

KIT-S5PC100 Use manual

Ver 1.0

Date: 2010-01-06



Boardcon Embedded design



Rev	Date	Description
1.0	2009-12-21	Initial Release Vesion



Index

KIT-S5PC100 USE MANUAL	1
CHAPTER I OVERVIEW	4
1.1 <i>Introduction</i>	4
1.2 <i>Accessories</i>	5
CHAPTER II HARDWARE INFORMATION.....	6
2.1 <i>Block diagram</i>	6
2.2 <i>CPU</i>	6
2.3 <i>MEMORY</i>	8
2.4 <i>Display</i>	9
2.5 <i>WIFI</i>	10
2.6 <i>Audio Codec</i>	10
2.7 <i>Power Supply</i>	10
2.8 <i>Accelerometer</i>	10
2.9 <i>Touch switch (Sound volume up/down)</i>	10
CHAPTER III ANDROID SYSTEM DEVELOPMENT.....	11
3.1 <i>Build Development environment</i>	11
3.2 <i>System Compiler</i>	12
CHAPTER IV UPDATE THE IMAGE.....	14

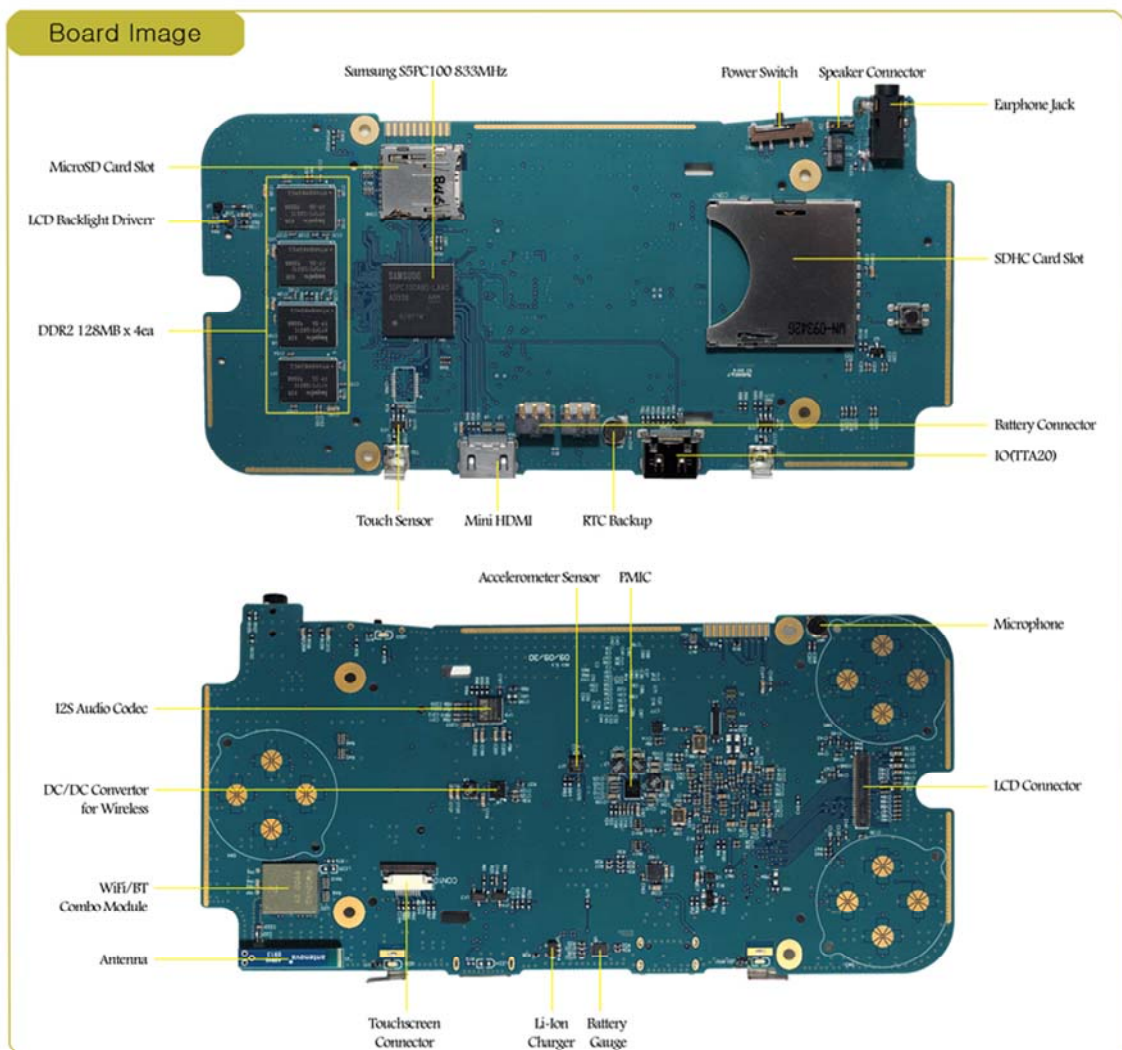


Chaper I Overview

1.1 Introduction

Boardcon KIT-S5PC100 Evaluation Board is a compact board using S5PC100 microprocessor. It takes full features of this processor and supports 512MB DDR2 memory. The board has exposed many other hardware interfaces including RS232 serial port, LCD/TSP, HDMI, SD/MMC, TF-card interface, keyboard, WiFi, The board boot the system from SD card. It is able to support Android OS and provided the source code for the Linux2.6.29 BSP and Android 1.5-R3.

Boardcon KIT-S5PC100 Evaluation Kit includes the KIT-S5PC100 evaluation board and all necessary accessories to help users start their design of multimedia applications. The board is preloaded with Android OS in SD card. User can display the subsystem using a 3.5" TFT LCD and Touch screen. Along with the kit, Boardcon provides user manual, schematic drawing and datasheet documents to help customers better understand and use the kit.



Pic 1.1 KIT-S5PC100 evaluation board



1.2 Accessories

The accessories of the KIT-S5PC100 Evaluation Kit are as following:

- 1 KIT-S5PC100 boardcon
- 1 Debug board
- 1 USB cable
- 1 Battery
- 1 Schematics and the Android source code
- 2GB MicroSD card
- 2GB SDHC Card



Pic 1.2 the accessories



Chapter II Hardware information

2.1 Block diagram

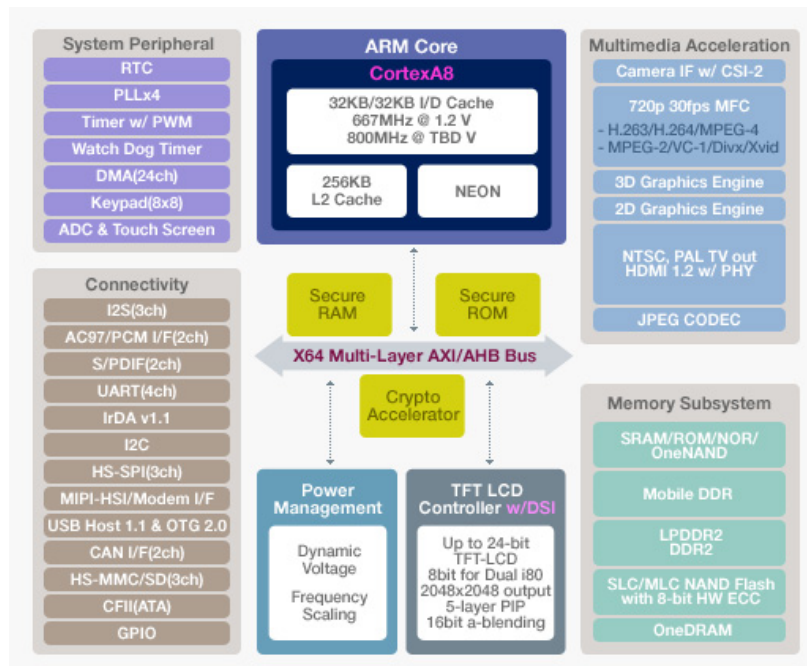
WiFi 802.11 b/g Bluetooth V2.x	HDTV 720P	3.5" TFT-LCD 320 X 480 LED Backlit	Capacitive Touch screen
Charger Circuit	Power Management IC	CPU SAMSUNG S5PC100	T-Flash (Micro) SD Card Slot
Li-Ion Rechargeable Battery 1300mA	Battery Gauge		Standard SD Card slot
High Speed USB	Keypad Accelerometer	1Gb 1Gb 1Gb 1Gb 512MB DRAM(DDR2)	Audio Codec Earphone Speaker/MIC

Pic 2.1 Block diagram

2.2 CPU

S5PC100 is a 32-bit RISC low power, high performance micro-processor solution for mobile phones and general applications, and integrates 833Mhz CortexA8 which implements the ARM architecture V7-A with numerous peripherals to support. To provide optimized H/W performance for the 3G & 3.5G communication services, S5PC100 adopts a 64-bit internal bus architecture and includes many powerful hardware accelerators for tasks such as motion video processing, display control and scaling. Integrated Multi Format Codec (MFC) supports encoding and decoding of MPEG4, H.263, H.264 and decoding of MPEG2, VC1, Divx/Xvid. This H/W Encoder/Decoder supports real-time video conferencing and Analog TV out for NTSC and PAL mode, HDMI output for HDTV. The S5PC100 has an optimized interface to external memory capable of sustaining the demanding memory bandwidth required in high-end communication services. The memory system has Flash/ROM external memory ports for parallel access and DRAM port for high bandwidth. DRAM port can be configured to support mobile DDR, LPDDR2, and DDR2. Flash/ROM Port supports SLC/MLC NAND Flash, NOR Flash, OneNAND and ROM type external memory.

Overview To reduce total system cost and enhance overall functionality, S5PC100 includes many hardware peripherals such as TFT 24-bit true color LCD controller, Camera Interface, MIPI DSI, CSI-2 and HSI, System Manager for power management, CF+/ATA I/F, 4-channel UART, 24-channel DMA, 4- channel Timers, configurable General I/O Ports, 3-ch IIS, 2-ch S/PDIF, 2-ch CAN bus, IIC bus interface, 3-ch HS-SPI, USB Host 1.1 operating at full speed(12Mbps), USB OTG 2.0 operating at high speed (480Mbps), 3-ch SD Host & High Speed Multi-Media Card Interface and PLLs for clock generation.



Features

- CortexA8 based CPU Subsystem with NEON
- 32/32KB I/D Cache, 256KB L2 Cache
- 833MHz Operating Frequency
- 64-bit Multi-layer bus architecture
- Advanced power management for mobile applications
- ROM for secure booting and RAM for security function
- 8-bit ITU 601/656 Camera Interface up to 8M pixel for scaled and 16M pixel for un-scaled resolution
- Multi Format CODEC provides encoding and decoding of MPEG-4/H.263/H.264 up to 30fps@HD(720p) and decoding of MPEG-2/VC1/Divx/Xvid video up to 30fps@HD(720p)
- JPEG codec support up to 30Mpixels/s
- 3D Graphics Acceleration with Programmable Shader up to 10M triangles/s (Transform only)
- 2D Graphics Acceleration with BitBlit and Rotation, up to 40Mpixels/s
- 1/2/4/8 bpp palletized or 8/16/24bpp non-palletized Color-TFT support up to 2048x2048
- TV-out for NTSC and PAL mode and HDMI 1.2 interface support with PHY
- MIPI-HSI, MIPI-DSI and MIPI-CSI interface support
- channel AC-97 audio codec interface, 2-ch PCM serial audio interface, and 3-channel 24-bit I2S interface support (5.1ch support)
- channel S/PDIF interface support for digital audio
- channel I2C interface (up to 400KHz) support including 1-channel for HDMI
- channel HS-SPI, up to 52Mbps
- channel UART including 4Mbps port for Bluetooth 2.0 and IrDA port for SIR/MIR/FIR
- On-chip USB 2.0 OTG supporting high speed (480Mbps, on-chip transceiver)
- On-chip USB 1.1 Host supporting full speed (12Mbps, on-chip transceiver)
- Asynchronous direct Modem Interface support including 16KB DPRAM
- 3-channel SD/SDIO/HS-MMC interface support including CE-ATA
- CF version 3.0 interface support for HDD



- 24-channel DMA controller
- Support 8x8 key matrix
- 10-ch 12-bit multiplexed ADC
- 2-ch CAN interface
- Configurable GPIOs
- Real time clock, PLL, timer with PWM and watch dog timer
- Memory Subsystem
- SRAM/ROM/NOR/NAND Interface with x8 or x16 data bus
- MuxedOneNANDInterface with x16 data bus
- 1 port Mobile DDR Interface with x32 data bus (up to 333Mbps/pin DDR)
- 1 port DDR2 interface with x16 or x32 data bus (333Mbps/pin DDR)
- 1 port LPDDR2 interface (up to 333Mbps/pin DDR)

2.3 MEMORY

2.3.1 DDR2 512MB

Hynix DDR2 memory chips provide 512MB RAM which opens up a wide range of production and development opportunities. DDR2 memory consumes more power than MobileDDR. But, it is pretty much cost effective for huge memory space.

You can choose memory vendor for example Samsung/Hynix/Elpida/Micron or others due to PC industry standard. DDR2 8Bit-bus-width memory is being used for PC/Notebook memory module.

Each chip has 8-bit data-bus-width with 128MB. We mounted 4 chips to make 32bit data bus and total memory size is 512MB.

2.3.2 2GB T-Flash(Micro-SD) Card

Traditional memory (Like NOR-FLASH / NAND-FLASH) can't be found in KIT-S5PC100. KIT-S5PC100 has totally different booting method. KIT-S5PC100 implements direct T-Flash booting with excellent iROM feature of S5PC100.

KIT-S5PC100 has 2Gbyte T-flash(removable) memory card for system area. Assume it has 3,862,528 blocks(sectors) and each block has 512bytes.

Area Name	Size in bytes	From(sector #)	To(Sector #)
<i>eFuse</i>	1K	3862526	3862527
<i>U-boot BL1</i>	8K	3862510	3862525
<i>U-boot Environment variables</i>	16K	3862478	3862509
<i>U-boot BL2</i>	512K	3861454	3862477
<i>Reserved</i>			
<i>Linux Kernel</i>	4M	3854336	3862527
<i>Reserved</i>			
<i>EXT3 for Android system</i>	1GB	0	2097152



Note: Reserved area will be used for FW update and Recovery purpose.

2.3.3 2GB SDHC Card

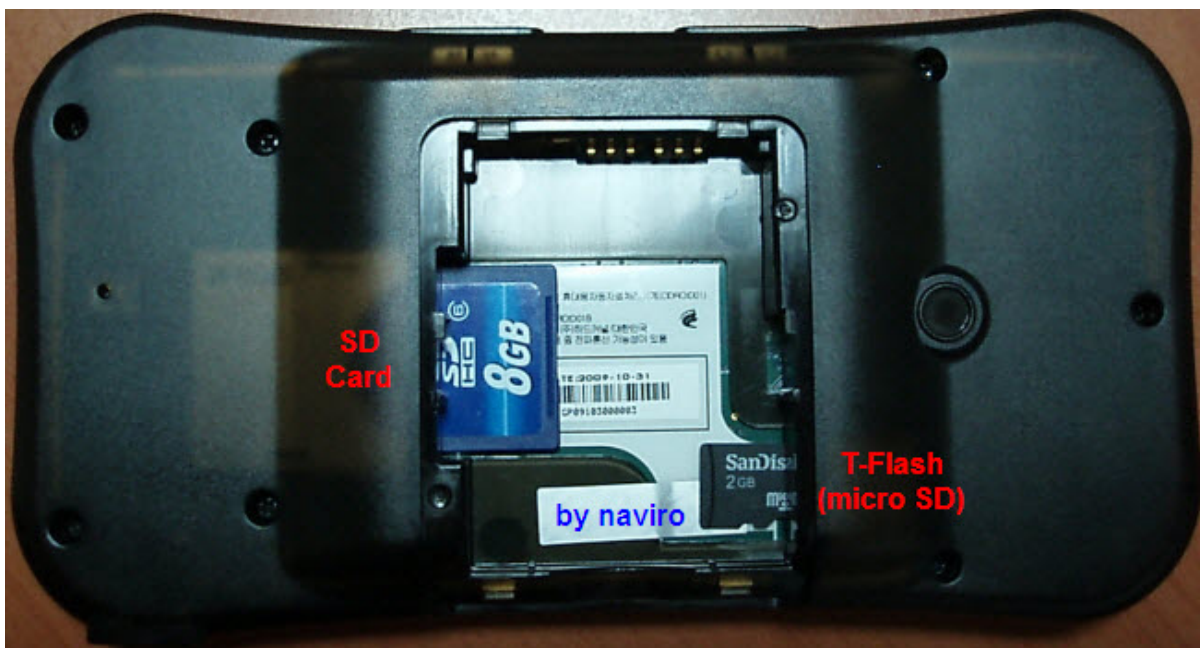
At this moment, Google Android supports only FAT32 file system. Other file system can be considered for big size 720p-HD contents. 2GB/4GB/8GB seems to be working well. But, 16GB memory card has some compatibility issues.

2.3.4 Where is T-flash (Micro-SD)

KIT-S5PC100 has 2 memory cards.

- T-flash card contains boot-loader, kernel, Android system and applications.
- SD card contains user files such as pictures, music, video clips and so on.

Open the battery cover and you can find them.



2.4 Display

2.4.1 LCD

- LMS350DF01-001 is a TMR(Transmissive with Micro Reflective) type color active matrix TFT (Thin Film Transistor) liquid crystal display (LCD) that uses amorphous silicon TFT as a switching devices.
- This model is composed of a TFT- LCD module, a driver circuit and a back-light unit.
- The resolution of a 3.5" contains 320 x 480 dots and can display up to 16,777,216 colors.

2.4.2 Back-light driver

DW8400 from DongWoon Anatech (<http://www.dwanatech.com>)

The DW8400 is a step-up DC/DC converter designed for driving up to 10 white LEDs in series from a single cell Lithium-Ion battery.

2.4.3 Capacitive Touch screen controller

TouchCore 3.0 from CoreRiver.



2.5 WIFI

This elegant combination from Fujitsu Component, based on Marvell's industry leading 88W8686 and CSR's BC4, brings both 802.11(b/g) to the KIT-S5PC100.

If you want to use this module for your own product, contact Fujitsu first.

- <http://www.fcl.fujitsu.com/en/contact/>
- Wireless Modules, Others
- Global Marketing 4 Division

2.6 Audio Codec

The WM8991 is a highly integrated low power hi-fi CODEC from Wolfson. A powerful 1W speaker driver can operate in class D or AB modes. Stereo 24-bit sigma-delta ADCs and DACs provide hi-fi quality audio record and playback, with a flexible digital audio interface.

2.7 Power Supply

2.7.1 PMIC

PMIC MAX8698C supplies important power rails of S5PC100 and other devices. It contains 3 Step-down DC/DCs and 9 LODs.

Unfortunately, the data sheet is not opened to public access because of Maxim's policy. We will discuss this issue with Maxim soon.

2.7.2 DC/DC for Wireless device

Wifi-BT combo module consumes a lot of power and PMIC is not enough. So, KIT-S5PC100 takes additional high efficient DC/DC converter MAX1556 of Maxim.

2.7.3 DC/DC for HDMI-5Volt

HDMI needs 5Volt power rail. FAN5602 is a charge pump based step up/down DC/DC from Fairchild.

2.7.4 Li+ Battery Fuel Gauge

KIT-S5PC100 reads reliable battery level information from dedicated Li+ battery gauge Max17040 from Maxim.

2.7.5 Li+ Battery Charger

The XC6802 is a constant-current/constant-voltage linear charger IC for single cell lithium-ion batteries. Current source is USB-bus power.

2.8 Accelerometer

The digital 3-axial acceleration sensor of Bosch Sensortec is included in KIT-S5PC100.

2.9 Touch switch (Sound volume up/down)

KIT-S5PC100 volume switches are implemented with touch sensitive IC TS01 from



AD-semiconductor.

Chapter III Android System Development

Note: Compile the Android system, please use the Ubuntu 9.04 version, the other version may be have some problem for compiling.

3.1 Build Development environment

Users in the use KIT-S5PC100 developed; they must first build a good ARM Linux cross-development environment. Below to ubuntu9.04 operating system as an example, cross-development environment to build.

3.1.1 The installation of cross-compiling environment

Insert CD-ROM, ubuntu mount the CD-ROM to / media / cdrom directory, cross-compilation tool stored in the / media /cdrom/tools directory, the name for the cross-4.2.2-eabi.tar.bz2.

Users ` the following command to install the cross compiler tools:

```
# sudo mkdir /usr/local/arm
# sudo tar xvfj cross-4.2.2-eabi.tar.bz2 -C /usr/local/arm
```

3.1.2 Add Path in your environment file

Modify your ~/.bashrc file to add a new path with editor (gedit or vi)

```
PATH=$PATH:/usr/local/arm/4.2.2-eabi/usr/bin
```

To apply this change login again or restart the .bashrc

```
# source .bashrc
```

3.1.3 Check whether tool-chain path set up is correct or not.

```
# arm-linux-gcc -v
Using built-in specs.
Target: arm-unknown-linux-gnueabi
Configured with:
/home/scsuh/workplace/coffee/buildroot-20071011/toolchain_build_arm/gcc-4.2.2/configure --prefix=/usr --build=i386-pc-linux-gnu --host=i386-pc-linux-gnu --target=arm-unknown-linux-gnueabi --enable-languages=c,c++ --with-sysroot=/usr/local/arm/4.2.2-eabi/ --with-build-time-tools=/usr/local/arm/4.2.2-eabi//usr/arm-unknown-linux-gnueabi/bin --disable-__cxa_atexit --enable-target-optspace --with-gnu-ld --enable-shared --with-gmp=/usr/local/arm/4.2.2-eabi//gmp --with-mpfr=/usr/local/arm/4.2.2-eabi//mpfr --disable-nls --enable-threads --disable-multilib --disable-largefile --with-arch=armv4t --with-float=soft --enable-cxx-flags=-msoft-float
Thread model: posix
```



```
gcc version 4.2.2
```

3.1.4 Download and Install essential packages for android

1) Android requires the following system packages:

- flex: This lexical analyzer generator is used to read a given input file for a description of a scanner to generate.
- bison: This is a general-purpose parser generator.
- gperf: This is a perfect hash function generator.
- libesd0-dev: This enlightened sound daemon (dev files) is used to mix digitized audio streams for playback by a single device.
- libwxgtk2.6-dev: This package provides GUI components and other facilities for many different platforms.
- build-essential: This package contains a list of packages considered fundamental to building Debian packages.
- Android source code includes a hard dependency on the Java Developer Kit (JDK) 5.0 Update 12 or greater.

Download packages.

```
# sudo apt-get install flex bison gperf libstdc++-dev libesd0-dev libwxgtk2.6-dev  
build-essential zip curl  
# sudo apt-get install valgrind
```

2) Android also needs Python 2.4 or higher. But, Ubuntu has Python 2.5 in general.

3) Install JDK 5.0 Update12 or later.

```
# sudo apt-get install sun-java6-jdk
```

4) Add Path in your environment file. Modify your ~/.bashrc file to add a new path with editor (gedit or vi)

```
export PATH=/home/$USER/bin:$PATH  
export JAVA_HOME=/usr/lib/jvm/java-6-sun-1.6.0.16  
export ANDROID_JAVA_HOME=$JAVA_HOME
```

5) To apply this change login again or restart the .bashrc

```
# source .bashrc
```

3.2 System Compiler

3.2.1 Ready to build file

All components of the system source code of the linux directory in the CD-ROM, the user before carrying out the development need to untar them to the linux system, such as:

```
# mkdir ~/work
```



```
# cd ~/work
```

```
# tar xvzf /media/cdrom/linux/android.tar.gz
```

```
# tar xvzf /media/cdrom/linux/linux.tar.gz
```

After you perform these operations, the current directory will be generated android, linux, uboot 4 directory.

3.2.2 Compile the bootloader

```
# cd ~/work/uboot
```

```
# make mrproper
```

```
# make hkdmc100_config
```

```
# make
```

If there is no error, u-boot.bin should be generated and file size is about 130~150Kbyte.

3.2.3 linux kernel compilation

To start Kernel compile, .config file should be made first. KIT-S5PC100 Kernel configuration file is located in arch/arm/configs/hkdmc100_android_defconfig You can make .config with below command.

```
# cd ~/work/linux
```

```
# make hkdmc100_android_defconfig
```

Or you can copy manually hkdmc100_android_defconfig to .config in linux directory.

Let's start compile !

```
# make zImage
```

If you can build kernel successfully, you will have "arch/arm/boot/zImage" (approximately 2Mbytes).

3.2.4 Compile the Android file system

To build the files, run make from within your working directory:

```
# cd ~/work/android
```

```
# ln -s vendor/sec/smdmc100/build_android.sh .
```

```
# ./build_android.sh
```

If there is no problem, you can see below message. rootfs directory is automatically generated and all root file system of Android is located in that directory.(~/work/rootfs)

```
ok success !!!
```

```
#
```

- Note: Compile takes about 20 minutes with latest multi-core CPU based computer. Some old computer or virtual machine may need about 2 hours.



Chapter IV Update the image

4.1 update the uboot image to Micro SD

You can write the raw image files with this special utility directly from XP PC to T-flash. The utility KIT-S5PC100_TF_Burner_098a.exe is at the CD:/tools/ folder, if you want to update the image to micro SD, you should use it.

- This utility is very useful when KIT-S5PC100 boot-loader is damaged or when you have new T-flash card.

1) Remove T-flash card from KIT-S5PC100 and plug it in to Card Reader. If you don't have the card reader, buy it.

This picture is just an example.

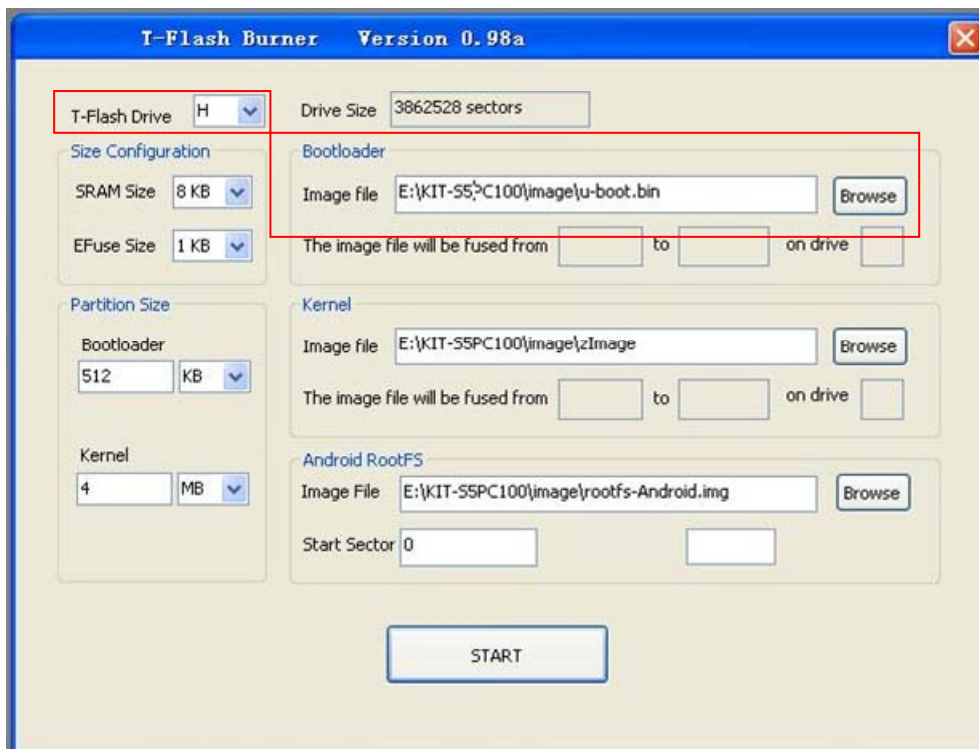


2) Connect the card reader to PC.

3) Check which drive is assigned for the card reader.

4) Run KIT-S5PC100_TF_Burner_098a.exe

5) Select drive of the card reader and Browse button near by Bootloader Image file and choose "u-boot.bin"



6) Click "START" button and writing process will be done quickly.

Note) Do not change other parameters. Changing will cause malfunction.

Where is T-flash (Micro-SD)

KIT-S5PC100 has 2 memory cards.

- T-flash card contains boot-loader, kernel, Android system and applications.
- SD card contains user files such as pictures, music, video clips and so on.

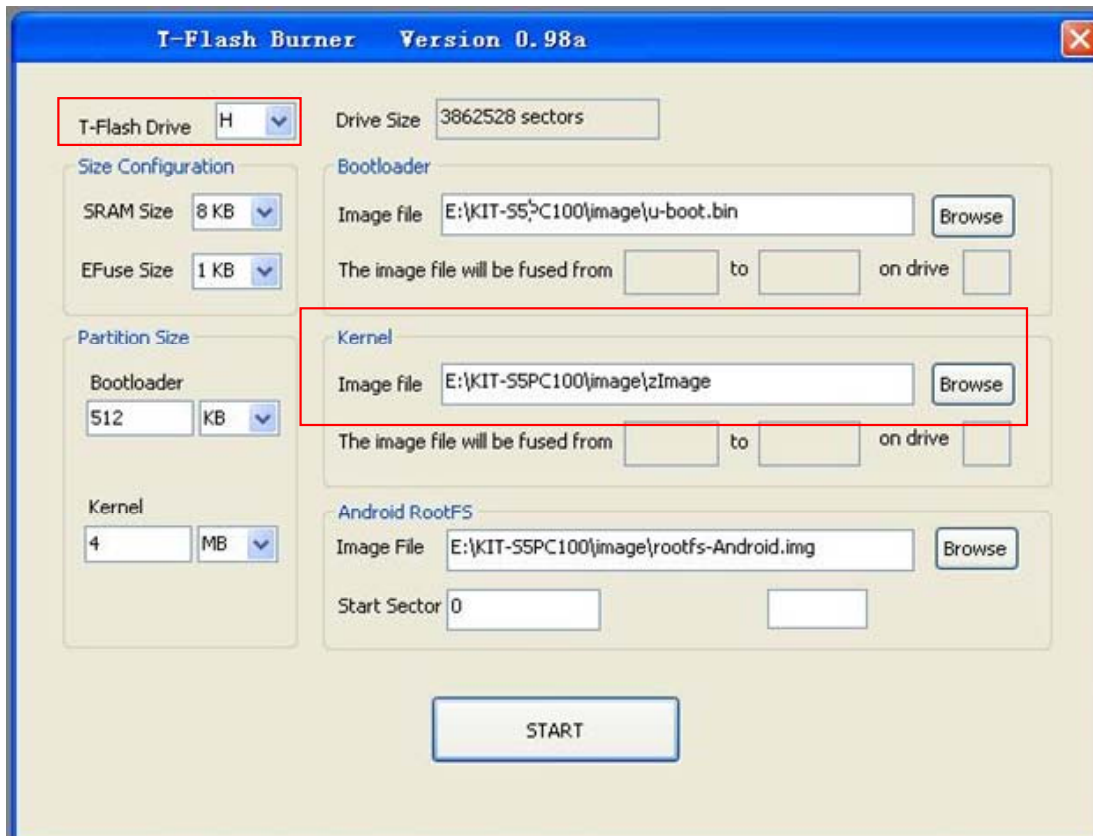
Open the battery cover and you can find them.



4.2 update the kernel image to Micro SD

This is same way to transfer u-boot.bin to T-flash(Micro-SD).

- 1) Run KIT-S5PC100_TF_Burner_098a.exe and select drive of the card reader and Browse button near by Kernel Image file and choose "zImage"
- 2) Click "START" button and writing process will be done quickly.



Note) Do not change other parameters. Changing will cause malfunction.

4.3 update the Android image to Micro SD

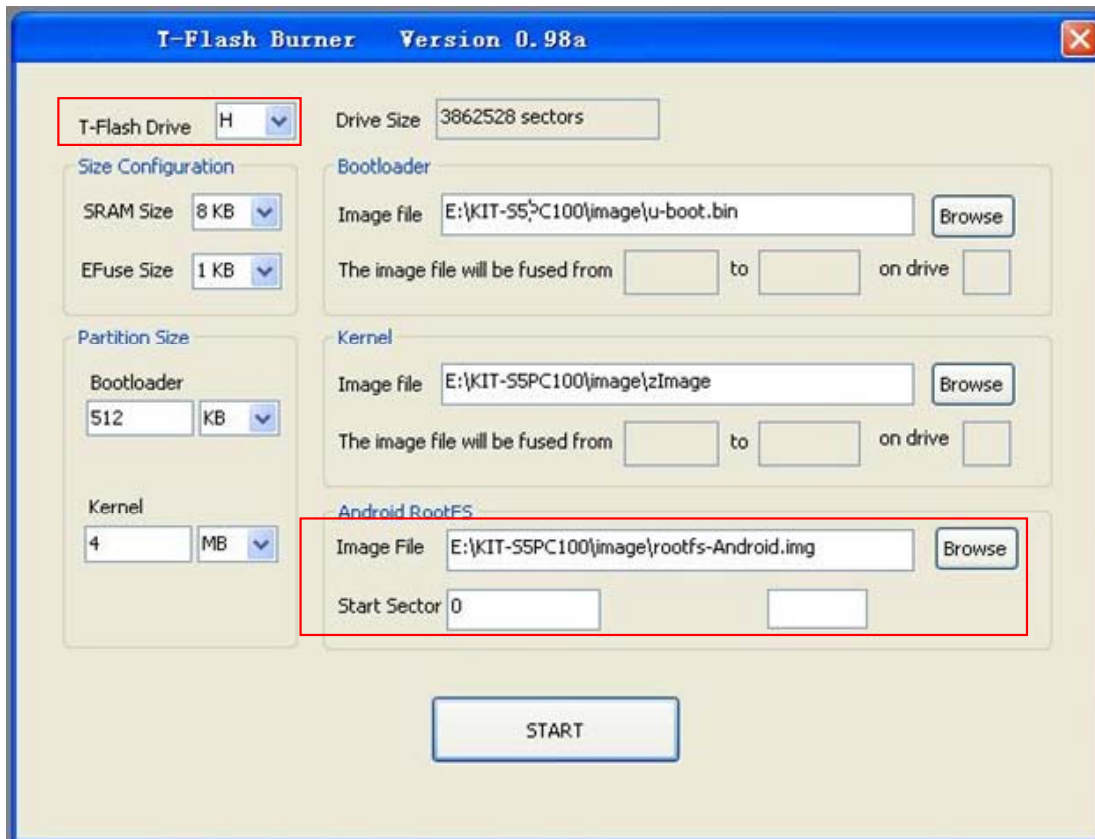
It has two methods to update the image for the board.

- If you want to update the rootfs that provide from the CD, you should use the method 1.
- If you want to update the rootfs that you Compiled, you should use the method 2.

<Method 1> update the rootfs that provide from the CD to Micro SD

This is same way to transfer u-boot.bin to T-flash(Micro-SD).

- 1) Run KIT-S5PC100_TF_Burner_098a.exe and select drive of the card reader and Browse button near by Rootfs Image file and choose " android-rootfs.img"
- 2) Click "START" button and you should will wait some moment.



Note) Do not change other parameters. Changing will cause malfunction.

<Method 2> update the rootfs that you Compiled to Micro SD

- 1) Insert the T-Flash in to Card reader and connect the card reader to Linux PC.
- 2) Copy all files in the KIT-S5PC100_rootfs to EXT3 partition of T-Flash.
- 3) Normally when you plug the Card reader into Ubuntu PC, Ubuntu will mount it automatically. You need to remember the mounting point.
- 4) To find the mounting points, perform below commands.

```
# mount
/dev/sdb1 on /media/disk-1 type ext3 (rw,nosuid,nodev,uhelper=hal)
---> Mount T-Flash first partition(Android data)
/dev/sdb2 on /media/disk-2 type ext3 (rw,nosuid,nodev,uhelper=hal) ---> Mount
T-Flash second partition(Android system) : remember this!
#
```

If the partition table is NOT correct, you need to make partition as below. You also need this process, if you want to use fresh(new) T-flash card.

```
==== Partitioning T-Flash card ====
1. Connect T-Flash card to PC via USB card reader.
   # sudo fdisk -l // Check the information of mounted disks.
2. If the T-Flash card is mounted, simply un-mount it.
   # sudo umount /dev/sdc1
3. You can double check whether the disk is un-mounted or not.
```



mount

4. Let's start partitioning with fdisk utility.

sudo fdisk /dev/sdc

5. Read current partition information with 'P' command.
6. Delete all partitions with 'D' command.
7. Create new partition with 'N' command.
8. Select primary partition with 'P' command.
9. Select partition number ==> '1'
10. Press Enter for first cylinder with default value.
11. Set partition size (+1G)

12. Create new partition with 'N' command.
13. Select primary partition with 'P' command.
14. Select partition number ==> '2'
15. Press Enter for first cylinder with default value.
16. Set partition size (+256M)

17. Create new partition with 'N' command.
18. Select primary partition with 'P' command.
19. Select partition number ==> '3'
20. Press Enter for first cylinder with default value.
21. Set partition size (+256M)

22. Configure disk system type with 'T' command for first partition.
partition number(1)
Hex code = 83 (83 is Linux)
23. Configure disk system type with 'T' command for second partition.
partition number(2)
Hex code = 83 (83 is Linux)
24. Configure disk system type with 'T' command for third partition.
partition number(3)
Hex code = 83 (83 is Linux)
25. Write and exit with 'W' command.
26. Final check the partition table.
sudo fdisk -l

Format those 3 partitions with EXT3 file system.

```
# sudo mkfs.ext3 /dev/sdc1  
# sudo mkfs.ext3 /dev/sdc2  
# sudo mkfs.ext3 /dev/sdc3
```



8.1 Copy rootfs to already formatted partition (Overwrite)

- Copy root file system to EXT3 partition of T-Flash.

- `# sudo cp -a ~/work/rootfs/* /media/disk-2 ---> mounting point (You may have different location)`
- `# sync`

- Unmount the T-Flash disk and move the card from PC to KIT-S%PC100 for testing.

- `# sudo umount /media/disk-1 (or) sudo umount /dev/sdb1 ---> Unmount T-Flash first partition(Android data)`
- `# sudo umount /media/disk-2 (or) sudo umount /dev/sdb2 ---> Unmount T-Flash second partition(Android system)`