step2: click File-> Open to open an image file, can also be in the toolbar, point  icon to open the file selection window:



Step3: Point File-> Convert, or click tool bar icon  to open the file output selection window:



click”确定”,and in the appropriate directory, ”StartLogo.c” file will generate:



Step5: the generated files replace the same name files (located in mini2440-BSP \ Src \ Kernel \ Oal directory)in the BSP, recompile the kernel, and program it into the board for running, you can see the WinCE startup screen that belongs to yourself:



4.2.5 Creating SDK

SDK applies: When developing host only installed VS2005, but did not install the Windows CE 6.0 Platform Builder plug, then developers want to develop mini2440 application program through VS2005, need an SDK, which is similar to the SDK that the Embedded Visual C++ requires.

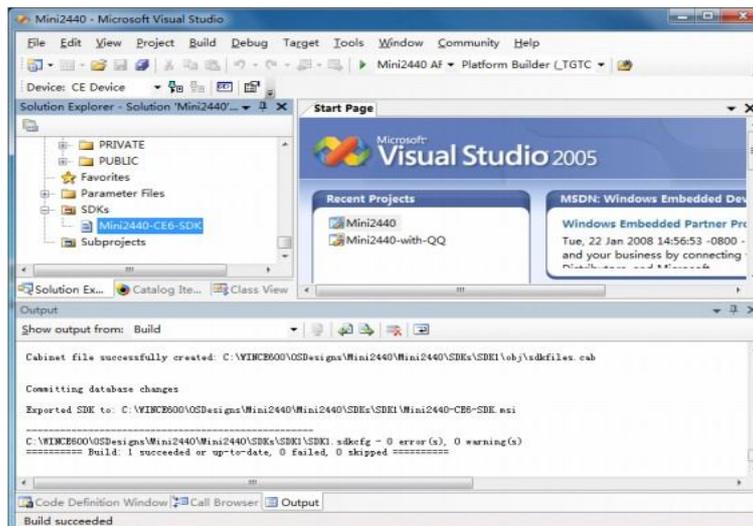


When you compiled the default kernel, created corresponding SDK by the VS2005 platform. Note: the SDK here applies only to the VS2005 development environment, it can not be installed to the EVC, and you can not install to VS2008, the following is the detailed steps to create the SDK.

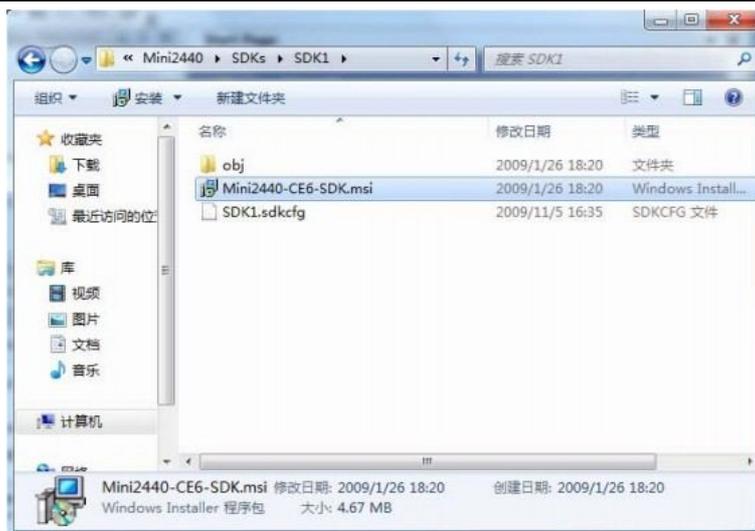
Step1: Running VS2005 and open the default kernel compiled sample project “mini2440”, find in Figure position, and right-click on the “Mini2440-CE6-SDK” menu, point “Build” to start creating the SDK



Step2: Wait a moment, SDK created as shown



Step3: In “C:\WINCE600\OSDesigns\Mini2440\Mini2440\SDKs for\SDK1” directory, you can see “Mini2440-CE-SDK.msi” installation files that has been generated.



4.2.6 Installing SDK

Developing Mini2440 application through VS2005, you need to install the SDK that just generated as the following steps:

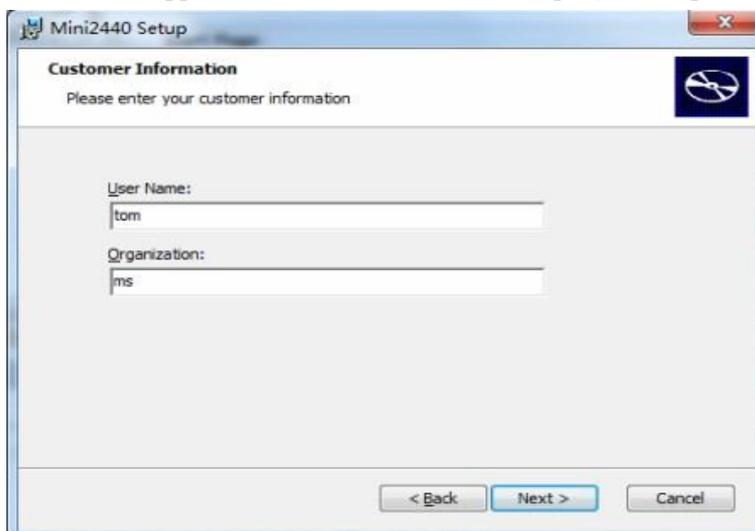
Step1: Double-click to run “Mini2440-CE6-SDK.msi”, the following interface appears, click “Next” to continue



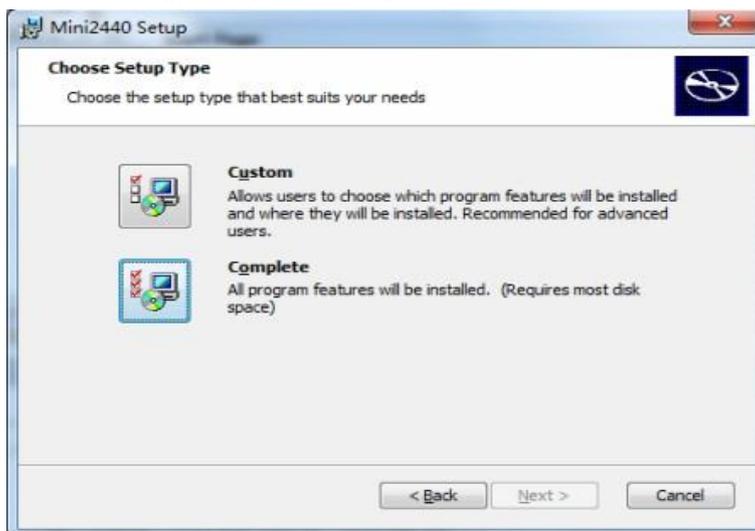
Step2: as shown,choose “I accept”, point “Next” to continue



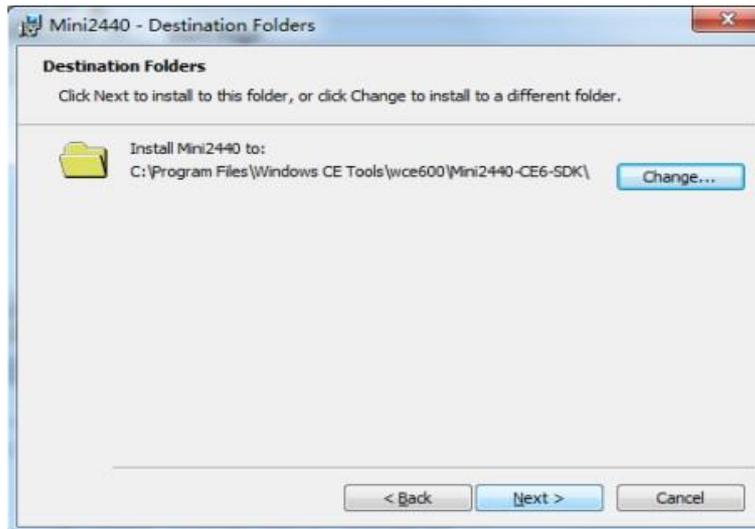
Step3: Figure interface appears, enter a user name and company name, point “Next” to continue



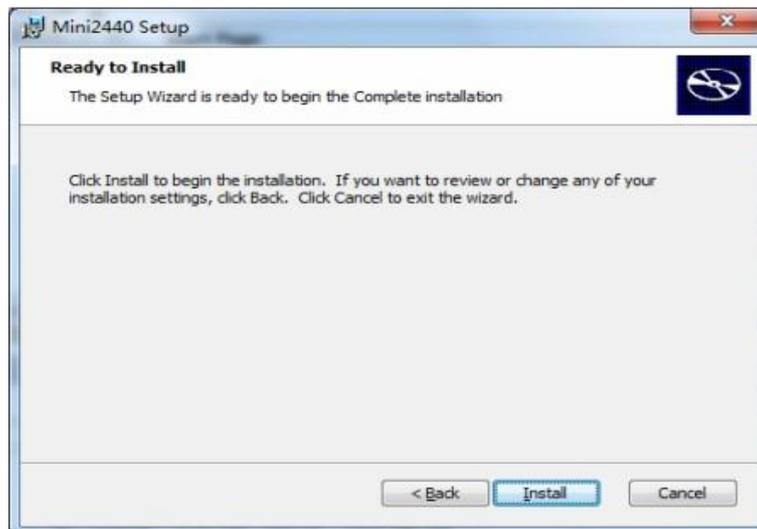
Step4: Figure interface appears, point “Complete” to continue



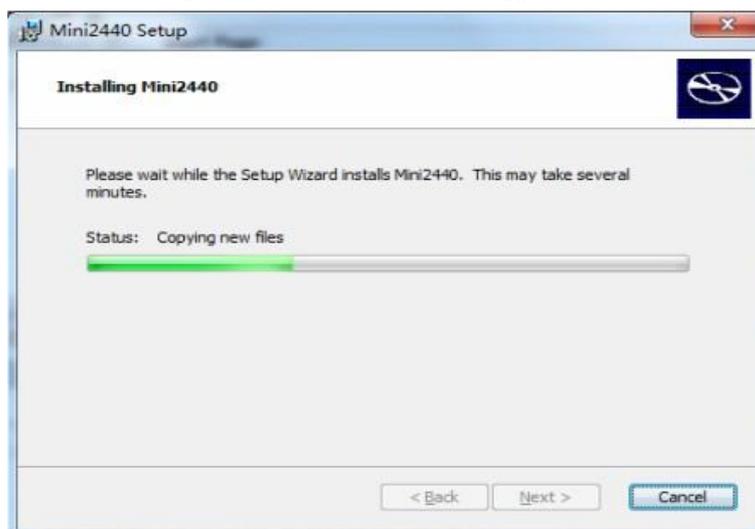
Step5: Figure interface appears, click “Next” to continue



Step6: Figure interface appears, click “Install” to continue



Step7: Figure interface of progress of the installation appears, wait a moment



Step8: End of the installation interface appears, point “Finish”



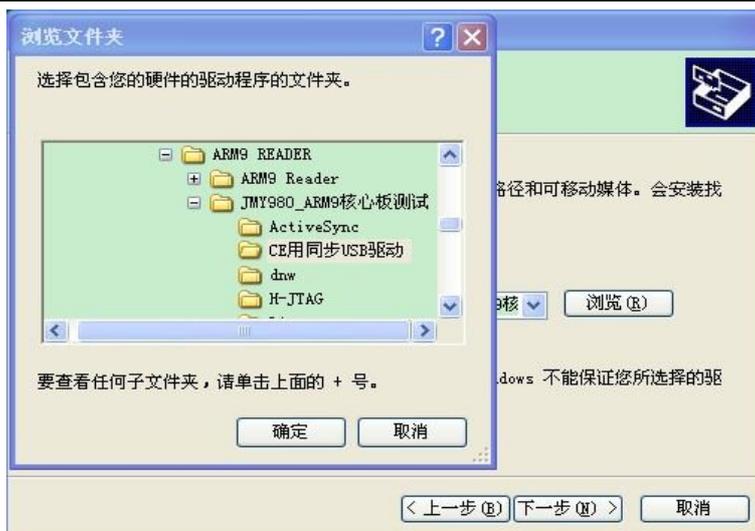
4.3 With PC synchronization

4.3.1 Installation of synchronous drive and software

Step1: JMY901 with PC connects by USB cable, and open JMY901 power ,the following interface will pop up on your PC, point the “下一步” to continue



Step2: Click “Browse”, select “CE sync USB driver, click” OK “, click” Next “to continue



Step3: The following interface appears, click “finish”



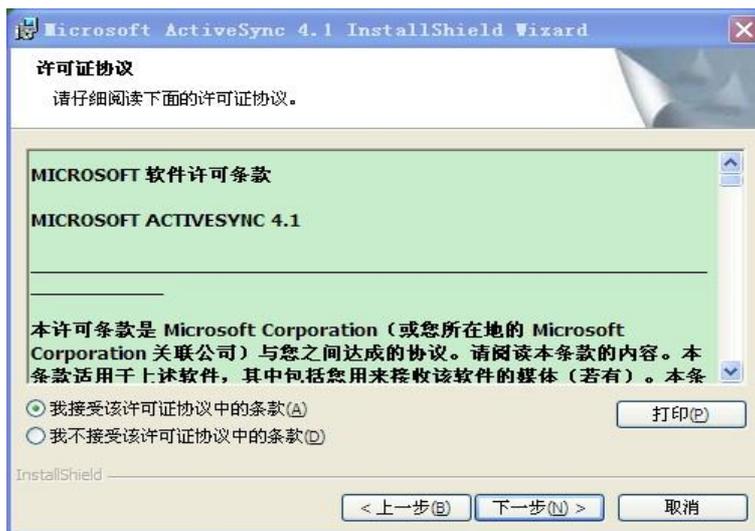
Step4: Installation ActiveSync synchronization software, the following interface appears, click “Next” to continue to



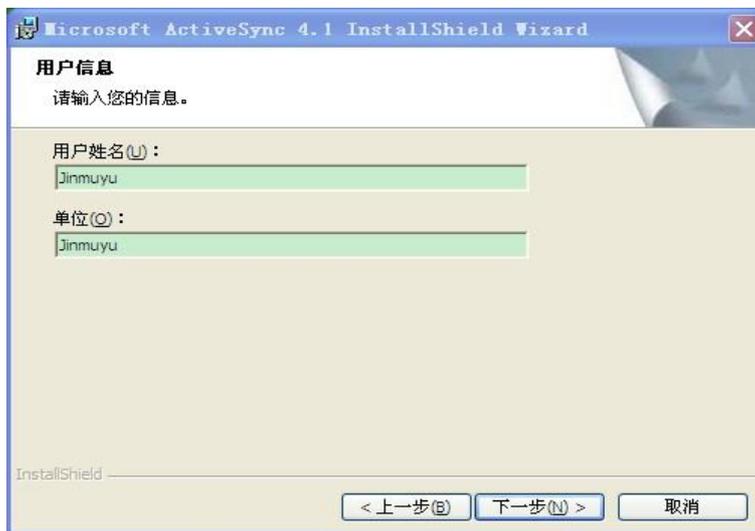
Step5: The following interface appears, select “I accept the terms of the license agreement, click” Next



“to continue



Step6: The following interface appears, enter your user name and company, and click “Next” to continue to



Step7: The following interface appears, select the installation path, the default can click “Next” to continue

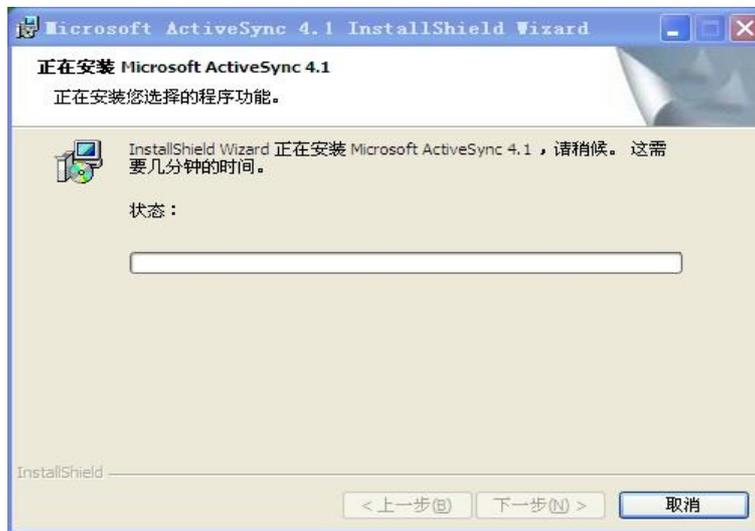




Step8: The following interface appears; click “Installation” to install the software



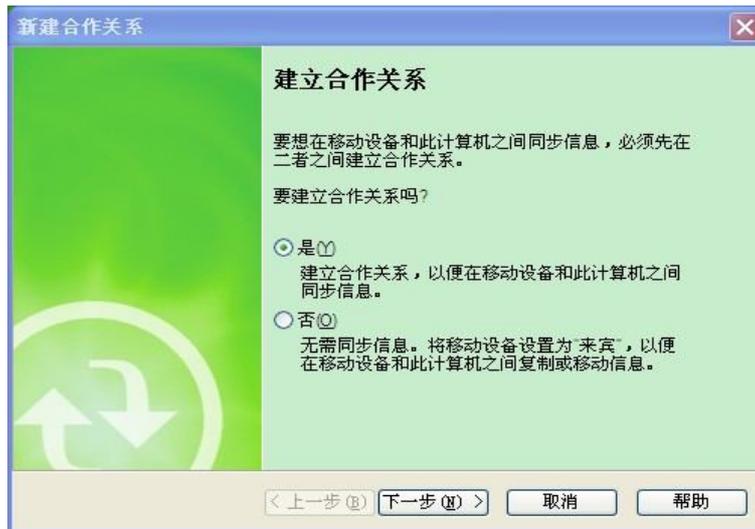
Step9: The following interface appears, please hold on



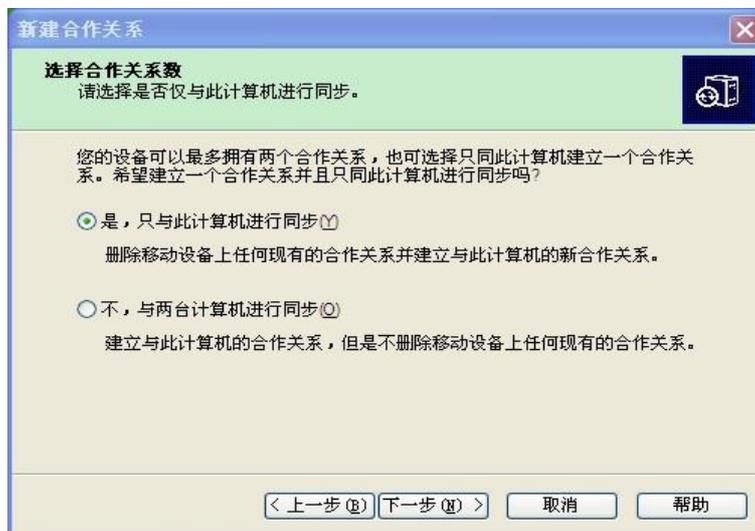
Step10: The following interface appears, click “Finish” to finish the installation



Step11: The following interface will pop up after installing the ActiveSync software in PC, select “Yes”, and click “Next” to continue



Step12: Showing the following interface, select “yes, ...” Click “Next” to continue to



Step13: Showing the following interface, do not any change, hit “next step”, and go on



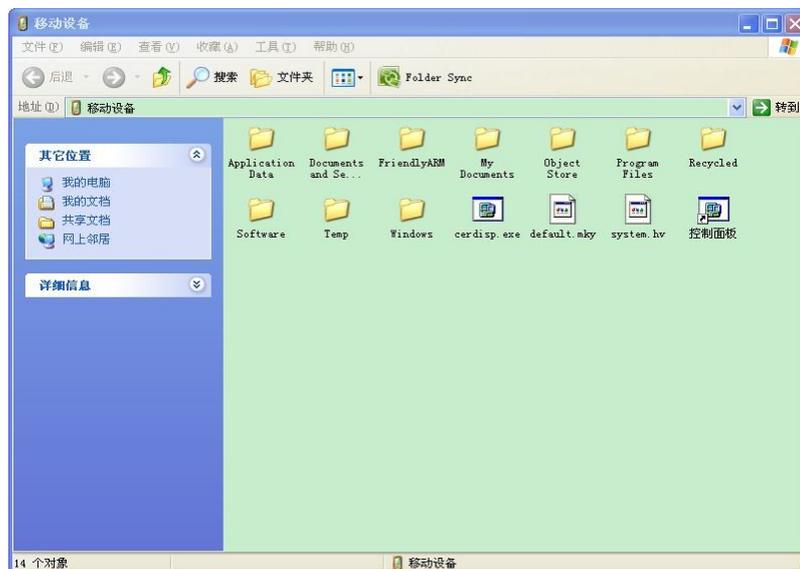
Step14: To show the following interface, click “Finish”, the end of the set



Step15: The PC will automatically pop up the following interface, showing “Connect”, to click “Browse”.



Step16: Pop-up the following interface: this is the WinCE6.0 file folder. PC and JMY901 completed synchronously.





4.4 Create an application, compile and download to development board running through VS2005

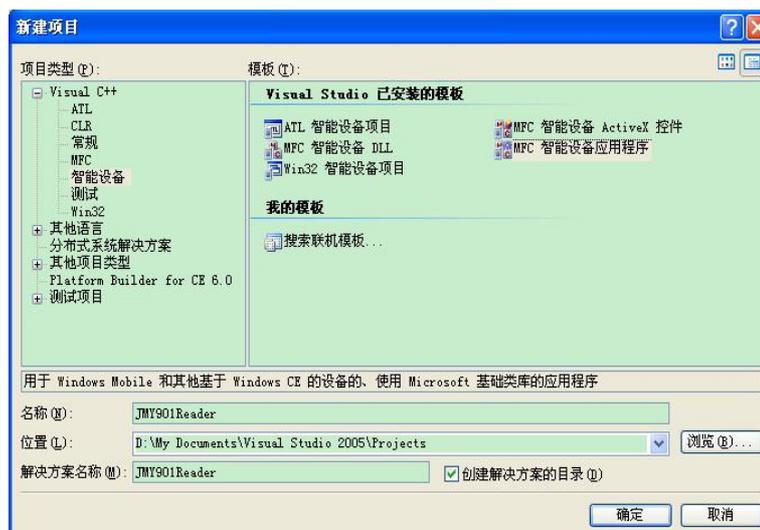
The following is the basic development steps using VS2005:

4.4.1 Creating a Project

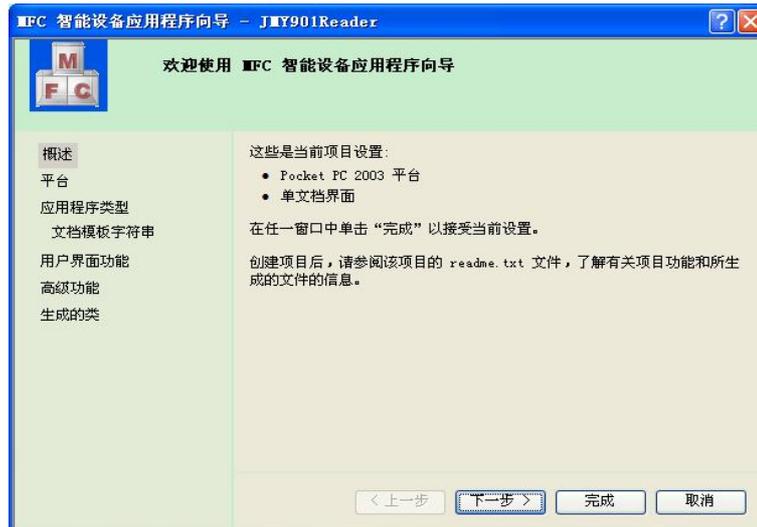
Step1: Open and running VS2005, hit menu file -> New -> Project, as shown:



Step2: "New Project" appears, select "Visual C ++" to Expand; select the Smart Device; Select "the MFC smart device application program" in "the Visual Studio has to install the template," enter name: "JMY901Reader", click OK



Step3: The following interface appears, click "Next"



Step4: The following interface appears, click , Canceled “Pocket PC 2003”



Step5: Selected “Mini2440-CE-SDK” on the left”, hit , hit “Next”





Step6: The following interface appears: select the "Dialog-based"; resource language "English (United States)" is selected, click "Next"



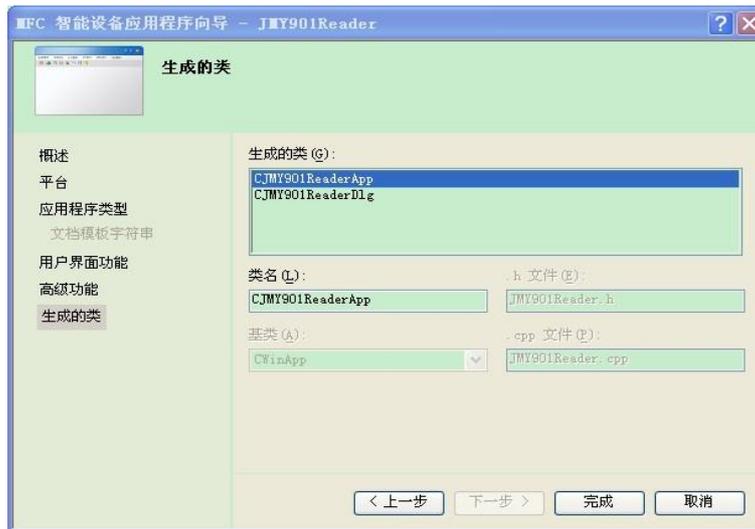
Step7: The following interface appears, click "Next"



Step8: The following interface appears, click "Next"



Step9: The following interface appears, click "Finish"



Step10: The following interface appears: WinCE application development set finished; then start to program!