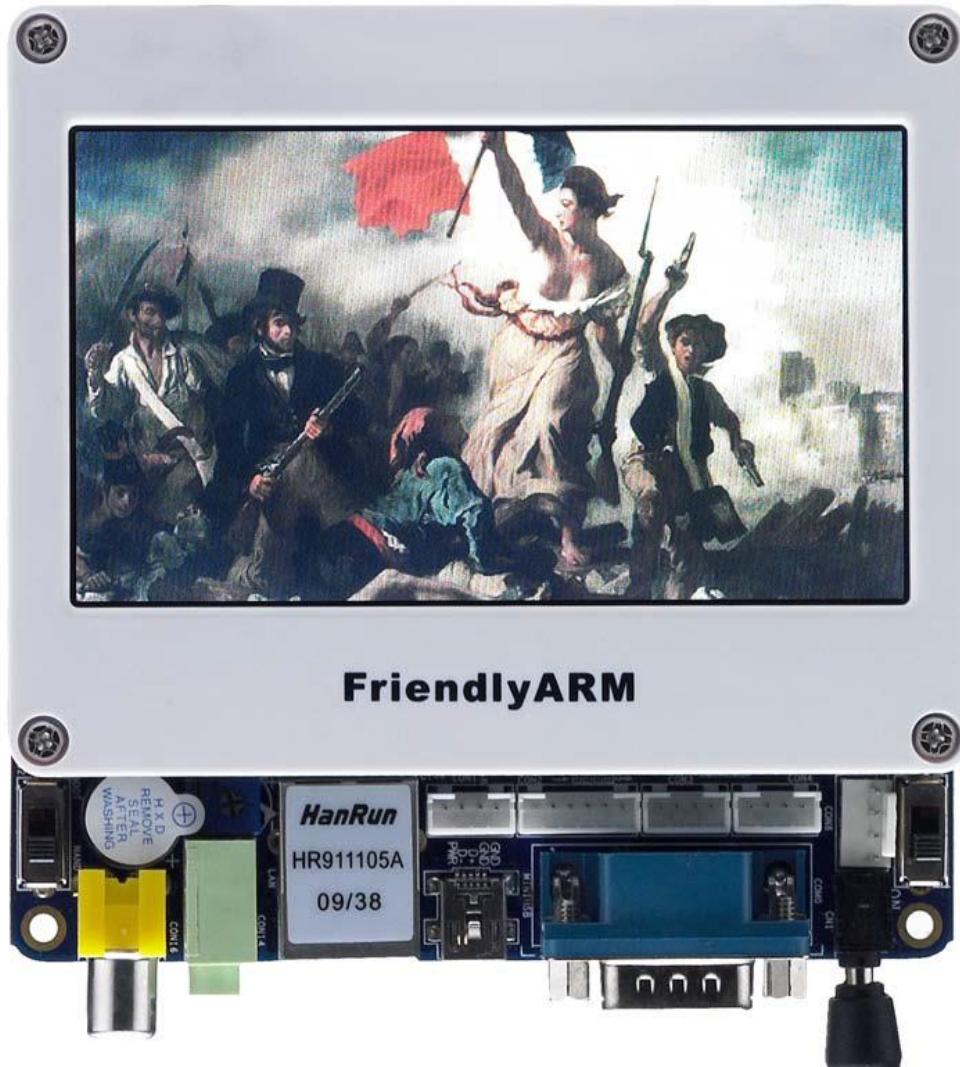




Mini6410 For Dummies



REVISION	ORIGINATOR	SCR	REV DATE
0.1.0	FriendlyARM Co., Ltd		March 28th, 2011
FriendlyARM Co., Ltd Confidential: This document and information contained in it shall not be reproduced by, used by, or disclosed to others except as expressly authorized in writing by FriendlyARM Co., Ltd.			FriendlyARM Co., Ltd Guangzhou, China

copyright@2010



COPYRIGHT STATEMENT

The content (content being images, text, programs and scripts) of this English manual is copyright © FriendlyARM Co., Ltd. All rights expressly reserved.

Any content of the manual printed or downloaded may not be sold, licensed, transferred, copied or reproduced in whole or in part in any manner or in or on any media to any person without the prior written consent of FriendlyARM Co., Ltd including but not limited to:

- transmission by any method
- storage in any medium, system or program
- display in any form
- performance
- hire, lease, rental or loan

Requests for permission to reproduce material from this manual should be addressed to FriendlyARM Co., Ltd.



Index

1 System Setup and Configurations	5
1.1 Boot Options.....	5
1.2 Connect Peripherals.....	5
1.3 Set up Super Terminal	5
2 Burn BIOS to SD Card.....	7
2.1 Introduction to BIOS.....	7
2.2 Burn BIOS in WindowsXP	9
2.3 Burn BIOS in Windows7.....	13
3 Introduction to Superboot.....	20
3.1 Install USB Download Driver	21
3.2 Superboot's Menu	29
4 Play with Superboot	31
4.1 Install WindowsCE.....	31
4.2 Install Android.....	32
4.3 Restore Linux	33
4.4 Run Ubuntu	34
4.5 About FriendlyARM.ini	34



This manual is intended to give you some quick knowledge about our Mini6410 step by step. We strongly recommend our users to read it first. Although those details may seem tedious and boring they list very useful references and notes which will be of great help for developers.

By default, all our systems have been preinstalled Linux (located in the shipped CDs' directory /images/Linux are **u-boot, zImage, root-qtopia-qt4.img** etc) therefore you could easily play once you get it.



1 System Setup and Configurations

1.1 Boot Options

Switch “S2” to the “SDBOOT” side, the system will boot from SD/SDHC card.

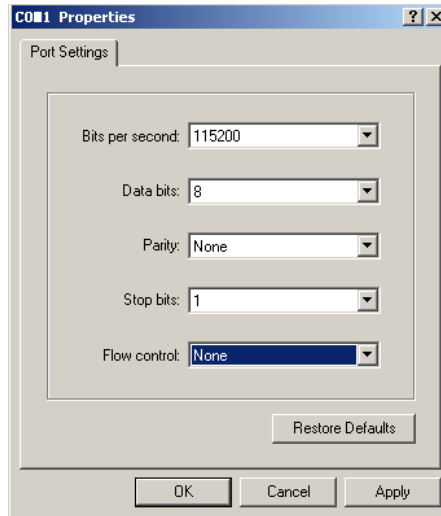
Switch “S2” to the “NAND” side, the system will boot from Nand Flash.

1.2 Connect Peripherals

- Connect Mini6410’s COM0 to a PC’s serial port with the shipped serial cable (**blue one**) in the box
- Connect the shipped 5V power supply
- Connect the LCD module (if you have)

1.3 Set up Super Terminal

Setup your serial terminal like this:





2 Burn BIOS to SD Card

Note1: common SD cards are formatted to FAT32 by default. If there is a huge amount of data in a SD card, auto burning will damage the data. To avoid this issue, in Vista/Windows7 we automatically format a SD card to two sections: one is FAT32 (named “FriendlyARM”) for users’ data and the other (by default 130M) left for the bootloader.

In fact, Vista/Windows7’s system security policies don’t permit unauthorized users to start auto burning an SD card thus ordinary users need to format the SD card first and then burn data into it. For WindowsXP users it is reported that quite a lot of them don’t want to upgrade their systems to Windows7 nor do they care much about this issue therefore we just set the burning mode to auto burning, the same as what Samsung does.

2.1 Introduction to BIOS

● U-Boot

Samsung offers a U-Boot that can download files via USB. We improved this function and make it open sourced for all embedded development lovers and fans. The main features are as follows:

1. Add a pull-down menu, which is similar to the one in Superboot’s USB download menu.
2. Add configuration options for SD card booting
3. Support direct downloading and burning of the YAFFS2 file system



4. Support burning of NBOOT for WindowsCE Boot Loader
5. Support burning of WindowsCE image
6. Support burning of standalone files i.e. standalone programs
7. Users can return to the startup shell during operations

Note: the open source u-boot can only support SLC nand flash and doesn't support MLC nand flash at present.

● Superboot

Besides the U-Boot, we developed another powerful bootloader – “Superboot”, It is not open sourced but free to use for all people.

Below is a table list of features of different bootloaders

Items	Superboot	U-Boot by Friendly ARM	Other U-Boot
Developer	FriendlyARM	FriendlyARM	Others
Download Menu	Yes	Yes	No
Auto detection of 128M/256M DDR RAM	Yes	No	No
Large size image file (> RAM)	Yes	No	No
USB download StepLoader, such as Nboot	Yes	Yes	Yes
USB download Linux kernel	Yes	Yes	Yes
USB download Yaffs2	Yes	Yes	No
USB download UBIFS	Yes	No	No
USB download EBOOT.nb0	No, unnecessary	No, unnecessary	Yes
USB download WindowsCE image NK.bin	Yes	No	No
USB download WindowsCE image NK.nb0	No, unnecessary	Yes	No
USB download WindowsCE boot logo	Yes	No	No
USB download standalone programs	Yes	Yes	Yes
Boot Configuration	Yes	Yes	Yes
SD card (< 2G)	Yes	Yes	Yes



SDHC	Yes	No	No
SD card quick-auto-burning bootloader	Yes	No	No
SD card quick-auto-burning Linux kernel	Yes	No	No
SD card quick-auto-burning Yaffs2	Yes	No	No
SD card quick-auto-burning UBIFS	Yes	No	No
SD card quick-auto-burning WindowsCE image NK.bin	Yes	No	No
SD card quick-auto-burning WindowsCE boot logo	Yes	No	No
SD card quick-auto-burning standalone programs	Yes	No	No
Running Linux on SD card	Yes	No	No
Running WindowsCE on SD card	Yes	No	No
Running Ubuntu on SD card	Yes	No	No
Running Android on SD Card	Yes	No	No
Running standalone programs on SD card	Yes	No	No
SD card rapid reading and burning (see note1)	Yes	No	No

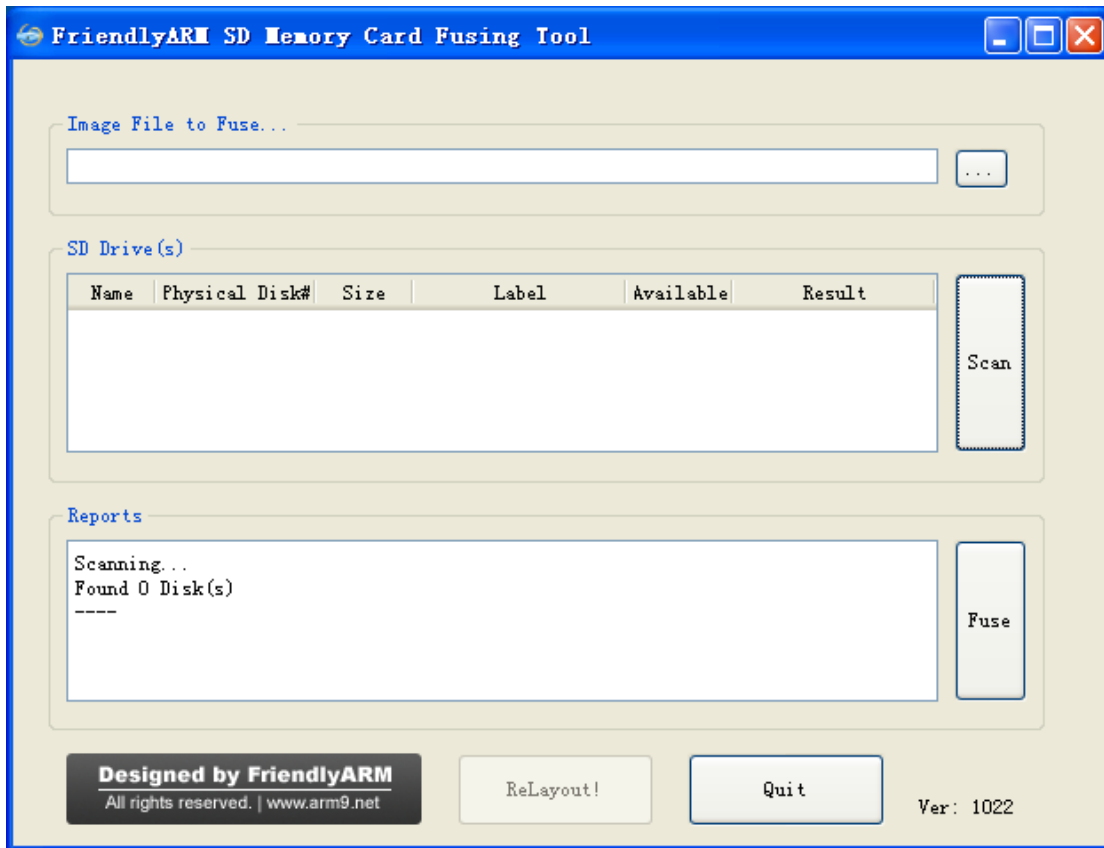
2.2 Burn BIOS in WindowsXP

Note: users complained that some notebook's integrated SD card reader cannot work properly with card burning or reading. So far we haven't encountered this issue and we suggest that you should try an external usb card reader in this case.

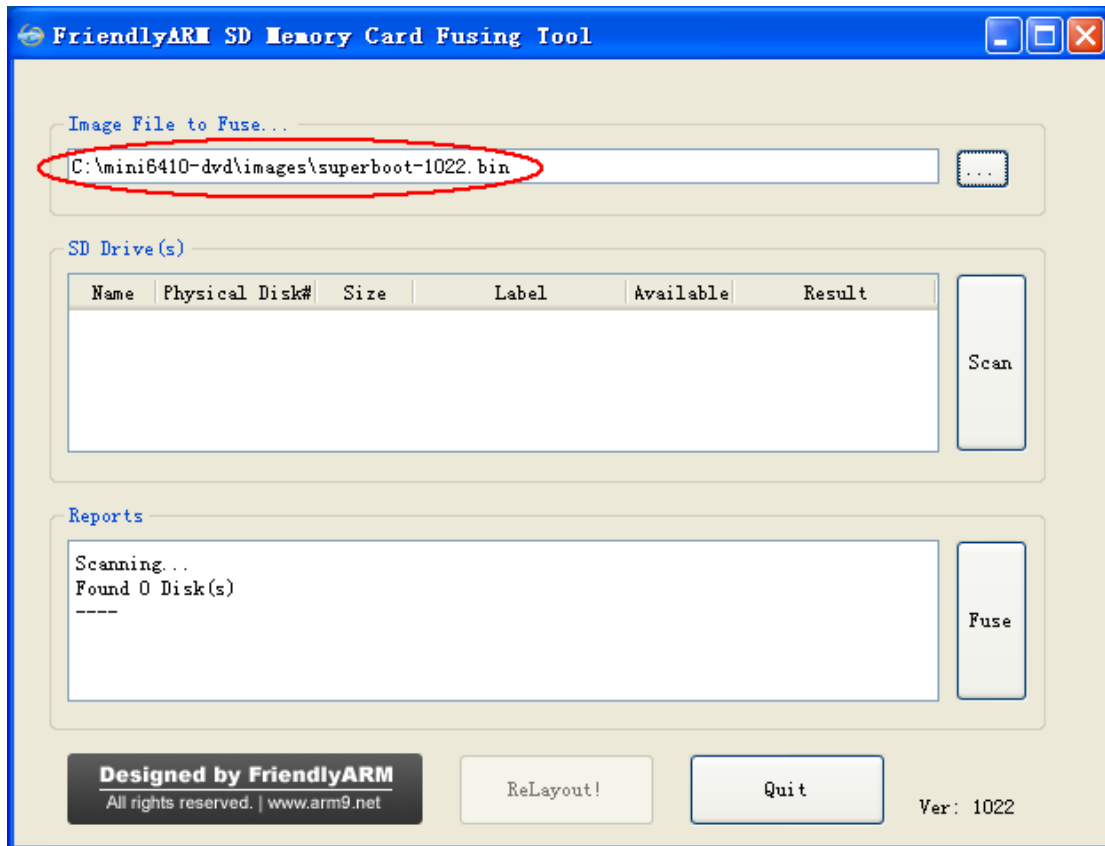
Our SD-Flasher.exe formats a 130M space for the bootloader therefore an SD card whose memory is less than 256M cannot work and we recommend using one whose memory is at least 4G.

Step1: launch the SD-Flasher.exe (under "\tools\"). Note: the "ReLayout" button is disabled.

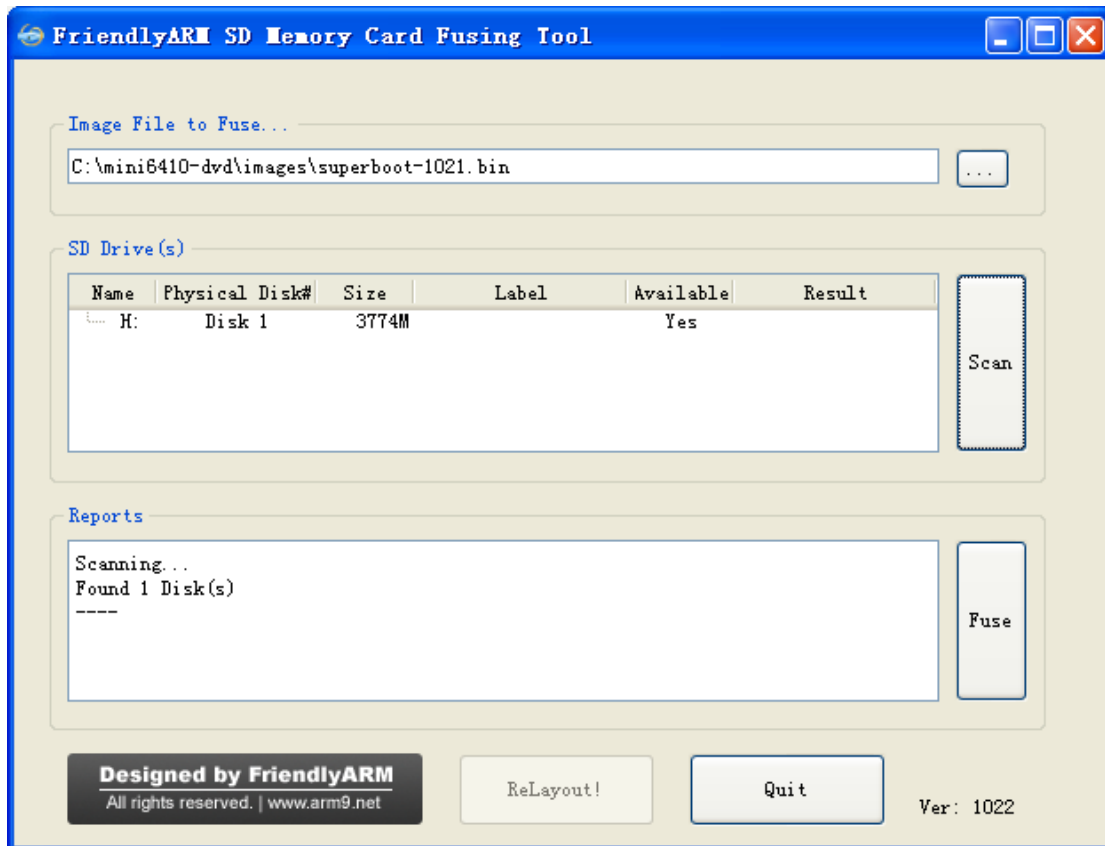
We set it purposely in WindowsXP.



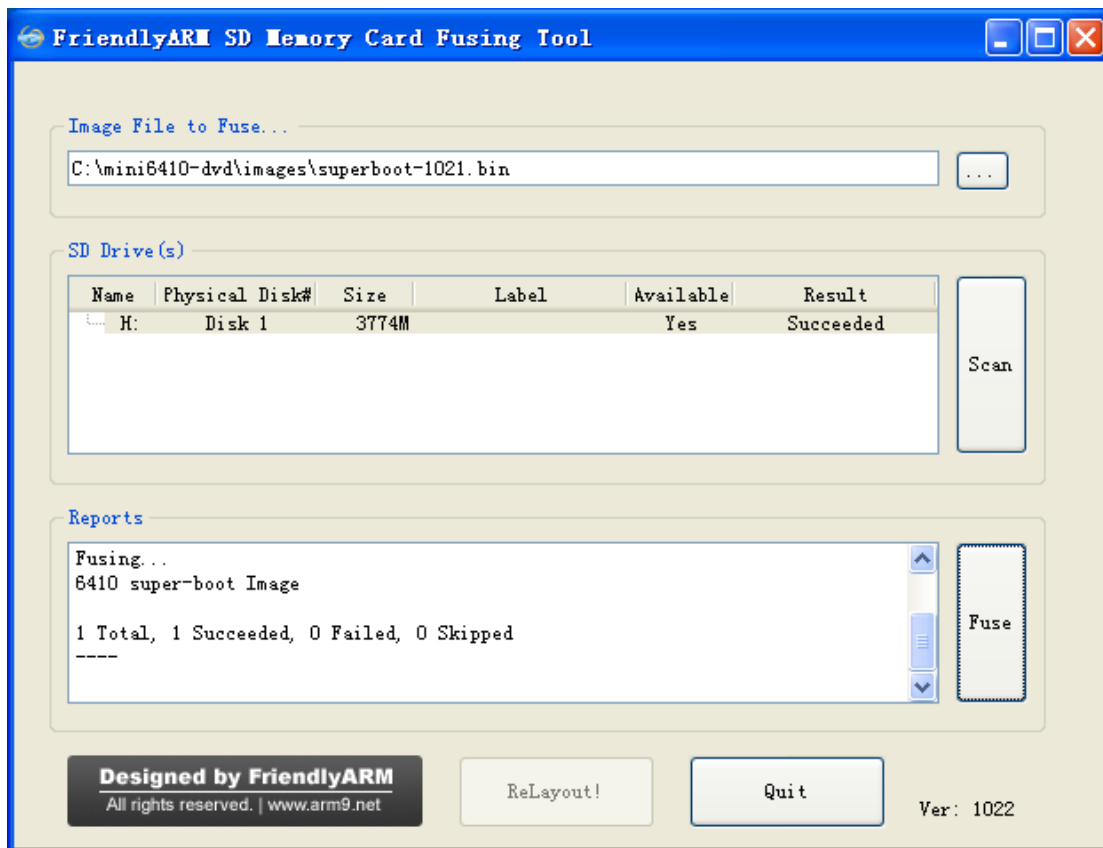
Step2: click on  to select Superboot



Step 3: insert a FAT32 SD card into your SD card socket (you can also use a USB card reader to connect to a PC) and click on “Scan”, all recognized SD cards will be listed.



Step 4: click on “Fuse”, Superboot will be burned into your SD card



The Superboot in your SD card is invisible. To verify it you can insert your SD card into your board's SD card socket and switch S2 to the "SDBOOT" mode, reboot your board and if LED1 is flashing it is indicating that your Superboot is functioning.

Note: if you use a TF card it might not work properly due to bad connection, so we suggest a common SD card be used.

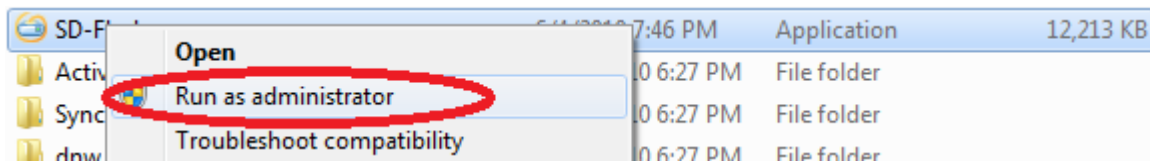
2.3 Burn BIOS in Windows7

Note: users complained that some notebook's integrated SD card reader cannot work properly with card burning or reading. So far we haven't encountered this issue and we

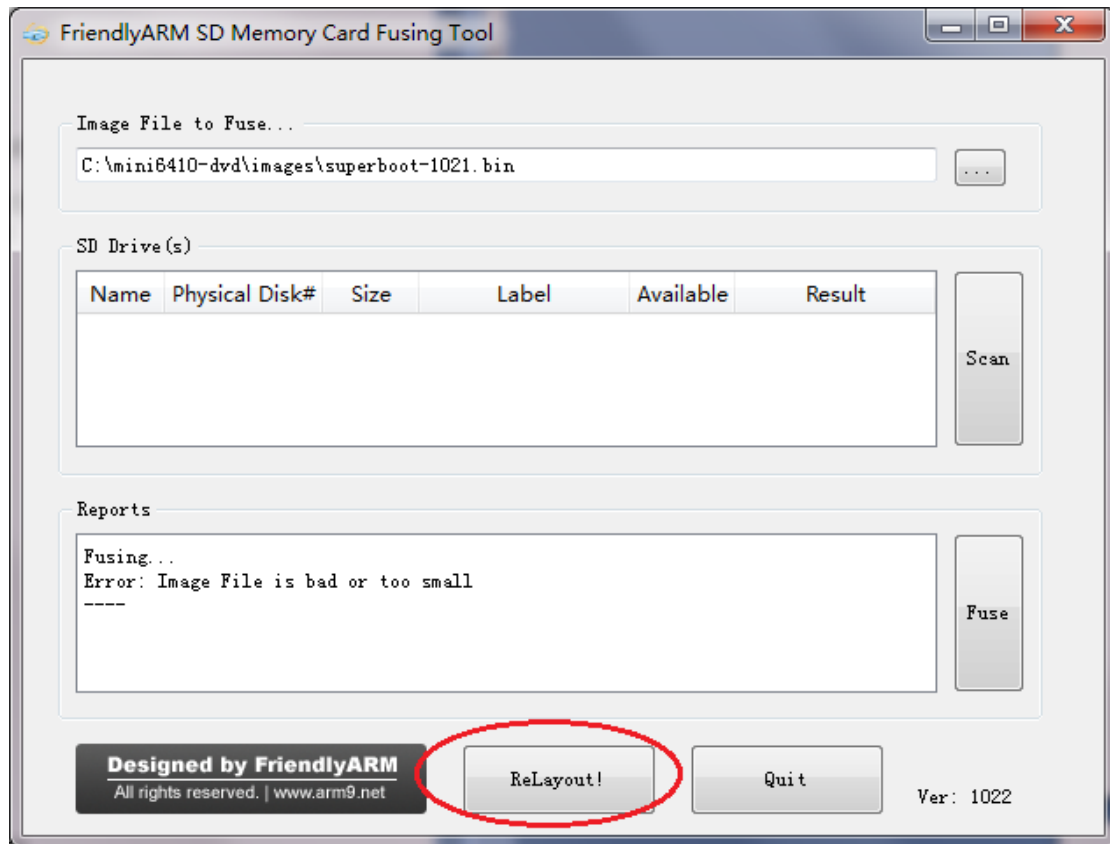
suggest that you should try an external usb card reader in this case.

Our SD-Flasher.exe formats a 130M space for the bootloader therefore an SD card whose memory is less than 256M cannot work and we recommend using one whose memory is at least 4G.

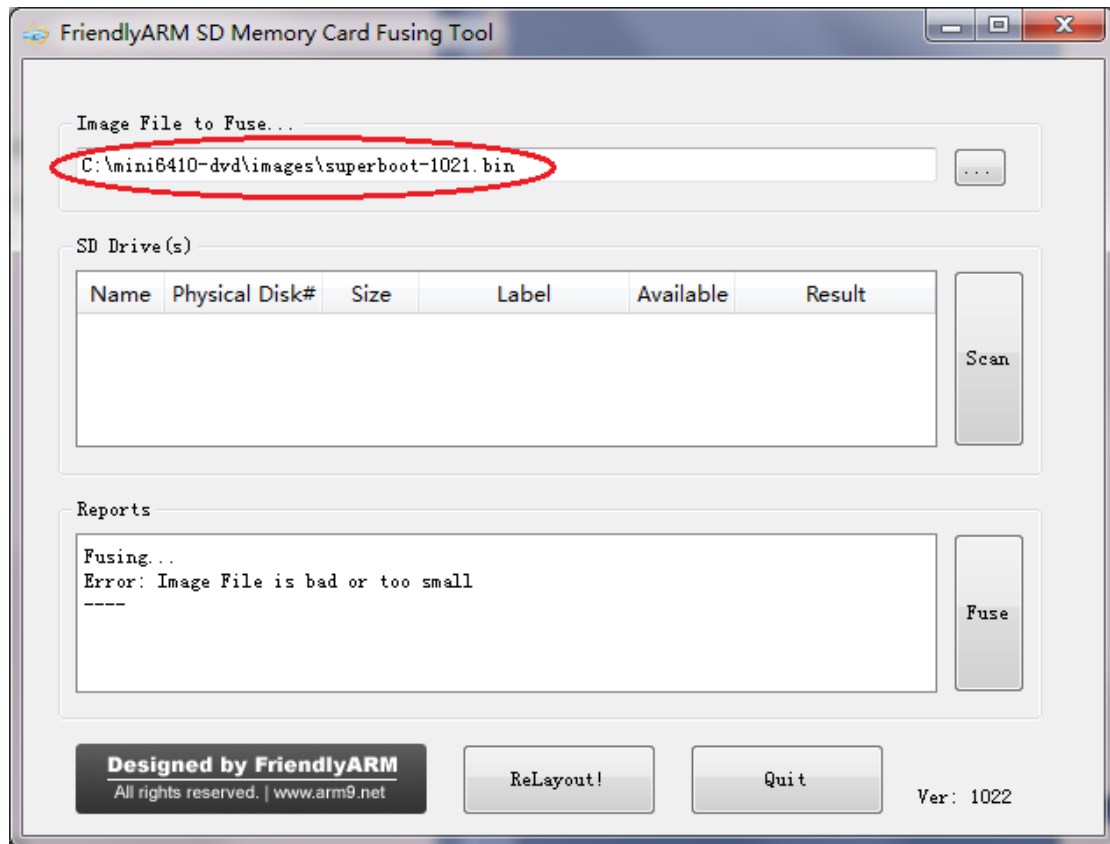
Step1: launch the SD-Flasher.exe (under “\tools\”). Note: you need to open it as “Administrator”



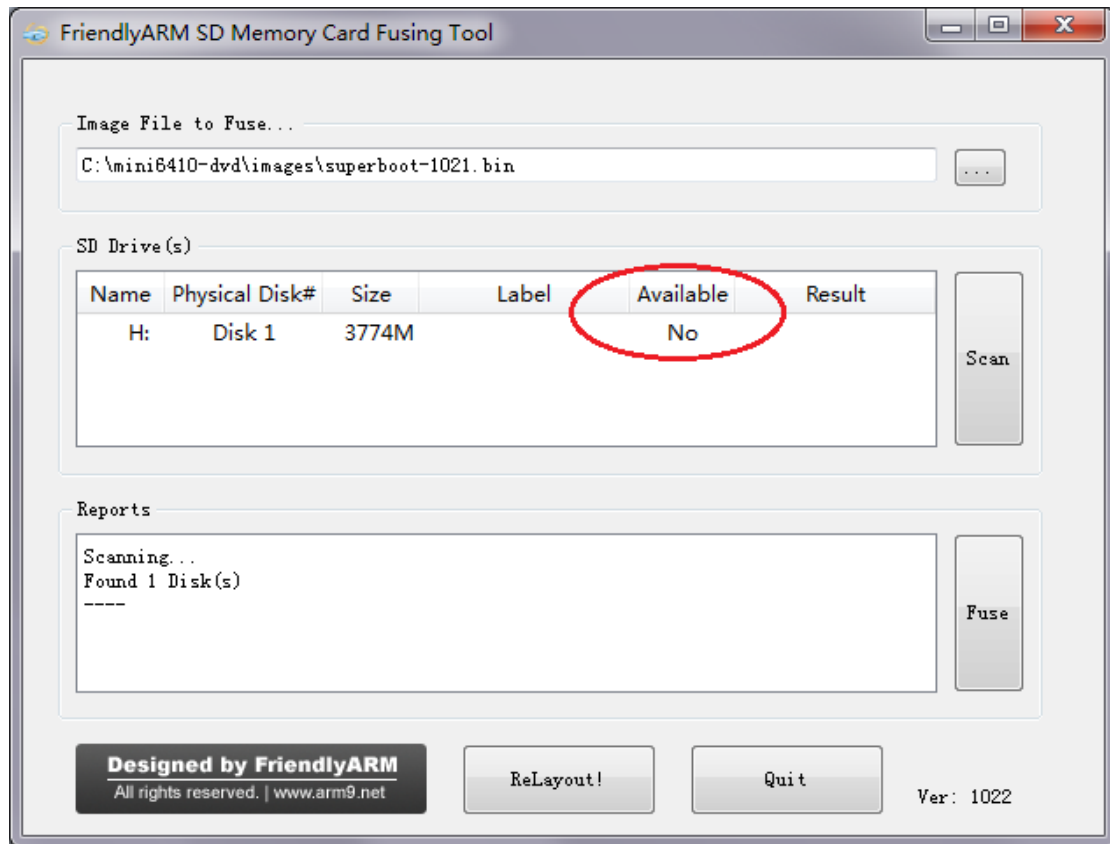
Below is the dialog you will see after it is started. Note: the “ReLayout” is enabled and we will format the SD card with this function.



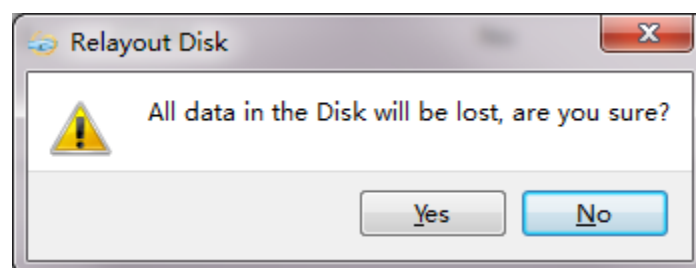
Step2: click on  to select Superboot



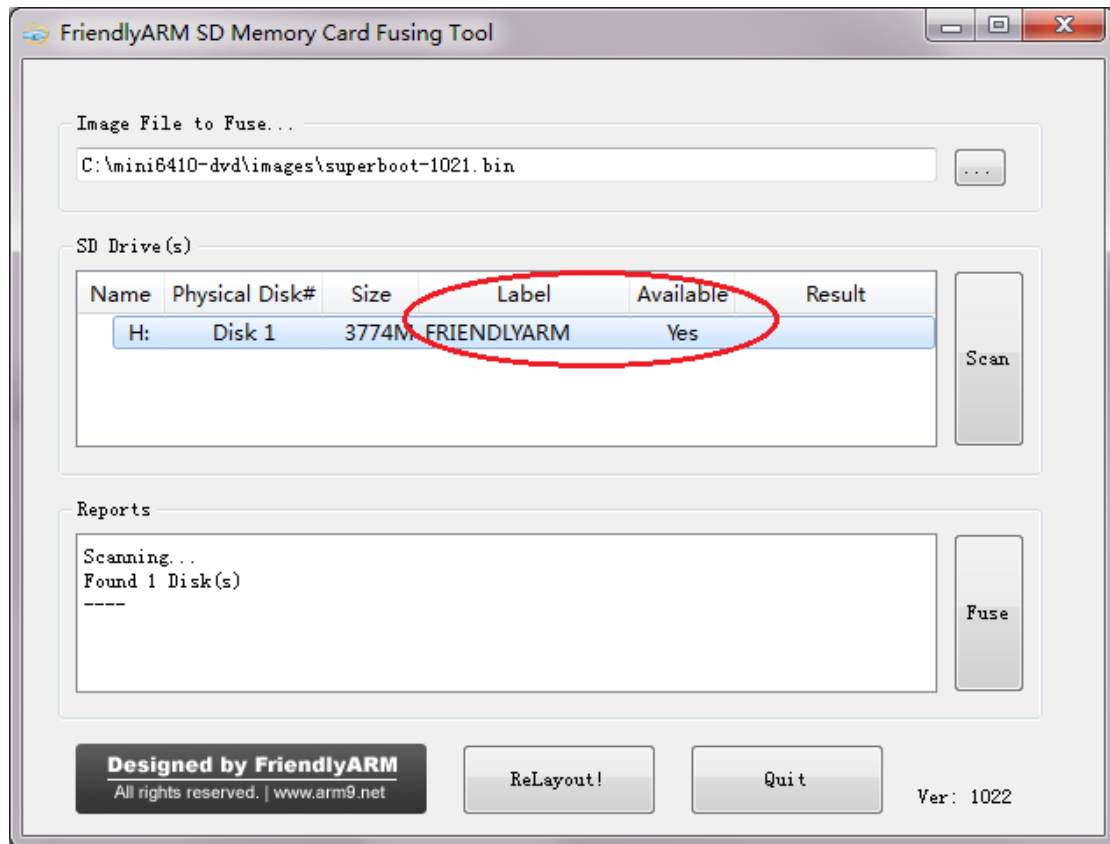
Step3: insert a FAT32 SD card into your SD card socket (you can also use a USB card reader to connect to a PC), backup your data in the card and click on “Scan”, all recognized SD cards will be listed. For now, the SD card cannot be burned (circled by red)



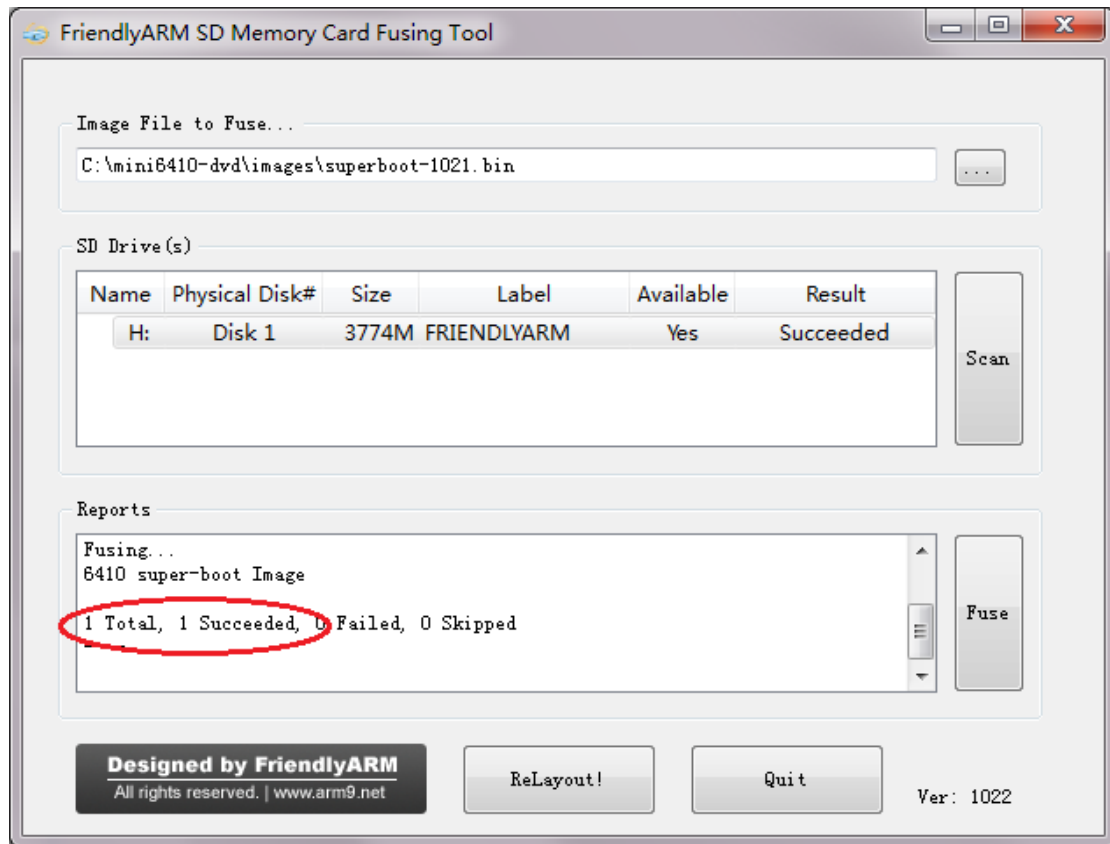
Step4: click on “ReLayout”, the following dialog will pop up prompting you that the data in your card will be lost. Just click on “Yes”



After formatting is done you will be directed back to the main menu. Click on “Scan”, you will see that a “FriendlyARM” section available.



Step5: click on “Fuse”, Superboot will be safely burned into the SD card. You can burn this card in WindowsXP without worrying about its FAT32 data being lost or damaged.



The Superboot in your SD card is invisible. To verify it you can insert your SD card into your board's SD card socket and switch S2 to the "SDBOOT" mode, reboot your board and if LED1 is flashing it is indicating that your Superboot is functioning.



3 Introduction to Superboot

When using Superboot as the SD card's bootloader to boot the board, users will enter the USB download mode in the following two situations:

1. "images\FriendlyARM.ini" doesn't exist in the SD card or its file name is not spelled correctly
2. When the ini file exists and "CheckOneButtons" is "Yes" users forget to press a key to continue the booting process.

In this mode the terminal will display the following menu and LED1 will be flashing continuously:



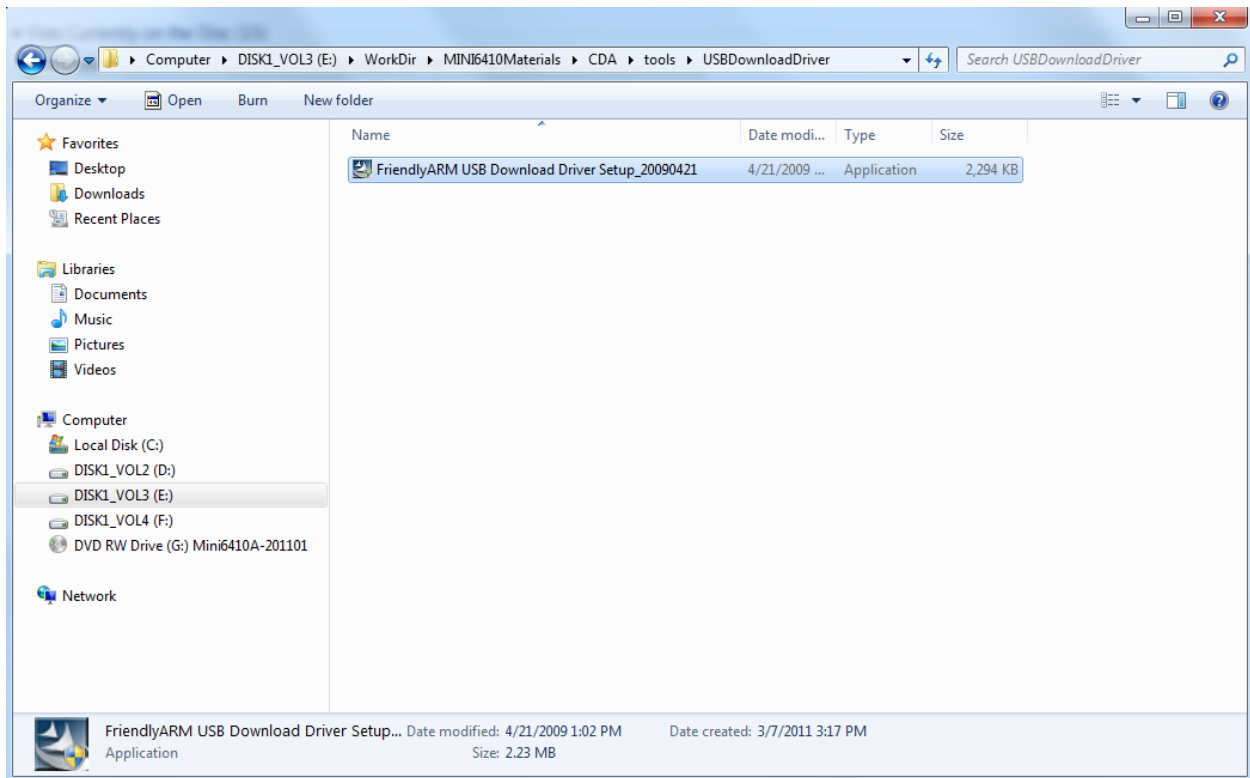
```
ttyS0 - HyperTerminal
File Edit View Call Transfer Help
Freeing init memory: 1404K
yaffs: dev is 32505858 name is "mtdblock2" rw
yaffs: passed flags ""
s3c6400_setup_sdhci_cfg_card: CTRL 2=c0004100, 3=00008080
mmc0: new high speed SDHC card at address 1234
mmcblk0: mmc0:1234 SA04G 3.68 GiB
  mmcblk0: p1
s3c6400_setup_sdhci_cfg_card: CTRL 2=c0004120, 3=80808080
##### FriendlyARM Superboot for 6410 #####
[f] Format the nand flash
[v] Download uboot.bin
[k] Download Linux/Android kernel
[y] Download root yaffs2 image
[u] Download root ubifs image
[a] Download Absolute User Application
[n] Download Nboot.nb0 for WinCE
[l] Download WinCE bootlogo
[w] Download WinCE NK.bin
[b] Boot the system
[s] Set the boot parameter of Linux
[d] Download and Run an Absolute User Application
[i] Version: 1051, RAM 256 MiB, NAND 256 MiB
Enter your Selection:_

Connected 0:11:17  Auto detect  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo
```

3.1 Install USB Download Driver

Installing this USB driver doesn't need to connect to a board. It is just for the PC system. It works in Windows7 but not in 64-bit Windows7.

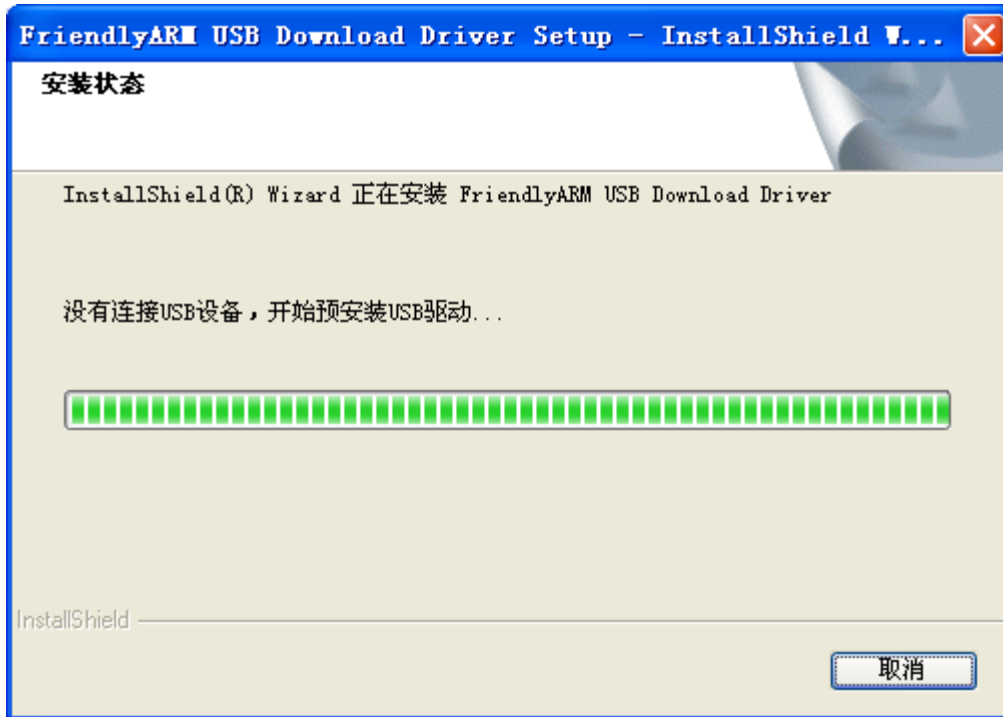
Open the shipped CD, double click on **“tools\USBDownloadDriver\FriendlyARM USB Download Driver Setup_20090421.exe”** **“WindowsPlatformTools\USBDownloadDriver\FriendlyARM USB Download Driver Setup_20090421.exe”** to start installation



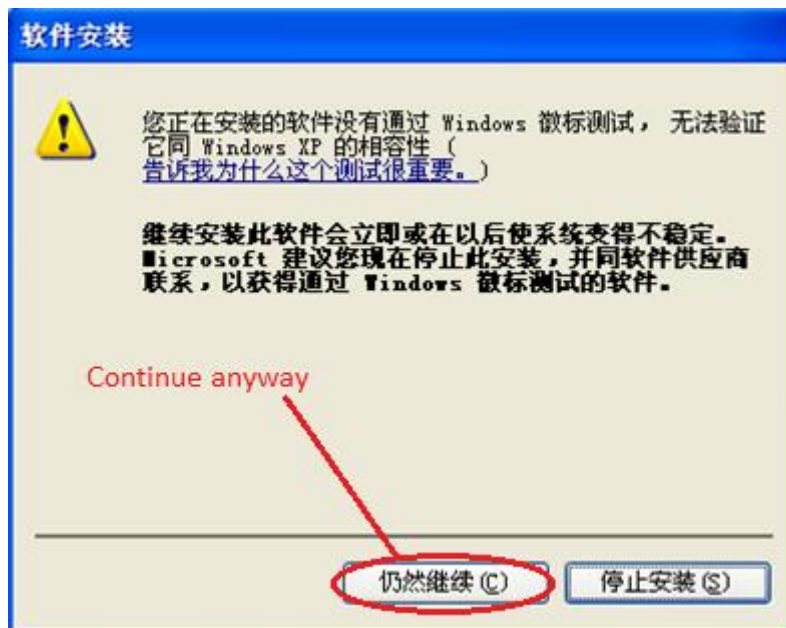
The following window will show up:



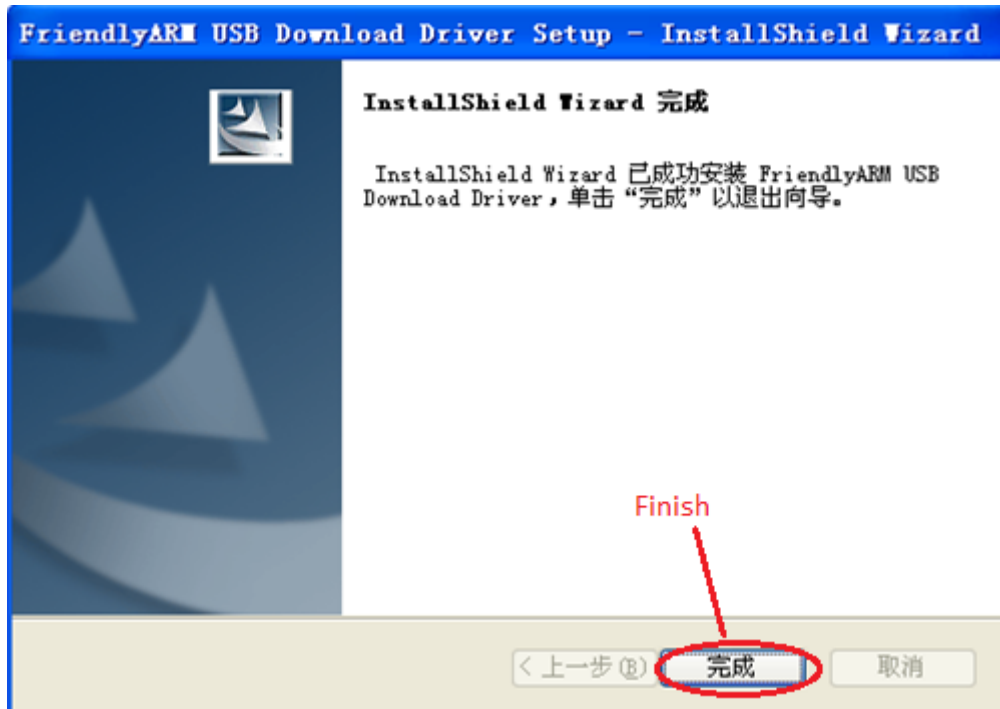
Click on (“Next”) the middle button



A warning message will pop up



Click on the (“continue anyway”) the left button to finish the installation.

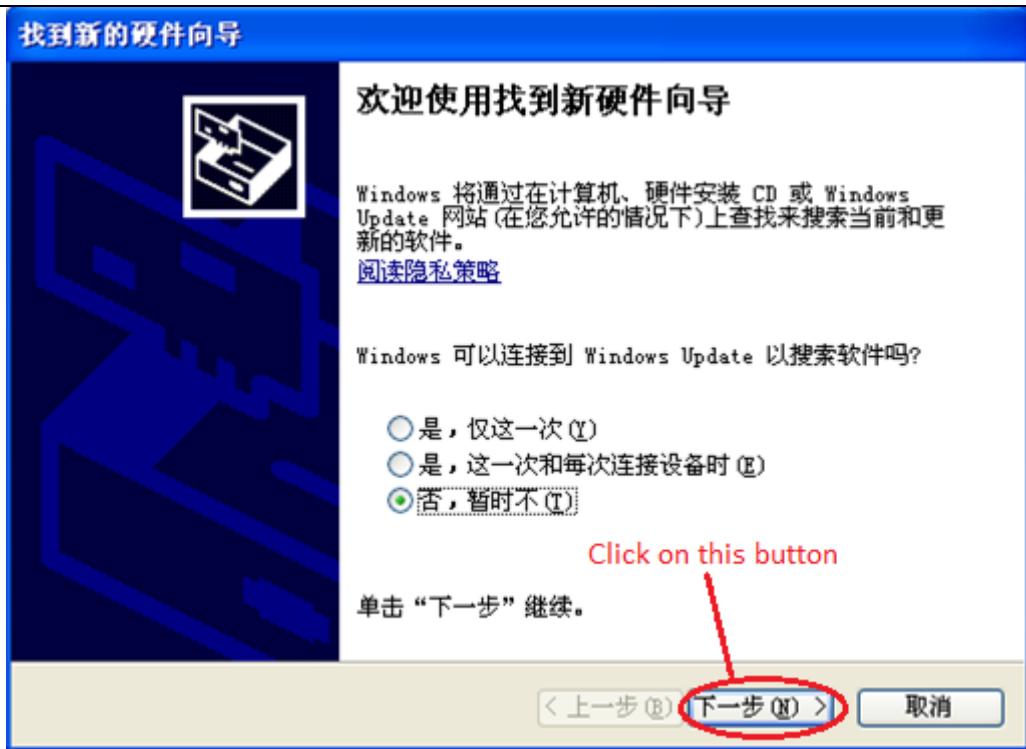


Now let's test the USB driver:

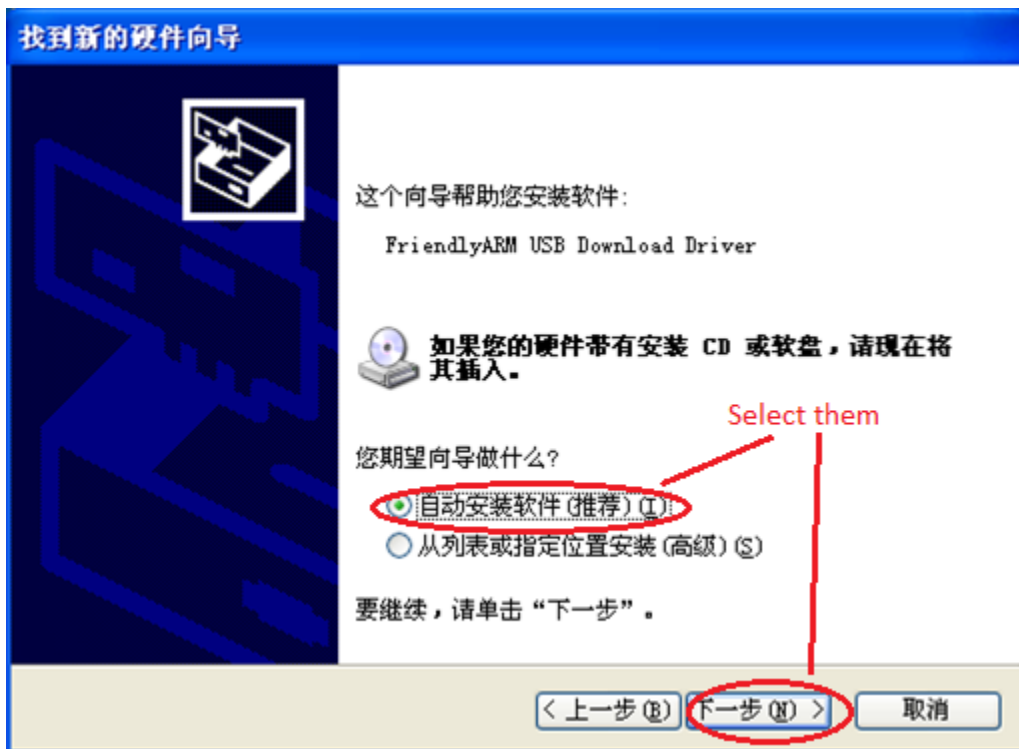
Connect the Mini6410 board to a host PC via a USB cable. Toggle the S2 switch to the “SDBOOT” side.

Turn on the S1 switch, if this is the first time you connect, Windows XP will prompt that a new USB device is found. Follow the steps below to install a USB driver

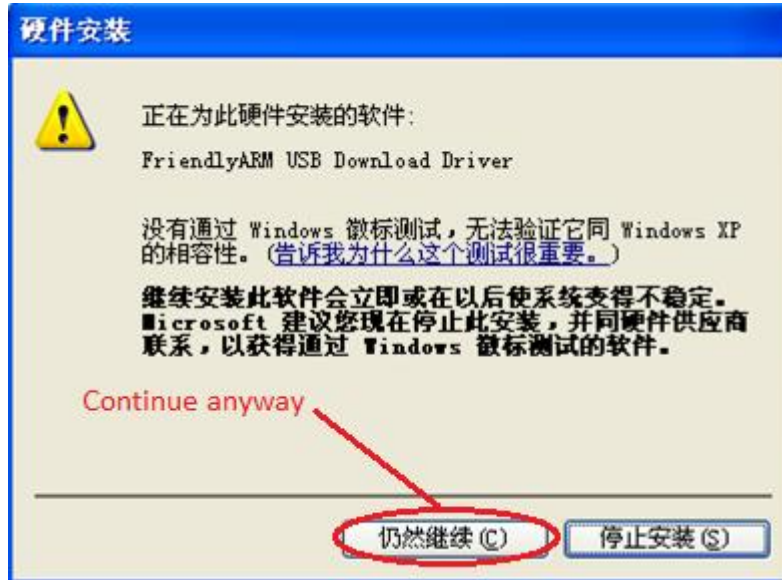
(1) After the following window pops up, check the third option and click on the “Next” button



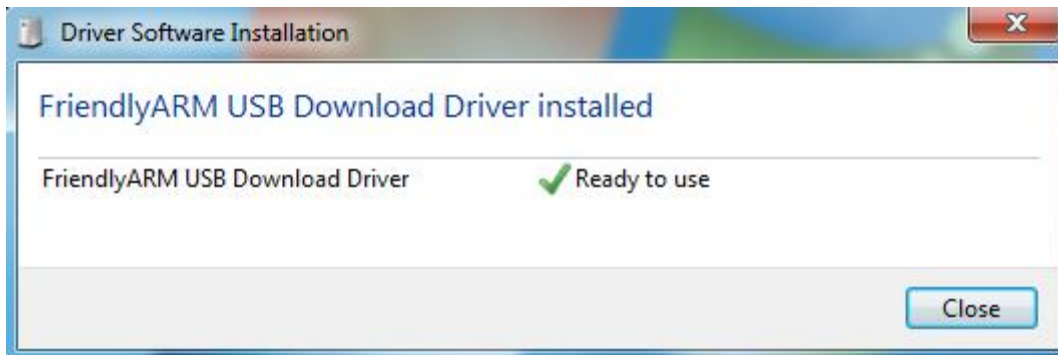
(2) On the window shown below, check the first option and click on the “Next” button



On the following popup window, click on the left button (“Continue anyway”).



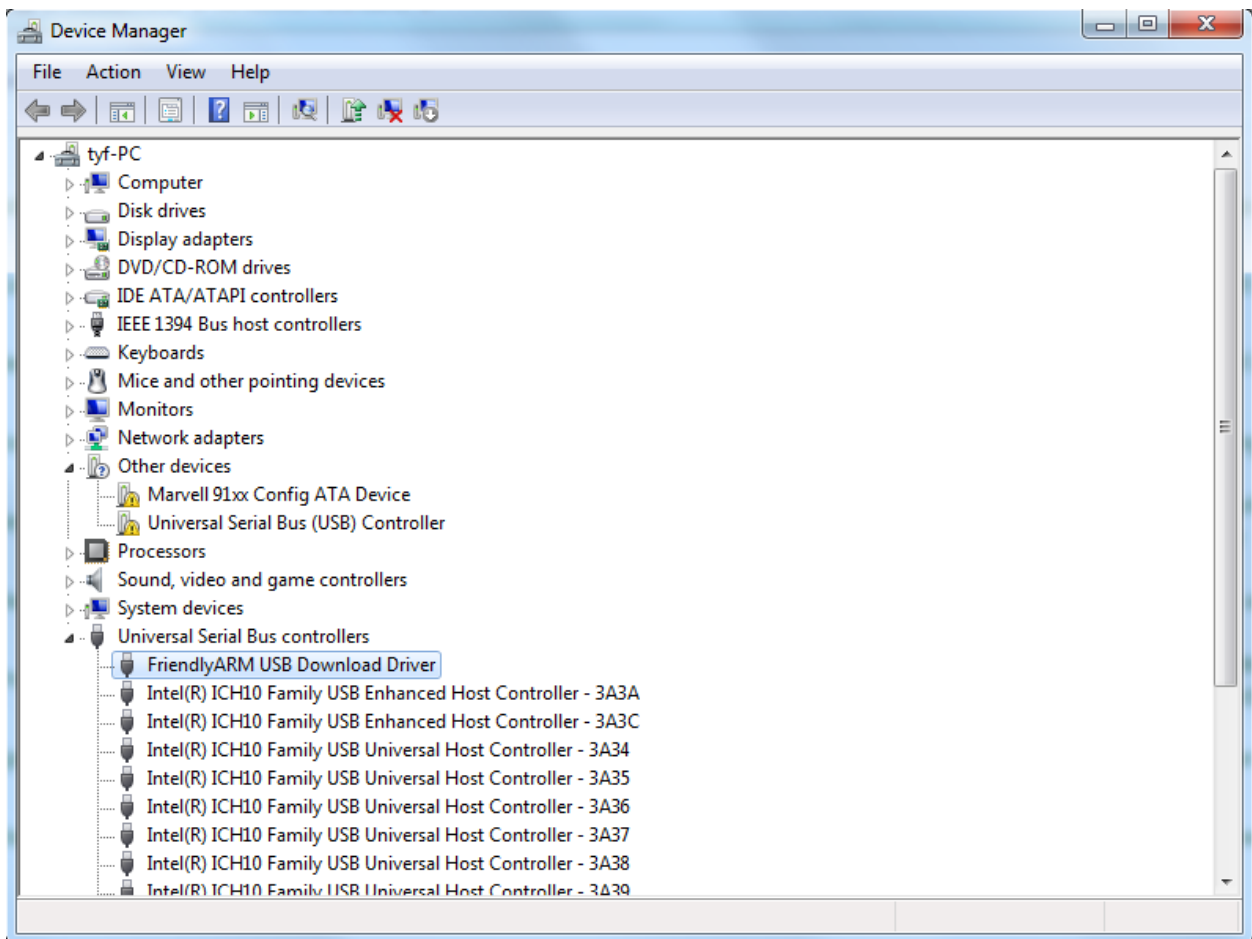
OK, our installation is done.



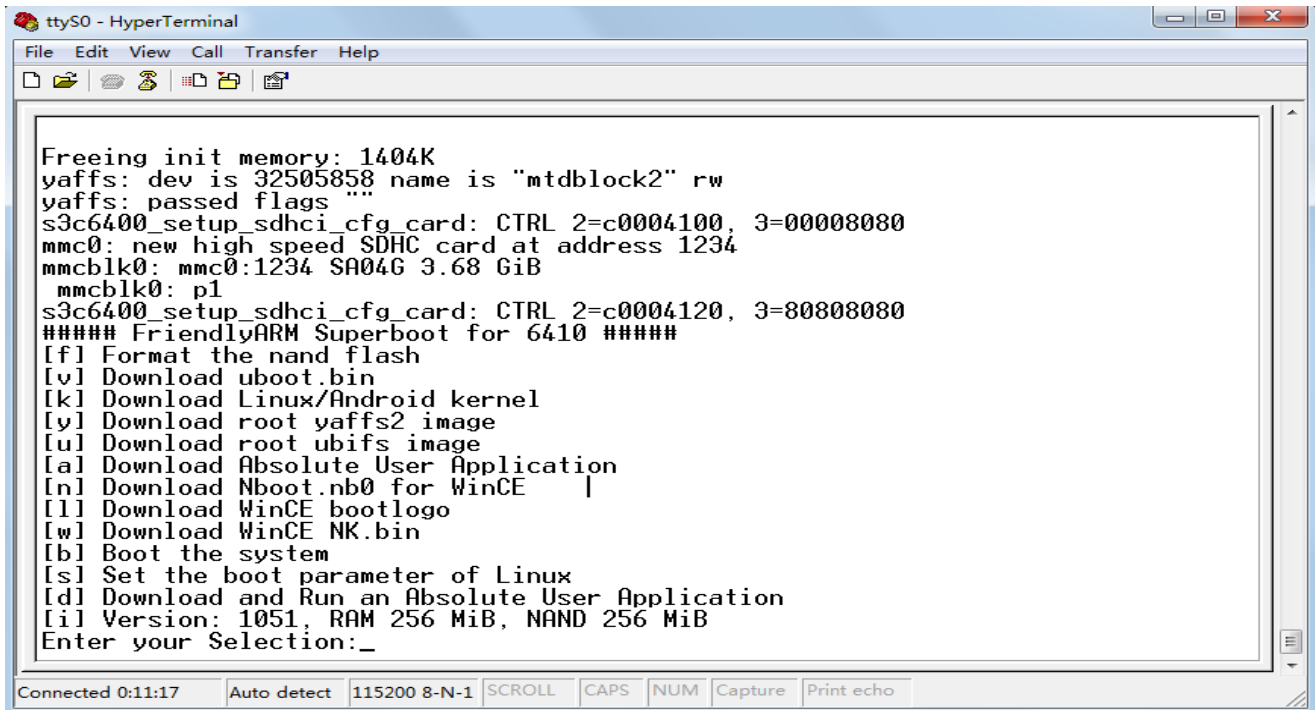
Open the CD, click on the dnw.exe, if you can see “USB:OK”, this means the installation is a success



In the device manager, you will see the installed USB driver information:



3.2 Superboot's Menu



```

Freeing init memory: 1404K
yaffs: dev is 32505858 name is "mtdblock2" rw
yaffs: passed flags ""
s3c6400_setup_sdhci_cfg_card: CTRL 2=c0004100, 3=00008080
mmc0: new high speed SDHC card at address 1234
mmcblk0: mmc0:1234 SA04G 3.68 GiB
mmcblk0: p1
s3c6400_setup_sdhci_cfg_card: CTRL 2=c0004120, 3=80808080
##### FriendlyARM Superboot for 6410 #####
[f] Format the nand flash
[v] Download uboot.bin
[k] Download Linux/Android kernel
[y] Download root yaffs2 image
[u] Download root ubifs image
[a] Download Absolute User Application
[n] Download Nboot.nb0 for WinCE
[l] Download WinCE bootlogo
[w] Download WinCE NK.bin
[b] Boot the system
[s] Set the boot parameter of Linux
[d] Download and Run an Absolute User Application
[i] Version: 1051, RAM 256 MiB, NAND 256 MiB
Enter your Selection:_

```

Note: the above menu may subject to changes.

Item[f]: Format the Nand Flash. This command will delete all the data in it

Item[v]: Download a linux bootloader e.g. U-boot

Item[k]: Download a linux/Android kernel

Item[y]: Download an image of the yaffs2 file system

Item[u]: Download an image of the UBIFS file system

Item[a]: Download Absolute User Program (standalone program), usually it is a bin file, such as uCos2.

Item[n]: Download a WinCE's bootloader Nboot

Item[l]: Download a WinCE's boot logo



Item[w]: Download an image of WinCE NK.bin

Item[b]: Boot system, if the board is installed with either WinCE or Linux, it will load it

Item[s]: Set boot parameters

Item[d]: Download and run an Absolute User Program (standalone program)

Item[i]: Display Superboot's version and the NAND Flash's memory



4 Play with Superboot

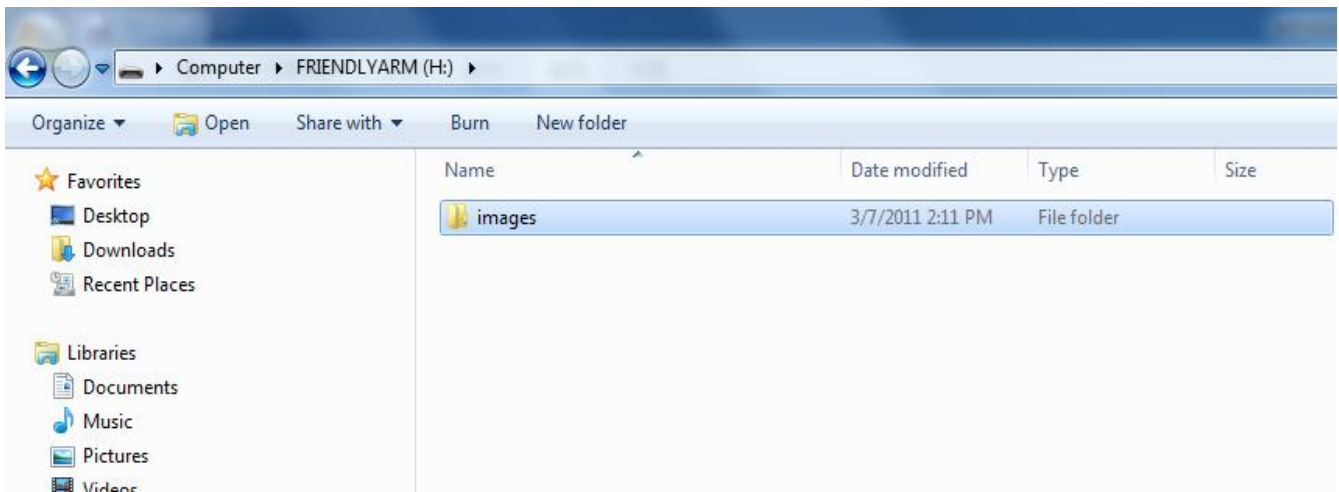
To run or install systems from an SD card which should have a Superboot some system files need to be copied to it. One of them is a configuration file named “FriendlyARM.ini”. Let us experience rapid installation and running via some basic steps. The following steps are based on 4.3-inch systems. You need to adjust your steps for other systems accordingly.

4.1 Install WindowsCE

Note: by default all boards are preinstalled with Linux. We will now install WindowsCE6.

Step1: copy the whole “images” directory into your SD card, double click to open the “images\FriendlyARM.ini” file, change “OS=Linux” to “OS=CE6”, save it and insert this card into your Mini6410 board.

Note: after the whole images directory is copied into the SD card’s root directory it will look like this:



Step2: toggle S2 to the “SDBOOT” mode and insert your SD card

Step3: power on your board, you will hear a beep from the buzzer and LED4 will be flashing

Step4: within few seconds you will notice that LED3, 2 and 1 begin to flash one by one until two beeps are heard and then all LEDs are on and round robin flashing. This indicates that burning is done. The whole process takes less than 20 seconds.

Step5: toggle S2 to the “NAND” mode, reboot the system and you will see WindowsCE is loading.

4.2 Install Android

Android is now very hot and you may want to give it a shot! Let us try it:

Step1: copy the whole “images” directory into your SD card, double click to open the “images\FriendlyARM.ini” file.

Step2: change “OS=CE6” to “OS=Android”, save it and insert this card into your Mini6410



board. Toggle S2 to the “SDBOOT” mode

Step3: power on your board, the installation process begins with a beep and ends with two beeps. The whole process takes less than 60 seconds.

Step4: toggle S2 to the “NAND” mode, reboot the system and you will see Android is loading.

Note: if you hear continuous beeps or see all four LEDs flash simultaneously it is an indication that your spelling in the ini file may be wrong.

4.3 Restore Linux

Step1: copy the whole “images” directory into your SD card, double click to open the “images\FriendlyARM.ini” file.

Step2: change “OS=Android” to “OS=Linux”, save it and insert this card into your Mini6410 board. Toggle S2 to the “SDBOOT” mode

Step3: power on your board, the installation process begins with a beep and ends with two beeps. Our Linux includes Qtopia-2.2.0, Qtopia4, QtE-4.7.0 and SMPlayer therefore the image file is relatively large. The whole process takes about 1 minute.

Step4: toggle S2 to the “NAND” mode, reboot the system and you will see Linux is loading.

Note: if you hear continuous beeps or see all four LEDs flash simultaneously it is an indication that your spelling in the ini file may be wrong.



4.4 Run Ubuntu

Step1: open the “images\FriendlyARM.ini” file.

Step2: change “OS=Linux” to “OS=Ubuntu” and “Action=Install” to “Action=Run”, save it and insert this card into your Mini6410 board. Toggle S2 to the “SDBOOT” mode

Step3: power on your board, you will be able to play with Ubuntu. You can even modify its configurations and will not lose it unless you lose your card. And this will not damage the NAND Flash data either unless there are actions to operate the NAND Flash.

Note: if you hear continuous beeps or see all four LEDs flash simultaneously it is an indication that your spelling in the ini file may be wrong.

4.5 About FriendlyARM.ini

The “FriendlyARM.ini” file is a configuration file for system installation and running from SD card. Below is a table which lists the most commonly used setting options:

Items	Note: different systems may have different default settings
CheckOneButton	=“yes”, users need to press any button before the system can reboot completely after the system is reset, = “No”, system will reboot completely after it is reset. For mass burning this item is usually set to “No”
Action	Set actions: Install/Run/Null Install – Install to the NAND Flash Run – Run from SD card Null – No action



	The default option is "Install"
OS	Operating system to be loaded: Linux/WindowsCE6/Ubuntu/Android/UserBin ; "UserBin" means standalone programs or single file image such as uCos2 and Rt-Thread. The default option is "Linux"
VerifyNandWrite	= "yes", system will verify after burning is done. This is more reliable ; = "No", system will not verify, this takes less time. The default option is "No"
StatusType	Status of the burning process: "LED", "Beeper" and "LED Beeper" The default option is "LED Beeper"
Linux' image settings, they can contain directories, "/" or "\"	
Linux-BootLoader	Bootloader's image: e.g. Linux-BootLoader=Linux/u-boot_nand-ram256.bin (by default)
Linux-Kernel	Linux kernel's image: e.g. Linux-BootLoader=Linux/zImage_n43 (by default)
Linux-CommandLine	Linux booting parameters, different file systems require differed settings for YAFFS2 recommended setting (this is also the default setting): Linux-CommandLine = root=/dev/mtdblock2 rootfstype=yaffs2 init=/linuxrc console=ttySAC0,115200 for UBIFS, recommended setting: Linux-CommandLine = root=ubi0:FriendlyARM-root ubi.mtd=2 rootfstype=ubifs init=/linuxrc console=ttySAC0,115200 running from SD card, you can go with the default setting
Linux-RootFs-InstallImage	Image for installation: yaffs2 / UBIFS We define it as follows: The "img" extension is for yaffs2 images The "ubi" extension is for UBIFS images The "ext3" extension is for EXT3 images (only for running from the SD card) e.g. Linux-RootFs-RunImage=Linux/root-qt4-topia-qt4.img (by default)
Linux-RootFs-RunImage	Image to be run from the SD card, e.g. Linux-RootFs-RunImage=Linux/root-qt4-topia-qt4.ext3 (by default)
WindowsCE6's image settings, they can contain directories, "/" or "\"	
WindowsCE6-Bootloader	Bootloader's image: e.g.WindowsCE6\NBOOT_N43-RAM256.nb0 (by default)



WindowsCE6-BootLogo	Boot logo of WindowsCE6. This logo will be burned into the NAND Flash. It can be a bmp file and the max size is 2M e.g. WindowsCE6-BootLogo=WinowsCE6\BootLogo.bmp (by default)
WindowsCE6-InstallImage	Image for installation, it should be an NK.bin file e.g. : WindowsCE6-InstallImage=WindowsCE6\NK_N43.bin (by default)
WindowsCE6-RunImage	Image to be run from the SD card,it should to be an Nk.bin file, e.g. :WindowsCE6-RunImage=WindowsCE6\NK_N43.bin (by default)
Ubuntu's image settings, they can contain directories, "/" or "\"	
Note: users can burn a Ubuntu system as UBIFS into the NAND Flash or run it as ext3 from the SD card.	
Ubuntu-BootLoader	Bootloader's image e.g. Linux-BootLoader=Linux/u-boot_nand-ram256.bin (by default)
Ubuntu-Kernel	Kernel image e.g. Ubuntu-BootLoader=Ubuntu/uzImage_N43 (by default)
Ubuntu-CommandLine	Ubuntu's booting parameters, different file systems require differed settings <ul style="list-style-type: none"> ● for UBIFS, recommended setting : Linux-CommandLine = root=ubi0:FriendlyARM-root ubi.mtd=2 rootfstype=ubifs init=/linuxrc console=ttySAC0,115200 (by default) ● To run from the SD card, users can go with the default setting
Ubuntu-RootFs-RunImage	Image to be run from the SD card e.g. Ubuntu-RootFs-RunImage = Ubuntu/rootfs_ubuntu.ext3 (by default)
Android's image settings, they can contain directories, "/" or "\"	
Android-BootLoader	Bootloader's image e.g. Android-BootLoader=Android/ u-boot_nand-ram256.bin (by default)
Android-Kernel	Kernel image, e.g. Android-BootLoader=Android/azImage_N43 (by default)
Android-CommandLine	Android's booting parameters, different file systems require differed settings <ul style="list-style-type: none"> for yaffs2 recommended setting: Android-CommandLine = root=/dev/mtdblock2 rootfstype=yaffs2 init=/linuxrc console=ttySAC0,115200 for UBIFS recommended setting: Android-CommandLine = root=ubi0:FriendlyARM-root ubi.mtd=2 rootfstype=ubifs init=/linuxrc console=ttySAC0,115200



	To run from the SD card, users can go with the default setting
Android-RootFs-InstallImage	Image for installation: yaffs2/UBIFS We define it as follows: The “img” extension is for yaffs2 images The “ubi” extension is for UBIFS images The “ext3” extension is for EXT3 images (only for running from the SD card) e.g. Android-RootFs-InstallImage = Android/rootfs_android.ubi (by default)
Android-RootFs-RunImage	Image to be run from the SD card e. g. Android-RootFs-RunImage = Android/rootfs_android.ext3 (by default)
Standalone program’s image settings, they can contain directories, “/” or “\”	
Note: in general standalone programs should have a RAM address for execution	
UserBin-Image	Standalone program’s image, it can be a bin or nb0 file; when “Action” is “Install”, it will be burned into the initial address of the NAND Flash’s Block0.
UserBin-StartAddress	When “Action” is “Run” the image will be loaded to the specified address of RAM

Notes:

1. Statements after “#” will not be executed by Superboot. Actually any character except key words can be used to comment. Using “#” is just widely accepted
2. To prevent our Superboot from being illegally copied we make it a rule that the first line of the ini file cannot be edited or deleted. It is:

#This line cannot be removed. by FriendlyARM(www.arm9.net)

Note: no space or any other character after the last “)” is allowed