SBC6000X Linux Application Development Guide

Rev. 1.3

Release: 2013-02-263-02-26

Revision history

Rev	Date	Description
1.1	20090514	Initial version
1.2	2009-7-30	
1.3	20120226	

Table of Contents

Preface

This Manual is to provide some basic knowledge on application development of embedded Linux, and help developers make preparations for advanced development of embedded Linux. It describes the set-up of development application environment and basic application test. User can learn the details in relevant chapters hereunder.

Chapter I Set-up of development environment

1.1 Software and hardware requirements

1.1.1 Hardware

- ✓ A host installed with Linux system
- ✓ Power supply
- ✓ Cross serial port wire
- ✓ A 10M/100M net cable
- ✓ A SBC6000X development board and some accessories

1.1.2 Software

- ✓ Configurations of serial port terminal

  The parameters of serial port terminal settings: 115200 8 N 1

Baud rate	115200
Data	8 bits
Parity	None
Stop	1 bit
Flow control	None

- ✓ ftp: to provide ftp services. User can download files over Internet.

1.2 Installation of arm-linux-gcc cross compiler

If you have not installed arm-linux-gcc, copy ".\tools\arm-linux-gcc-3.4.5-glibc-2.3.6-linux.tar.bz2" from CD to the folder in your Linux system. The installation command is: "tar –zxvf

arm-linux-gcc-3.4.5-glibc-2.3.6-linux.tar.bz2 –C /". Then define the path of compiler:

```
tar –zxvf arm-linux-gcc-3.4.5-glibc-2.3.6-linux.tar.bz2 –C /
export PATH=/usr/crosstool/gcc-3.4.5-glibc-2.3.6/arm-linux/bin/:$PATH
```

Notes: every time you open a terminal, you need define the path of compiler, so every time you open a terminal, you need execute it once

" export PATH=/usr/crosstool/gcc-3.4.5-glibc-2.3.6/arm-linux/bin/:$PATH"

1.3 Demonstrate cross compilation of hello world program

1.3.1 Edit source code

Edit the following source code on PC, and save it as hello.c

```
#include <stdio.h>
int main(int argc, char *argv[])
{
    printf("Hello World\n");
    return 0;
}
```

1.3.2 Compile hello

Use the following command in the compilation: a hello executable file will be generated.

```
# arm-linux-gcc -o hello hello.c
```

1.4 Download it to target board

1.4.1 Download it using USB-disk

Insert USB-disk into USB port of PC, then execute the following command to copy

hello to USB-disk.

```
#mount /dev/sda1 /mnt
#cp hello /mnt
#umount /mnt
```

Pull out the USB-disk and insert it into USB HOST port of SBC6000X, then perform operations following the commands below:

```
#mount /dev/sda1 /mnt      ; mount USB-disk
#cp /mnt/hello /bin        ; copy hello to the folder bin
#chmod a+x hello           ; change executable authority of hello
#hello                     ; execute hello
```

1.4.2 Download it over Internet

User can download it over Internet following the steps below: first, copy hello to the tftp shared folder, then use tftp to download it on SBC6000X, and change the execution authority as below:

If it is executed on PC:

```
# cp hello /home/tftp/      ; copy hello to tftp shared folder
```

If it is executed on SBC6000X:

```
#cd /bin                          ; enter the folder bin
# tftp -r hello -g 192.192.192.105 ; download hello program through tftp
#chmod a+x hello                  ; change executable authority of hello
#hello                            ; execute hello
```

Chapter II SBC6000X application test

SBC6000X test is based on the graphics. You can input what you want to test on the screen.

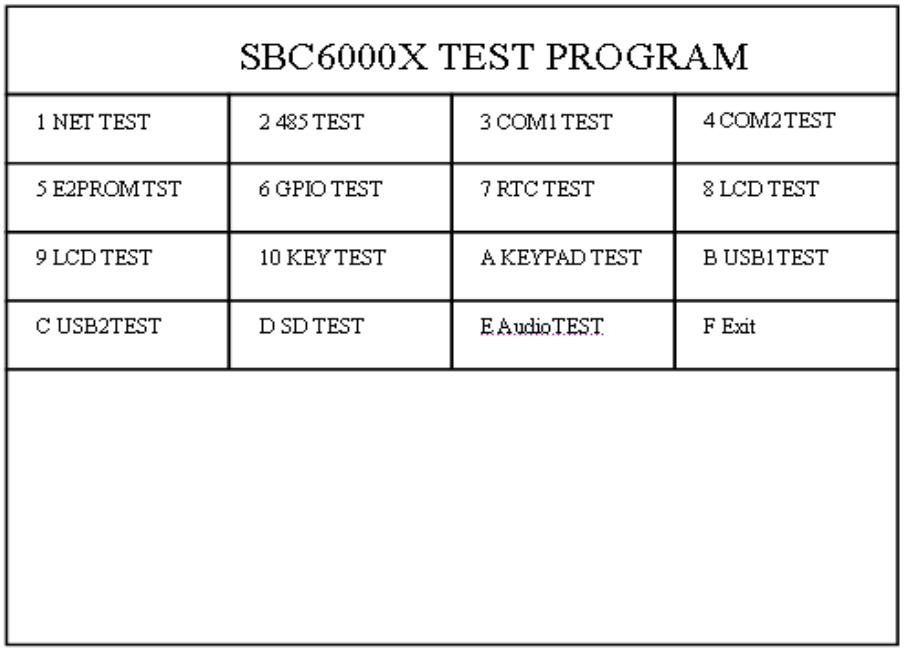

Figure 2 Graphics of the SBC6000X test

The SBC6000X test is simple. You can run the script to test the target board function.

```
./sbc6000x.sh
```

In the terminal after running sbc6000x.sh script, there will be running a touch screen calibration program. Click the center of the cross on the LCD screen.then You will enter the above interface.

2.1 RS485 test

We Know that RS485 and COM2 share a UART from the hardware manual. If you want to test the RS485 function, you should open the COM2 jumper cap. Make sure the TXD2 and RXD2 opened.

Found Interface plug-ins (J13) on the target board SBC6000X, closed the **RTS2** and

CTS2， **TXD24** and **TXD2**, **RXD24** and **RXD2**, **DTX-** and **DRX-, DTX+** and **DRX+** by the jumper cap.

DTX+	29	30	DTX-	RS485
DRX+	31	32	DRX-	
TXD24	37	38	TXD2	JP485
RXD24	39	40	RXD2	

Table 2.1 The RS485 pin

2.2 COM1/ COM2 test

COM1 and COM2 are five-wire UART. UART data is sent and received via TX output and RX input lines. Two additional lines RTS output and CTS input may also be used in this test.

Found Interface plug-ins (J13) on the target board SBC6000X, closed **TXD1** and **RXD1**, **RTS1** and **CTS1**, **TXD2** and **RXD2**, **RTS2** and **CTS2** by the jumper caps.

RTXD1	7	8	TXD1	COM1
RRXD1	9	10	RXD1	
RRTS1	11	12	RTS1	
RCTS1	13	14	CTS1	
RTXD2	15	16	TXD2	
RRXD2	17	18	RXD2	COM2
RRTS2	19	20	RTS2	
RCTS2	21	22	CTS2	

Table 2.2: COM1/COM2 pin

2.3 Keypad test

The Keypad test should conneted a 4x4 mtrix Keypad.

Found Interface plug-ins (J13) on the target board SBC6000X, the C(0~3) are the column(0~3),and the R(0~3) are the keypad row(0~3).

C0	11	12	R0	Keypad
C1	13	14	R1	
C2	15	16	R2	

C3	17	18	R3	

Table 2.2: 4-by-4 keypad pin

2.4 E2PROM test

The eeprom is accessed using an industry standard two-wire(I2C) interface.The memory array is logically organized as a 256 x 8 memory array.

In addition if you want to expand outside I2C interface devices, we also provide a way to expand the plug-interface I2C interface.

The I2C interface on the J23 plug-interface.

PA7	19	20	PA23	GPIO/TWI

Table 2.3 I2C interface pin

2.5 Button test

We provide two buttons that the SW1 and SW2 are on the target board. You should do nothing.

2.6 GPIO test

The GPIO tested method is to let GPIO output high-low level. GPIO is provided by the two parts IO. Six GPIO is extended from the TSC2301.

The GPIO interface (J23) show in the table below.

PC4	3	4	PC5	GPIO
PC6	5	6	PC7	
PA16	7	8	PB24	
PB23	9	10	PB26	
PB25	11	12	PC3	
EXG1	13	14	EXG2	
EXG3	15	16	EXG4	
EXG5	17	18	EXG6	

Table 2.3: GPIO pin

2.7 LCD Test

LCD test used the three colors to check if the lcd is OK. The three colors are red, bule and green.

2.8 USB/SD Card Test

When you put the USB disk into the USB host interface, the USB disk is mounted at /media/ folder. You can check if the USB disk or SD card are mounted at /media/ folder.

2.9 RTC Test

First set the hard real time clock, and then read the hard real-time clock to check if the real-time clock changes.

2.10 Audio Test

Madplay that we provide can play MP3 format music. When you run the bellow command.

```
madplay /home/mp3/You_and_Me.mp3
```

Then you can hear the beautiful voice.

Chapter III The WEB Server Test

First you should connect the target board net interface(J12) to the PC net interface. And also you should connect the target board debug serial interface to the PC serial interface. Open the power, and wait for a moment to enter the Linux system.

You can use the ifconfig command to check the network interface eth0 opened in the serial terminal. If the network interface eth0 was not opened, you can use the bellow command to open the eth0,and configure IP for the eth0.

```
~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr DE:AD:BE:EF:01:01
          inet addr:192.192.192.200  Bcast:192.192.192.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
          RX packets:5708 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:558722 (545.6 KiB)  TX bytes:0 (0.0 B)
          Interrupt:56 Base address:0xe000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
~ $ ifconfig eth0 192.192.192.200
~ $
```

Chech the web server is run by ps command in the serial terminal.

```
~ $ ps
PID   USER     COMMAND
    1 root     init
    2 root     [kthreadd]
    3 root     [ksoftirqd/0]
    4 root     [watchdog/0]
    5 root     [events/0]
    6 root     [khelper]
   59 root     [kblockd/0]
   72 root     [khubd]
   75 root     [kseriod]
   80 root     [kmmcd]
  102 root     [pdflush]
  103 root     [pdflush]
  104 root     [kswapd0]
  105 root     [aio/0]
  717 root     [mtdblockd]
  768 root     [kpsmoused]
  797 root     [rpciod/0]
  809 root     /sbin/udevd --daemon
  857 root     /home/lianxj/apache/bin/httpd
  859 root     -sh
  860 root     init
  876 nobody   /home/lianxj/apache/bin/httpd
  877 nobody   /home/lianxj/apache/bin/httpd
  878 nobody   /home/lianxj/apache/bin/httpd
  881 nobody   /home/lianxj/apache/bin/httpd
  882 nobody   /home/lianxj/apache/bin/httpd
  884 root     ps
~ $
```

If the web server is not run , now you can use the bellow command to start it.

```
~ $ apachectl start
/home/lianxj/apache/bin/apachectl start: httpd started
~ $
```

If you want to restart the web server, you can use the bellow command to restart it.

```
~ $ apachectl restart
/home/lianxj/apache/bin/apachectl restart: httpd restarted
~ $
```

If you want to stop the web server, you can use the bellow command to stop it.

```
~ $ apachectl stop
/home/lianxj/apache/bin/apachectl stop: httpd stopped
~ $
```

After start web server, you can input the address of the target board IP address in the PC browser to access the httpd server of the target board.

```
http://192.192.192.200
```

Now you can see the result of web server

Table 3: web server