GoBian

Getting Started with GoBian

v1.0

Copyright Statement

Copyright in this document is owned by GoWarrior Community. Any person is hereby authorized to view, copy, print and distribute this document subject to the following conditions:

- The document may be used for informational purposes only
- The document may be used for non-commercial purposes only
- Any copy of this document or portion thereof must include this copyright notice

This document is provided "as is" without any warranty of any kind, either express or implied, statutory or otherwise; without limiting the foregoing, the warranties of satisfactory quality, fitness for a particular purpose or non-infringement are expressly excluded and under no circumstances will GoWarrior Community be liable for direct or indirect loss or damage of any kind, including loss of profit, revenue, goodwill or anticipated savings.

This document is intended only to assist the reader in the use of the product. GoWarrior Community makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, which is used at your own risk and should not be relied upon. The information could include technical inaccuracies or typographical errors. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property right is granted by this document.

The product described in this document is subject to continuous development and improvements. GoWarrior Community also reserves the right to make changes to the specifications and product description at any time without notice.

Third-party brands and names mentioned in this publication are for identification purpose only and may be the property of their respective owners.

Android$^{\text{TM}}$ is a registered trademark of Google Inc. Linux® is a registered trademark of Linus Torvalds. Microsoft® and Windows® are registered trademarks of Microsoft Corporation. Supply of this product does not convey a license nor imply a right under any patent, or any other industrial or intellectual property right of Google Inc., Linus Torvalds, and Microsoft Corporation to use this implementation in any finished end-user or ready-to-use final product. It is hereby notified that a license for such use is required from Google Inc., Linus Torvalds and Microsoft Corporation.

For the latest version of this document refer to:

www.gowarriorosh.com

CONTENTS

List of Tables

List of Figures

Preface

Overview

GoBian is a Linux-based operating system provided by GoWarrior. This document describes GoBian by the following chapters:

Chapter 1: Meet GoBian

This chapter introduces GoBian and its features.

Chapter 2: TIGER Board

This chapter introduces the hardware platform.

Chapter 3: Getting Started

This chapter describes how to install GoBian and get GoBian started.

Chapter 4: GoBian Desktop

This chapter describes the GoBian desktop environment.

Chapter 5: Multimedia Framework

This chapter introduces Kodi, GOF SDK and DMA multimedia framework

Chapter 6: Interfacing TIGER Board

This chapter provides demo projects with the board.

Chapter 7: Building GoBian

This chapter describes how to compile GoBian.

Audience

This manual is applicable for the users who wish to learn how to run GoBian on TIGER Board for the first time. Readers are assumed to have certain knowledge and background on Linux embedded development.

Applicable Products

This manual is applicable for the GoWarrior TIGER Board.

Reference Documents

N/A.

Conventions

Typographical Conventions

Item	Format
codes, keyboard input commands, file names, equations, and math	`Courier New, Size 10.5`
Variables, code variables, and code comments	*`Courier New, Size, Italic`*
Menu item, buttons, tool names	**Ebrima, Size 10.5, Bold** e.g. Select **USB Debugging**
Screens, windows, dialog boxes, and tabs	**Ebrima, Size 10.5, Bold** **Enclosed in double quotation marks** e.g. Open the **"Debug Configuration"** dialog box

Table 1. Typographical Conventions

Symbol Conventions

Item	Description
⚠*Caution*	Indicates a potential hazard or unsafe practice that, if not avoided, could result in data loss, device performance degradation, or other unpredictable results.
Note	Indicates additional and supplemental information for the main contents.
💡 *Tip*	Indicates a suggestion that may help you solve a problem or save your time.

Table 2. Symbol Conventions

How to Contact Us

Submit all your comments and error reports to the author at:

info@gowarriorosh.com

Tel: +886-2-8752-2000

Fax: +886-2-8751-1001

For questions regarding GoWarrior, contact our support team at the email listed below:

support@gowarriorosh.com

1 Meet GoBian

GoBian is a Linux-embedded operating system running on TIGER Board for the GoWarrior platform. GoBian is developed based on Raspbian which is from Debian 7 wheezy armhf, and differs from Raspbian for the extra features, for example, GoBian encapsulates the RPi.gpio and other I/O libraries to facilitate transplanting projects which use the related libraries from Raspberry Pi to TIGER Board. Furthermore, GoBian provides support for multimedia by integrating GOF, Kodi and other middleware modules and applications out-of-the-box.

GoBian is characterized by the following features:

1. Networking & Remote Access

GoBian enables the Internet connection through Ethernet or Wi-Fi through the Ethernet port and Wi-Fi module on TIGER Board, and thus supports various methods to access to your projects and transfer data, such as FTP, SSH.

2. Timekeeping

GoBian automatically synchronizes the system time with Internet time servers using the NTP protocol.

3. File System

GoBian integrates a built-in file system to facilitate your data management.

4. Built-in Programming Environments

C, C++, Python, Perl, and shell script.

5. I/O Interfaces

GoBian lets you call the GPIO/I2C/UART/SPI interfaces directly in your projects with the built-in RPi.gpio and other libraries.

6. Multimedia

GoBian makes it easy for the secondary development of multimedia applications with the customized GOF middleware for TIGER Board.

7. Multitasking

The on-board ALi's M3733-AFAAA processor makes GoBian a multitasking system with good performance.

8. Linux Software

The software programs that are available for Debian are basically compatible with GoBian.

9. Low-Power Sleep Mode

GoBian supports the ultra-power-saving sleep mode (PMU Standby), with the entire board power consumption as low as 0.35W.

10. Multi-Screen Sharing

GoBian integrates with DLNA to fully support the multimedia sharing and multi-screen interaction.

11. Community Support

The GoWarrior community is already launched to support your projects with GoBian.

2 TIGER Board

TIGER Board is the highly-integrated and powerful development board of GoWarrior. The current version of TIGER Board is v1.1.

TIGER Board is equipped with two 40-pin expansion connectors J3 and J4 to connect external devices through GPIO, I2C, SPI, PWM, etc, and J3 is fully compliant with Raspberry Pi 2 Model B. The board supports various types of peripherals, such as TV sets, MicroSD cards, USB Flash drives, USB hard disks, and sensors.

TIGER Board also supports connecting USB cameras while running GoBian. Most USB cameras with a standard USB Video Class (UVC) driver are supported by TIGER Board for image shooting and video recording via third party programs, such as guvcview.

"*TIGER Board System Reference Manual*" describes the detailed specifications about the board. The following figures give you a bird's eye view of the main components and connectors on the board.

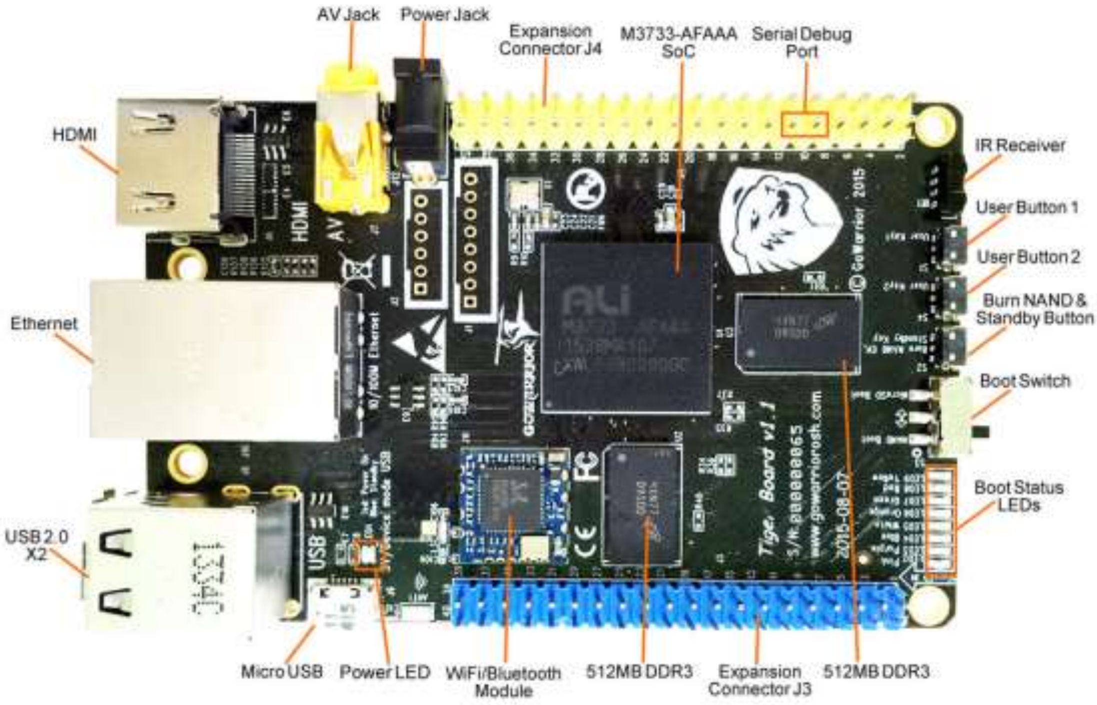


Figure 1. TIGER Board Top Side

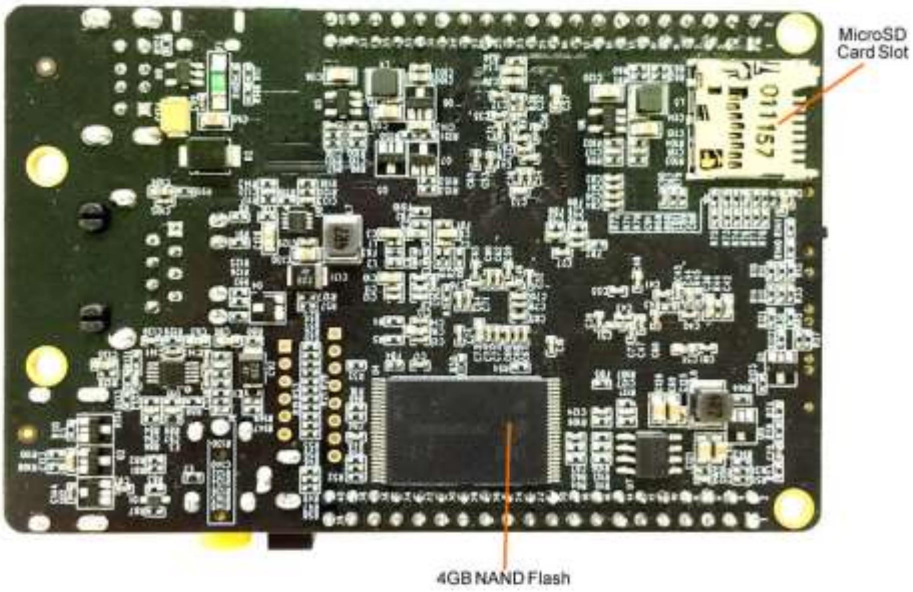


Figure 2. TIGER Board Bottom Side

3 Getting Started

GoBian is not an OOTB operating system for TIGER Board, you need to install it to the board with a MircoSD card. However, installing GoBian to TIGER Board does not delete GoDroid which is the default operating system of the board, because GoBian is installed on the MicroSD card which is used to boot the board instead of the on-board NAND Flash. That is, after installing GoBian, you can decide (by switching the Boot Select button on the board) which operating system runs on the board for each boot. The following figure shows the procedure of getting GoBian started with TIGER Board for the very first time.

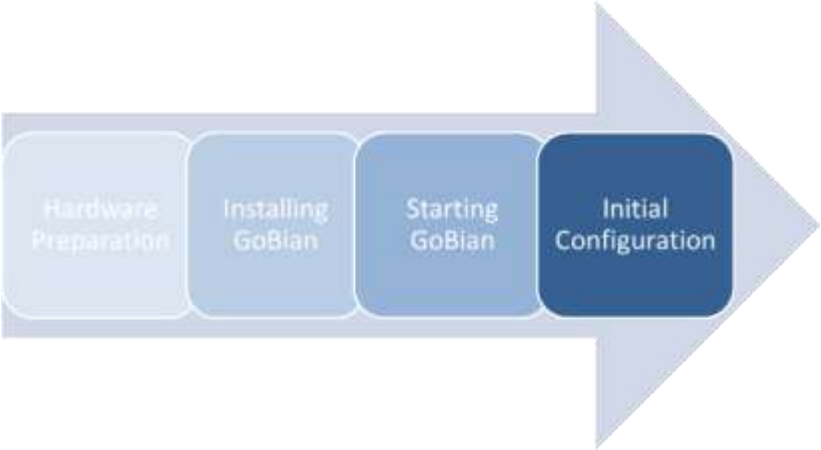


Figure 3. Procedure of Getting Started

3.1 Hardware Preparation

Hardware	Options	Description
Power supply	5V/2A Power Supply	Provides power supply through the 5V power Jack on the board.
	PC (with a Micro-USB cable)	Provides power supply by connecting to the board with a Micro-USB cable.

Hardware	Options	Description
Display device	TV/Monitor (with an HDMI or AV Jack cable)	Supports HDMI or CVBS as input video source for GoBian initial configuration.
Input devices	USB mouse + keyboard	Serve as input device for GoBian initial configuration.
PC (with a MicroSD card reader)	N/A	Handles GoBian image file
MicroSD card	N/A	8 GB or 16 GB

Table 3. Hardware Preparation

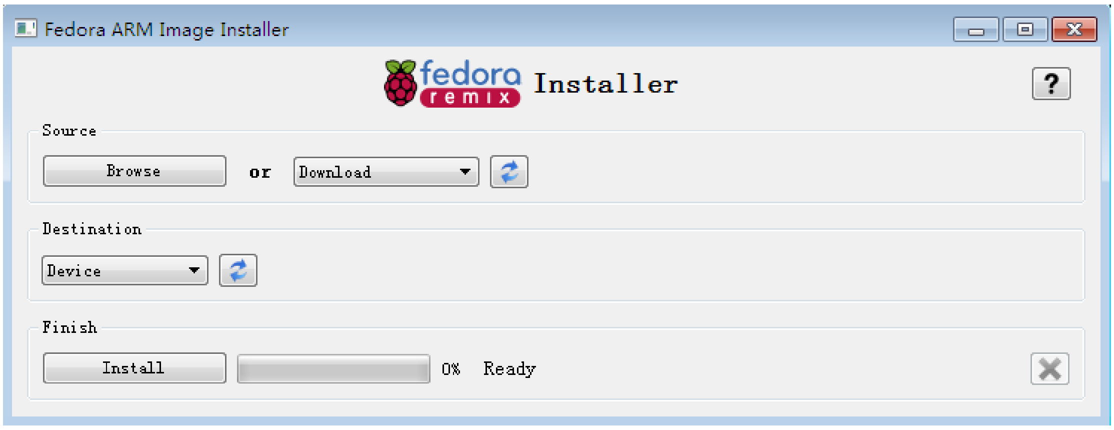
3.2 Installing GoBian

Before starting to install GoBian, ensure that you have made hardware preparation for the installation, and download the GoBian image file from GoWarrior website to your PC.

For the installation, you have options to install with a PC running Windows, Linux or OS X.

3.2.1 Installing with a Windows PC

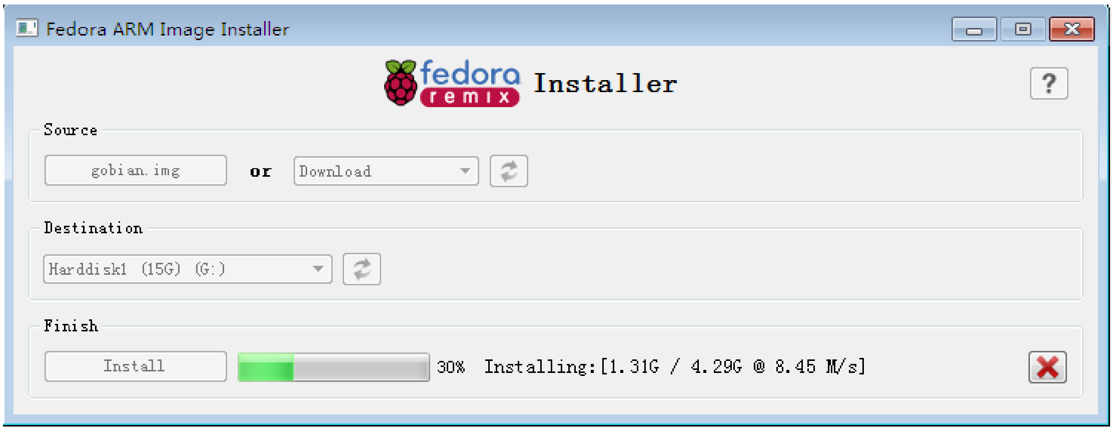
1. Download Fedora ARM Image Installer and install it to your PC.
2. Insert your MicroSD card to PC, and then launch the Installer.
3. In the Installer, click **Browse** to select the GoBian image, and then click **Device** to select your MicroSD card.



0% Ready

Figure 4. Selecting Source

4. Click **Install**.



Source: gobian.img

Destination: Harddisk1 (15G) (G:)

Finish: 30% Installing:[1.31G / 4.29G @ 8.45 M/s]

Figure 5. Clicking Install

3.2.2 Installing with a Linux PC

1. Insert your MicroSD card to PC.
2. Install GoBian to the card.

For example, if your card is at /dev/sdb1, then execute the following command:

```
sudo ddrescue -d -D --force GoBian.img /dev/sdb1
```

3.2.3 Installing with a Mac OS X

1. Insert your MicroSD card to PC.
2. Unmount the MicroSD card.

For example, if your card is at /dev/disk3s1by device name, then execute the following command in Terminal utility:

```
sudo diskutil unmount /dev/disk3s1
```

3. Use the device name of the card to work out the raw device name by replacing disk with rdisk and omitting the s1.

For example, the raw device name for the device `/dev/disk3s1` is `/dev/rdisk3`.

4. Install GoBian to the card.

For example, if your card is at `/dev/rdisk3` by the raw device name, then execute the following command:

```
sudo dd bs=1m if=~/gobian.img of=/dev/rdisk3
```

3.3 Starting GoBian

Follow these steps to start GoBian for the first time:

1. Insert your MicroSD card to the slot on the board.
2. Switch the **Boot Select** button to MicroSD Boot.
3. Connect your display device, USB keyboard and mouse to the board.
4. Power on the board with your power supply.

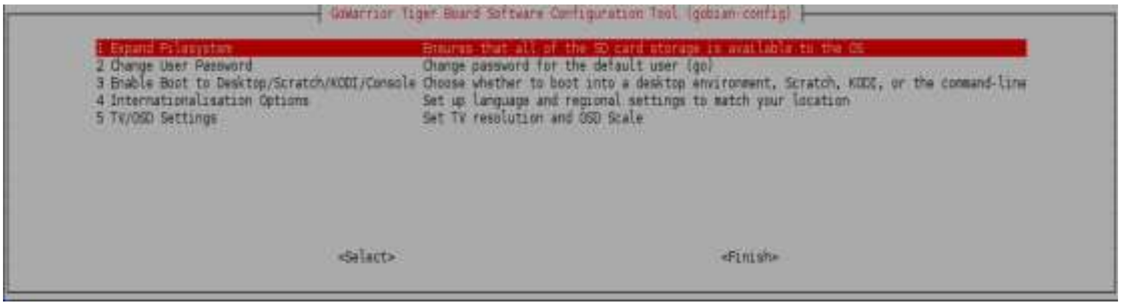
For the position of each connector on the board, see **section 2 TIGER Board**.

3.4 Initial Configuration

After powering up the board for the first time, initialize GoBian with the gobian-config tool, and set your MAC address.

3.4.1 Initializing GoBian with gobian-config Tool

At the very first starting of GoBian, gobian-config tool is launched for your initial configuration to GoBian as shown in the following figure.



```
GoWarrior Tiger Board Software Configuration Tool (gobian-config)

1 Expand Filesystem                        Ensures that all of the SD card storage is available to the OS
2 Change User Password                     Change password for the default user (go)
3 Enable Boot to Desktop/Scratch/KODI/Console Choose whether to boot into a desktop environment, Scratch, KODI, or the command-line
4 Internationalisation Options             Set up language and regional settings to match your location
5 TV/OSD Settings                          Set TV resolution and OSD Scale

              <Select>                              <Finish>
```

Figure 6. gobian-config

The following table describes each option in the menu of the tool.

Option	Description
Expand Filesystem	Enables you to use the whole MicroSD card.
Change User Password	Enables you to change the password for the default user go. The default password for the user go is gobian.
Enable Boot to `Desktop/Scratch/KODI/Console`	Selects boot to `desktop/console/scratch/KODI`.
Internationalisation Options	Changes locale, timezone and keyboard layout.
V/OSD Setting	Sets TV resolution and OSD scale ratio. Changes to this option take effect after restarting GoBian.

Table 4. Options of Configuration Tool

To launch the tool again after entering GoBian, execute the following command:

```
sudo gobian-config
```

Other than setting the resolution and OSD scale ratio in the tool, you can also set them by configuring the relevant parameters in the file `/boot/config.txt` after entering GoBian.

To set the resolution, change the value for the tv_mode parameter in the file according to the table below.

Resolution	Value
PAL	tv_mode=0
NTSC	tv_mode=1
PAL_M	tv_mode=2
PAL_N	tv_mode=3
PAL_60	tv_mode=4
NTSC_443	tv_mode=5
SECAM	tv_mode=6
MAC	tv_mode=7
LINE_720_25	tv_mode=8
LINE_720_30	tv_mode=9
LINE_1080_25	tv_mode=10
LINE_1080_30	tv_mode=11
LINE_1080_50	tv_mode=12
LINE_1080_60	tv_mode=13
LINE_1080_24	tv_mode=14
LINE_1152_ASS	tv_mode=15
LINE_1080_ASS	tv_mode=16

Resolution	Value
PAL_NC	tv_mode=17
LINE_576P_50_VESA	tv_mode=18
LINE_720P_60_VESA	tv_mode=19
LINE_1080P_60_VESA	tv_mode=20

Table 5. tv_mode Parameter for Setting Resolution

The following is an example of setting the resolution

```
tv_mode=13
```

To set the OSD scale ratio, change the value for the osd_ratio parameter in the file.The following is an example of setting the OSD scale ratio.

```
osd_ratio=95
```

3.4.2 Setting MAC Address

After entering GoBian, set your MAC address by changing the value of the mac parameter in the file /boot/config.txt. The following is an example of setting MAC address.

```
mac=00:11:22:33:44:55
```

To get the MAC address, check the packing box of your TIGER Board. The new MAC address takes effect after rebooting the system.

4 GoBian Desktop

GoBian brings you Windows-like desktop environment with terminal application to provide command line interface, thus you do not need to reboot the system for getting to the command line console when you are using the GoBian desktop.

GoBian enters the command line console by default after booting. To enter the desktop, execute the following command:

```
go@tigerboard:~$ startx
```

The following is a snapshot of the GoBian desktop.

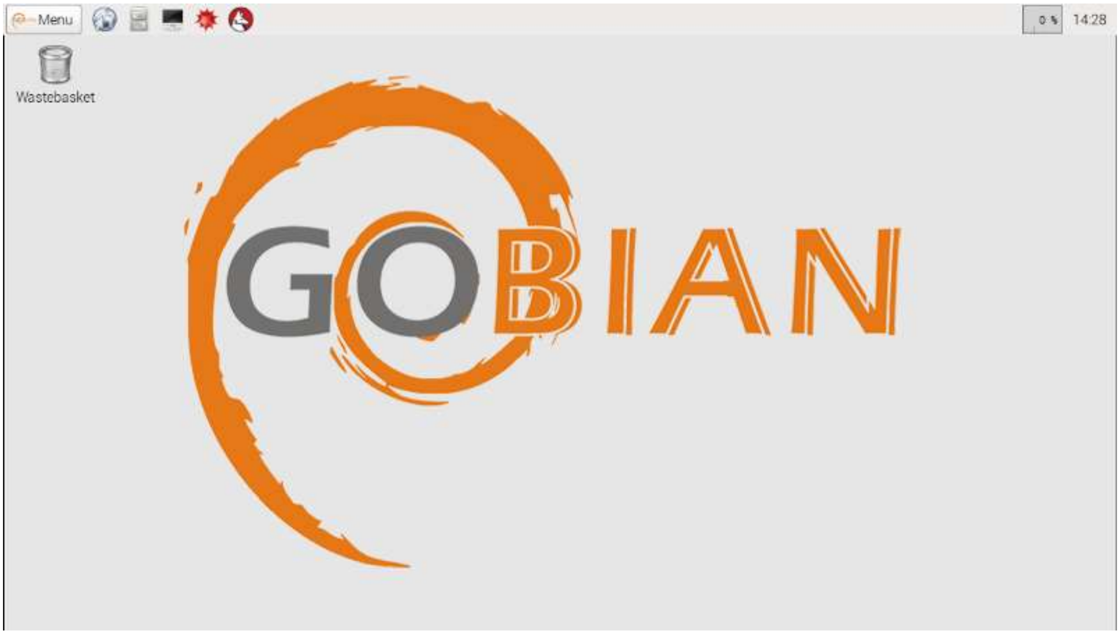
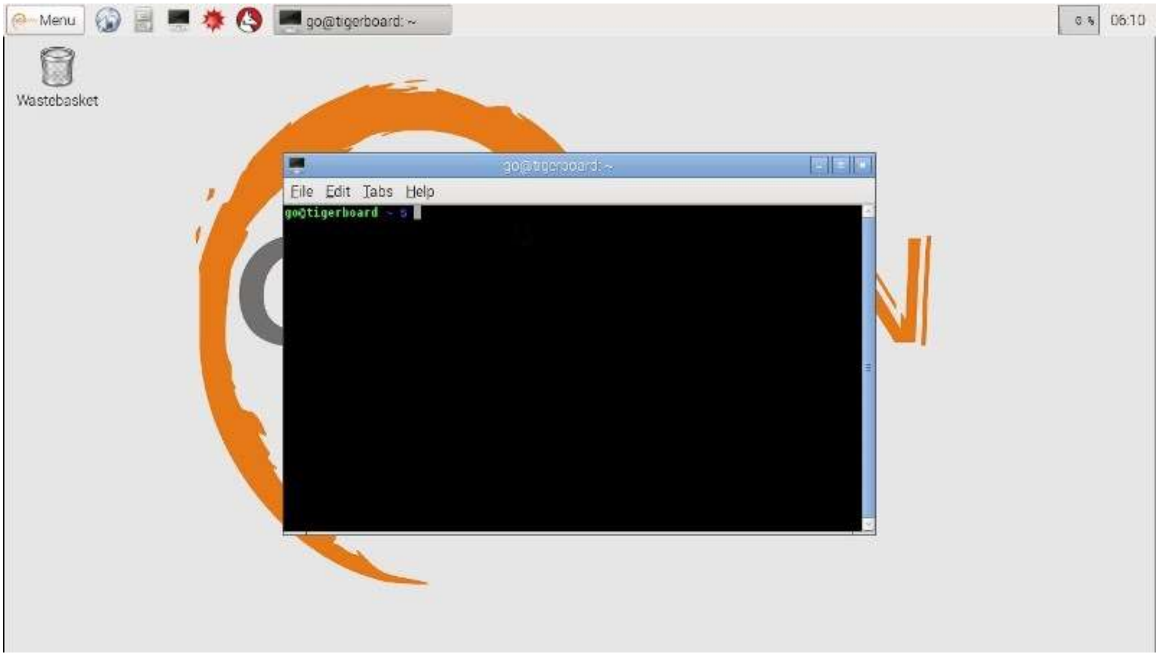


Figure 7. GoBian Desktop

4.1 Terminal

GoBian provides a terminal application out-of-the-box where you can get a command line interface on the desktop.



```
go@tigerboard ~ $
```

Figure 8. GoBian Terminal

To launch a terminal, select **Menu** > **Accessories** > **Terminal**, or click on the top panel.

You can also retrieve up to six virtual consoles on the desktop environment by pressing **Ctrl + Alt +F1-F6**. To return to the desktop from a virtual console, press **Ctrl + Alt + F7**.

4.2 File Manager

File Manager is a file management tool, which enables you to manage your files in GoBian as simple as in Windows.

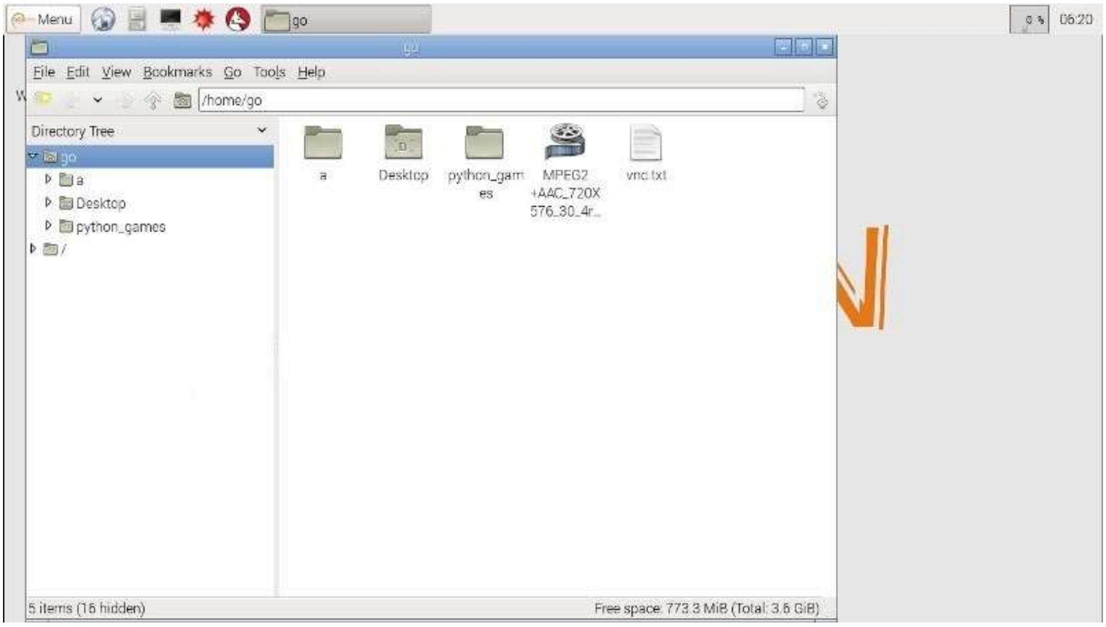
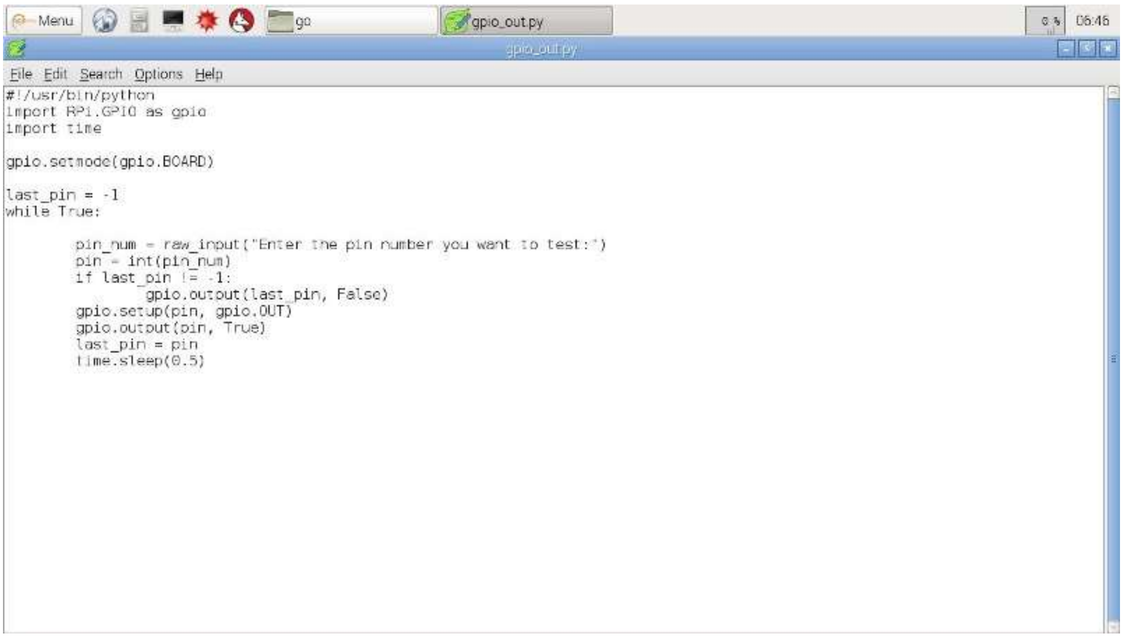

Menu
go
06:20
File Edit View Bookmarks Go Tools Help
/home/go
Directory Tree
go
a
Desktop
python_games
/
a
Desktop
python_games
MPEG2 +AAC_720X 576_30_4r...
vnc.txt
5 items (16 hidden)
Free space: 773.3 MiB (Total: 3.6 GiB)

Figure 9. File Manager

To launch it, select **Menu** > **Accessories** > **File Manager**, or click 🗄 on the top panel.

4.3 Text Editor

Leadpad, a simple text editor, is pre-installed with GoBian.



```
#!/usr/bin/python
import RPi.GPIO as gpio
import time

gpio.setmode(gpio.BOARD)

last_pin = -1
while True:

        pin_num = raw_input("Enter the pin number you want to test:")
        pin = int(pin_num)
        if last_pin != -1:
                gpio.output(last_pin, False)
        gpio.setup(pin, gpio.OUT)
        gpio.output(pin, True)
        last_pin = pin
        time.sleep(0.5)
```

Figure 10. Text Editor

To launch Leadpad, select **Menu** > **Accessories** > **Text Editor**.

4.4 Script

As the other UNIX/Linux operating systems, the script to be executed in GoBian also need a correct shebang line in it, and the user who executes the script should be assigned proper permission.

4.4.1 Shebang

The Shebang used in scripts indicates an interpreter for execution under GoBian.

For a Bash script, add the following line to the top of your script.

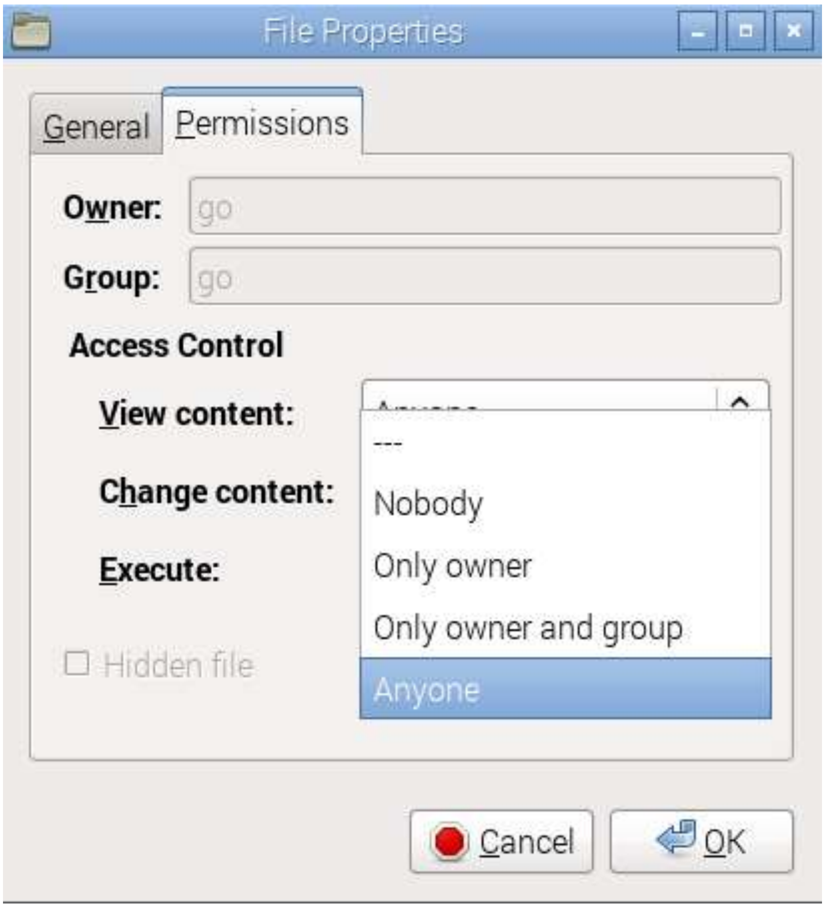
```
#!/bin/bash
```

For a Python script, add the following line to the top of your script.

```
#!/usr/bin/python
```

4.4.2 Permission

Other than the *chmod* command, you can also change the access permissions to a script by on the desktop. To do so, right click on a script and select **File Properties**, select the **Permissions** tab and change the access permissions accordingly.



Owner: go

Group: go

Access Control

View content: Anyone

Change content:

Execute:

□ Hidden file

Figure 11. Permission

4.5 Remote Access with TightVNC

GoBian supports for remote access with TightVNC. To use TightVNC, install and launch it by performing the following steps:

1. Install TightVNC by executing the following command:

```
go@tigerboard:~$ sudo apt-get install tightvncserver
```

2. Launch TightVNC by executing the following command:

```
go@tigerboard:~$ vncserver

You will require a password to access your desktops.

Password:
Verify:
Would you like to enter a view-only password (y/n)?
```

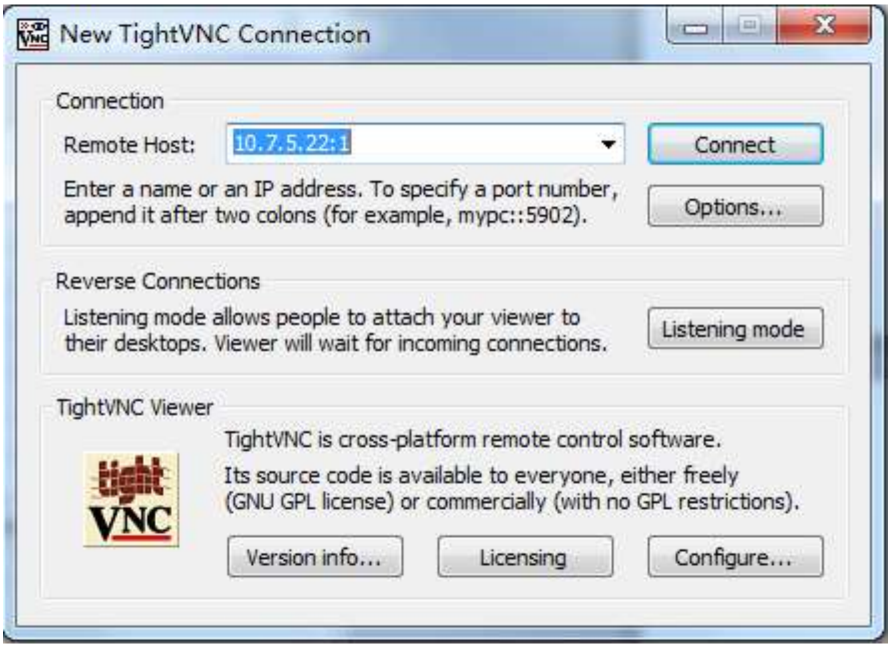
```
New 'X' desktop is tigerboard:1

Creating default startup script /home/go/.vnc/xstartup
Starting applications specified in /home/go/.vnc/xstartup
Log file is /home/go/.vnc/tigerboard:1.log
```

TightVNC asks for setting a password for the first start.

3. In your remote PC, install and launch TightViewer.

4. In the TightViewer, fill in the IP address of your TIGER Board with a desktop port number (same to the port number prompted by executing the `vncserver` command on the board) as shown in the following figure.



New TightVNC Connection

Connection

Remote Host: 10.7.5.22:1

Enter a name or an IP address. To specify a port number, append it after two colons (for example, mypc::5902).

Reverse Connections

Listening mode allows people to attach your viewer to their desktops. Viewer will wait for incoming connections.

TightVNC Viewer

TightVNC is cross-platform remote control software.

Its source code is available to everyone, either freely (GNU GPL license) or commercially (with no GPL restrictions).

Figure 12. **TightVNC Connection**

5. Click **Connect**, and enter the password that you set in step 2.

5 Multimedia Framework

GoBian is highly compatible with Kodi, GOF SDK and DMA multimedia frameworks.

This chapter describes how to set up and use them in GoBian.

5.1 Kodi

Kodi (formerly known as XBMC) is an award-winning free and open source (GPL) software media center for playing videos, music, pictures, games, and more. Kodi runs on Linux, OS X, Windows, iOS, and Android, featuring a 10-foot user interface for use with televisions and remote controls. It allows users to play and view most videos, music, podcasts, and other digital media files from local and network storage media.

To install Kodi, execute the following commands:

```
sudo apt-get update

sudo apt-get install alikodi
```

To start Kodi, execute the following commands:

```
sudo config_fbdev.sh

kodi
```

After exiting Kodi, execute the following command:

```
sudo config_x11.sh
```

5.2 GOF SDK

GOF SDK is a full set of multimedia framework dedicated for ALi SoC chipset, which is developed based on Gstreamer, OpenMAX and FFmpeg.

To install GOF SDK, execute the following commands:

```
sudo apt-get update

sudo apt-get install gof
```

After the installation is finished, GOF SDK can be found at `/ali/`.

GoBian provides a test shell, gof_test.sh at `/usr/bin`, to run GOF player on GOF SDK.

To play a local file, refer to the following example:

```
go@tigerboard:~$ gof_test.sh “file:///home/go/AVC_AAC.mp4”

Test stream [file:///home/go/AVC_AAC.mp4]

start player server

ipcrm: invalid key (100220)

CLIENT INFO:++++ client send set source command ++++

CLIENT INFO:++++ send set play url by vmsg ++++

[ PID 5246 player_server_streaming ]player_server_streaming, Adr Hld
Library version HLD3.0@C3701C_DDK5.0da.3.0_20130509x

No such file or directory

STREAMING WARNING: >>======= construct streaming end=======



---- file name : file:///home/go/AVC_AAC.mp4 ----

…
```

To play a remote file by URL, refer to the following example:

```
go@tigerboard:~$ gof_test.sh
“http://example.com/13/27/61/2131371603.0.flv”

Test stream [http://example.com/13/27/61/2131371603.0.flv]

start player server

ipcrm: invalid key (100220)

CLIENT INFO:++++ client send set source command ++++

CLIENT INFO:++++ send set play url by vmsg ++++
[ PID 5317 player_server_streaming ]player_server_streaming, Adr Hld
Library version HLD3.0@C3701C_DDK5.0da.3.0_20130509x

No such file or directory

STREAMING WARNING: >>======= construct streaming end=======



---- file name : http://example.com/13/27/61/2131371603.0.flv ----

…
```

5.3 DMA

GoBian supports for AirPlay, DLNA and Miracast after installing the ALi's DMA application that streams photos, videos and audios.

To install the DMA application, execute the following commands:

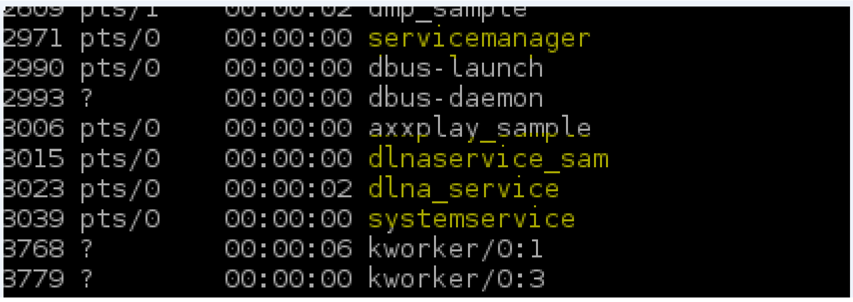
```
sudo apt-get update

sudo apt-get install alidma
```

To start the DMA service, execute the following command:

```
sudo ./alidma.sh
```

And you can find the following DLNA processes by typing `ps -A`.



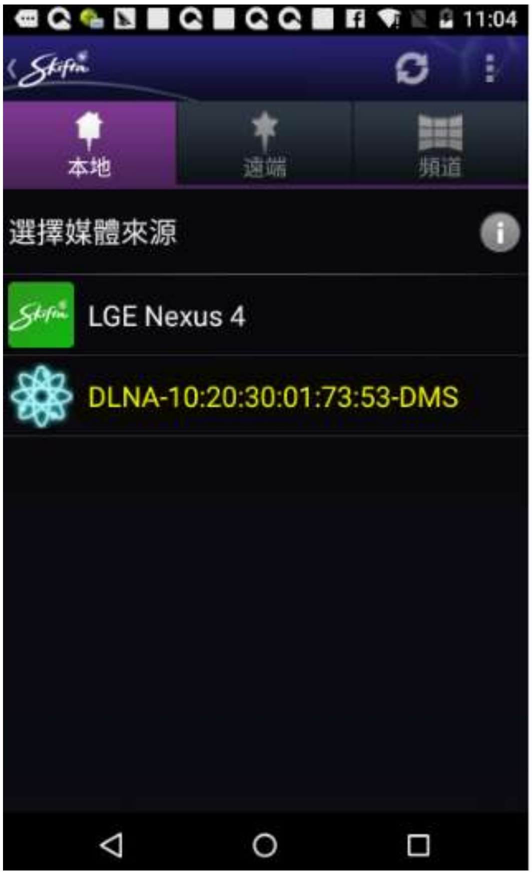
```
2869 pts/1    00:00:02 dmp_sample
2971 pts/0    00:00:00 servicemanager
2990 pts/0    00:00:00 dbus-launch
2993 ?        00:00:00 dbus-daemon
3006 pts/0    00:00:00 axxplay_sample
3015 pts/0    00:00:00 dlnaservice_sam
3023 pts/0    00:00:02 dlna_service
3039 pts/0    00:00:00 systemservice
3768 ?        00:00:06 kworker/0:1
3779 ?        00:00:00 kworker/0:3
```

5.3.1 Streaming between GoBian and Mobile Device

This section shows how to stream photos, videos and audios between GoBian and mobile device.

The following is an example procedure of viewing the multimedia files on GoBian from mobile device with Skifta installed.

1. In the mobile device, select **DLNA-10:20:30:01:73:53-DMS** as media source.



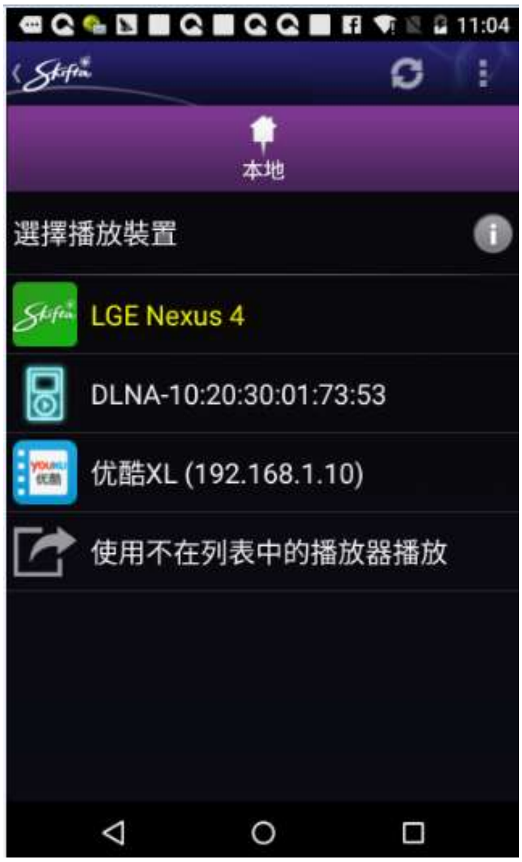
選擇媒體來源

LGE Nexus 4

DLNA-10:20:30:01:73:53-DMS

Figure 13. Selecting Media Source

2. Select **LGE Nexus 4** as media player.



11:04

本地

選擇播放裝置

LGE Nexus 4

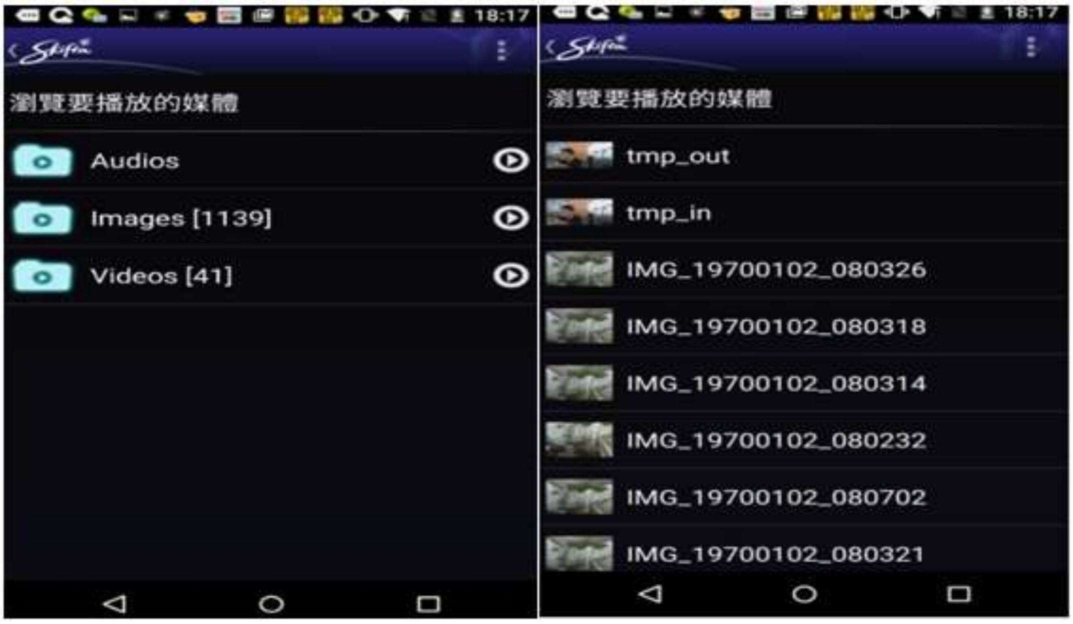
DLNA-10:20:30:01:73:53

优酷XL (192.168.1.10)

使用不在列表中的播放器播放

Figure 14. Selecting Meida Player

3. Select a multimedia file. For example, select the **Images** folder from GoBian, and select an image in the folder to view.



18:17

瀏覽要播放的媒體

Audios

Images [1139]

Videos [41]

18:17

瀏覽要播放的媒體

tmp_out

tmp_in

IMG_19700102_080326

IMG_19700102_080318

IMG_19700102_080314

IMG_19700102_080232

IMG_19700102_080702

IMG_19700102_080321

Figure 15. Selecting an Image

The following is an example procedure of streaming the multimedia files from mobile device to GoBian.

4. In the mobile device, select **LGE Nexus 4** as media source.

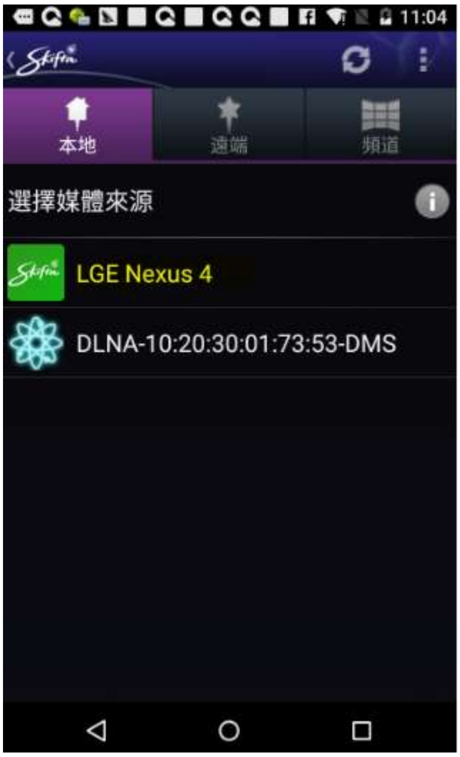


Figure 16. Selecting LGE Nexus as Media Source

5. Select **DLNA-10:20:30:01:73:53** as media player.

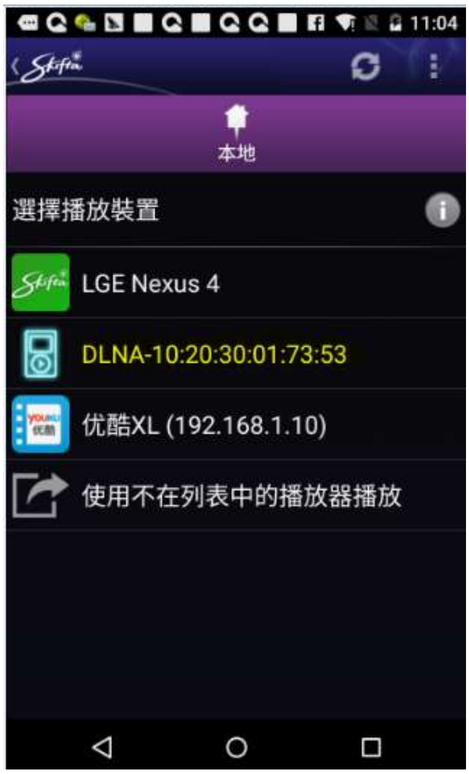


Figure 17. Selecting DLNA as Media Player

6. Select a multimedia file. For example, select the **Images** folder from mobile device, and select an image in the folder to stream.

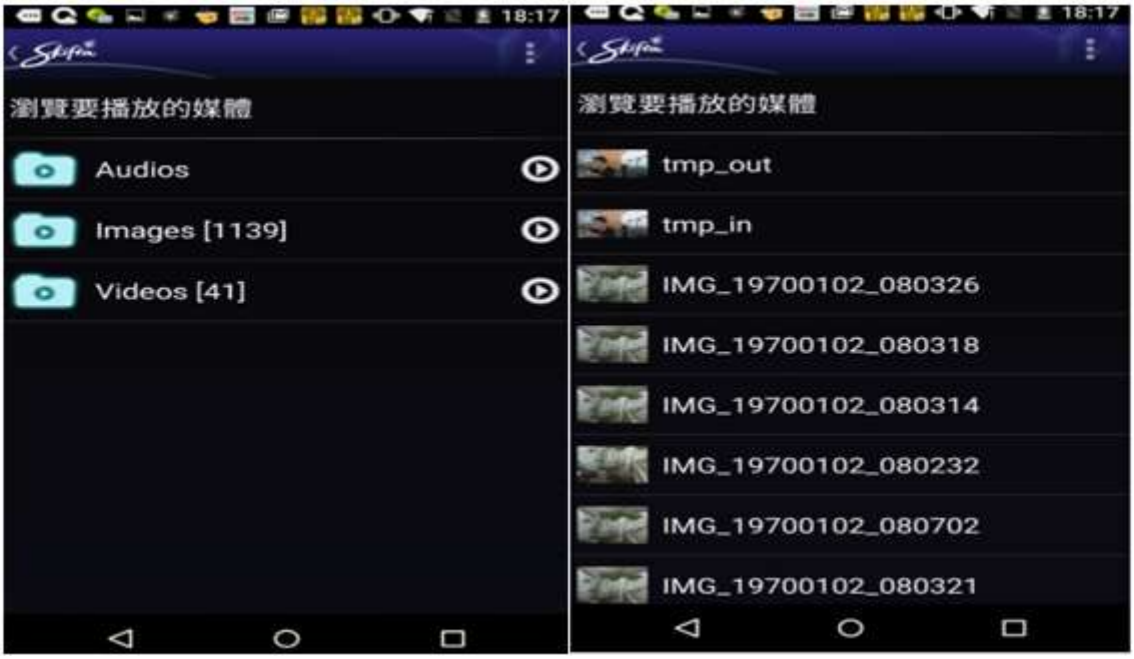


Figure 18. Selecting an Image

5.3.2 Playing with DMP

Before using DMP, run the following scripts to start the DMP service:

```
go@tigerboard ~ $ cd /opt/ali/bin/dma/

go@tigerboard /opt/ali/dma/bin $ ./dmp_test.sh

go@tigerboard /opt/ali/dma/bin $  dmp_sample.cpp (610): [dlnaservice
sample] Send command to /home/go/dmp_cmd_receiver

go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh start
```

The following is an example procedure of playing with DMP.

1. To get the DMS list, execute the following command:

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh getdms

  dmp_sample.cpp (517): sendcmd_handler:    getdms

getDms,305,device count=8

the total page is 1, current page is 1

  0: PC-JASMINE: jasmine.li:

  1: N7-GOLIATH-PENG: goliath.peng:

  2: P7-TRUEVE-HU: trueve.hu:

  3: DLNA-10:20:30:01:78:17-DMS

  4: DLNA-10:20:30:01:78:15-DMS

  5: DLNA-10:20:30:01:00:00-DMS

  6: DLNA-10:20:30:01:78:45-DMS

  7: android_dlna-DMS
```

2. Select a DMS device by executing the following command according to the device number in the list above.

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh sel=4

 dmp_sample.cpp (517): sendcmd_handler:    sel=4

dmsLs,233,there were 3 results
```

```
the total page is 1, current page is 1

   0: Audios

   1: Videos

   2: Images
```

3. Select a folder in the selected DMS device by executing the following command according to the folder number in the list above.

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh sel=1

  dmp_sample.cpp (517): sendcmd_handler:    sel=1

dmsLs,233,there were 20 results


the total page is 1, current page is 1


   0: tests

   1: phei.avi

   2: cut.ts

   3: ST_MAX3580.SearchResults

   4: LGS_demo.SearchResults

   5: mt2066.SearchResults

   6: sharp_dvb_t2.SearchResults

   7: sharp.SearchResults

   8: tda18252.SearchResults

   9: ?.?涓..涓...绗.??.?儿.?.?瀵.mkv

   10: hosts

   11: run-parts

   12: cpustats

   13: file_contexts

   14: seapp_contexts

  15: property_contexts
  16: aging_test.mpg

  17: error.ts

  18: factory.ts

  19:
bbb_short.ffmpeg.480x360.mp4.libx264_500kbps_25fps.libfaac_stereo_
128kbps_44100Hz.mp4
```

4. To go to another page in the folder or DMS list, executing the following command to specify a page number:

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh page=2

 dmp_sample.cpp (517): sendcmd_handler:    page=2



the total page is 26, current page is 2



  30: shachenbao.png

  31: yangsha.png

  32: qing.png

  33: logo_clearmemory.png

  34: green.png

  35: orange_height.png

  36: orange.png

  37: daxue.png

  38: red.png

  39: qiangshachenbao.png

  40: logo_douban.png

  41: xiaoxue_zhongxue.png

  42: shortcut_add.png

  43: logo_dlna.png

  44: lightgreen_long.png
  45: xiaoyu.png

  46: baoxue.png

  47: fuchen.png

  48: purple_height.png

  49: grid_bg.png

  50: xiaoxue.png

  51: latest_1.png

  52: latest_2.png

  53: fg_live_bottom.png

  54: lightgreen.png

  55: green_height.png

  56: red_long.png

  57: dayu_baoyu.png

  58: orange_long.png

  59: wallpaper.png
```

5. To go back to the upper directory, execute the following command:

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh return

 dmp_sample.cpp (517): sendcmd_handler:    return

dmsLs,233,there were 3 results


the total page is 1, current page is 1


   0: Audios

   1: Videos

   2: Images
```

6. To get the instruction of the command, execute the following command:

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh help
```

```
 dmp_sample.cpp (517): sendcmd_handler:    help

    -h --help


    *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

    Command Description:

    "echo start > /home/pi/dmp_cmd_receiver":  Start DMP service.

    "echo getdms > /home/pi/dmp_cmd_receiver":  get dms from dms list.

    "echo stop > /home/pi/dmp_cmd_receiver":  Stop DMP service.

    "echo sel='number' > /home/pi/dmp_cmd_receiver":  chose the DIR or
File.

    "echo  page='number' > /home/pi/dmp_cmd_receiver":  show the
files of the page='number'.

    "echo return > /home/pi/dmp_cmd_receiver":  show the files of the
page='number'.

    "echo help > /home/pi/dmp_cmd_receiver":  Display help guide.

    *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

7. To shut down the media controller, execute the following command:

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh stop

 dmp_sample.cpp (517): sendcmd_handler:    stop
```

8. To stop the playing an audio and video file, execute the following command:

```
go@tigerboard /opt/ali/bin/dma $ ./dmp_cmd.sh close

 dmp_sample.cpp (517): sendcmd_handler:    close
```

6 Interfacing TIGER Board

GoBian integrates the RPi.GPIO Python library out-of-the-box. You can use Python directly to access the pins on the board, or port a Python project from Raspberry Pi to TIGER Board without too much effort. This chapter provides project samples for getting you started to program with Python on TIGER Board.

6.1 Blinking an LED

In this project, we are going to blink an LED with Python.

6.1.1 What You Will Need

Prepare the following electronic devices:

- A breadboard
- A 500 ohm resistor
- An LED
- Jumper wires

6.1.2 Wiring Up to TIGER Board

Follow the steps below to get your board wired up:

1. Connect the negative rail of the breadboard to the ground pin of the TIGER Board, for example pin39 on J3, with a jumper wire.
2. Place a 500 ohm resistor, and connect one side to negative rail of the breadboard.
3. Place an LED in the breadboard so that the cathode side (the shorter wire) is connected to the other side of 500 ohm resistor.

4. Connect pin33 on J3 to the anode side of the LED with a jumper wire.
5. Connect the power source pins as pin1 on J3 to the positive rail of breadboard with a jumper wire.

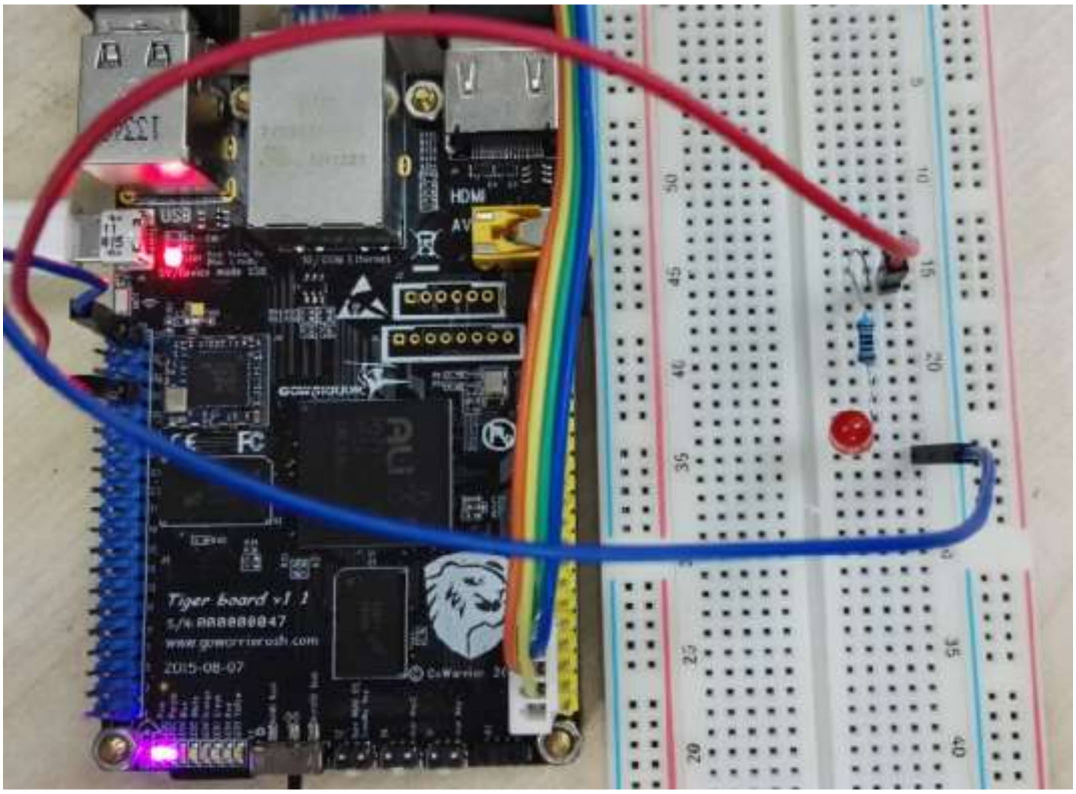


Figure 19. Wiring Up

6.1.3 Coding with Python

Open a new text file in File Manager, and copy the following code to the file:

```
import RPi.GPIO as gpio  # Include the RPi.GPIO library as gpio.
import time             # Include the Python time functions.

gpio.setmode(gpio.BOARD) # Set mod as BOARD, which means to use the numbering from the physical layout.
while True:
    gpio.setup(33, gpio.OUT)          # Set pin 33 on J3 to be an output.
    gpio.output(33, gpio.HIGH)        # Set pin 33 on J3 as high.
    time.sleep(1)                    # Wait for one second.
```

```
gpio.output(33, gpio.LOW)           # Set pin 33 on J3 as low.


    time.sleep(1)                 # Wait for one second.
```

Save the file as blink.py, and run the script by the following command:

```
go@tigerboard~ $ python blink.py
```

Notice that the LED starts to blink. You can modify the script to change the LED blink frequency by using decimals in the `time.sleep()` functions.

6.2 Reading a Button

In this project, we are going to read the state of a button with Python.

6.2.1 What You Will Need

Prepare the following electronic devices:

- A breadboard
- A push button
- A 10K pulldown resistor

6.2.2 Wiring Up to TIGER Board

Follow the steps below to get your board wired up:

1. Place a push button into the breadboard so that it straddles the center channel.
2. Connect one lead of the button to the positive rail.
3. Connect the other lead of the button to the pin33 on J3.
4. Connect a 10K pulldown resistor from the ground rail to the button lead that connects to the pin33 on J3.

6.2.3 Coding with Python

Open a new text file in the File Manager in GoBian, and copy the following code to the file:

```
import RPi.GPIO as gpio
import time


gpio.setmode(gpio.BOARD)
while True:
    gpio.setup(33, gpio.IN)           # Set up pin 33 on header J3 to
be used as an input.
    if gpio.input(33):                # Read the value of pin 33; and
execute the indented if it’s high.
      print “Button pressed!”          # Print a message to the terminal.


    time.sleep(0.1)                   # Wait for 0.1 second.
```

Save the file as button.py, and run the script by the following command:

```
go@tigerboard~ $ python button.py              # Wait for one second.
```

Notice that the screen prompts “Button pressed!” when you press the button.

6.3 Controlling an LED Brightness

In this project, we are going to control an LED's brightness.

Before the project, prepare the electronic devices and wire them up to the TIGER Board as described in the project of blinking an LED.

Open a new text file in File Manager, and copy the following code to the file:

```
import RPi.GPIO as gpio


led_pin = 33
gpio.setmode(gpio.BOARD)
gpio.setup(led_pin, gpio.OUT)


pwm_led = gpio.PWM(led_pin, 500)  # Create a PWM instance, first
```

```
parameter is pin, second parameter is frequent.

pwm_led.start(100)  # Start PWM, the parameter 100 here is duty cycle.


while True:

        duty_s = raw_input("Enter Brightness (0 to 100)")  # Get input.

        duty =100- int(duty_s) # change string to int.

        pwm_led.ChangeDutyCycle(duty) # Change the duty cycle to set the
led’s brightness.
```

Save the file as brightness.py, and run the script by the following command:

```
go@tigerboard~ $ python brightness.py
```

Notice the LED achieves its full brightness at first, then fades down according to your input for the brightness value, and fades up or down according to your next input and so on.

7 GoBian Kernel Building

The PDK source code of GoBian is available on GitHub for free download, which allows you to customize GoBian in a Kernel level.

7.1 Setting up Building Environment

Before building your own GoBian, set up your local work environment for both hardware and software.

7.1.1 Hardware Environment Deployment

The following table shows the recommended server hardware configurations.

Server	Description
CPU	Intel® Xeon® processor, more than 8 cores
Memory	4 GB
Hard disk	32 GB
Network	10/100M Ethernet

Table 6 Hardware Configurations

7.1.2 Software Environment Deployment

The building server requires the Ubuntu 64-bit as operating system with Oracle Java SE Development Kit 6u45 installed.

Follow the steps below to set up the software environment on Ubuntu:

1. Install the basic tools by executing the following commands:

```
$ sudo apt-get install git gnupg flex bison gperf build-essential \
  zip curl libc6-dev libncurses5-dev:i386 x11proto-core-dev \
  libx11-dev:i386 libreadline6-dev:i386 libgl1-mesa-glx:i386 \

  libgl1-mesa-dev g++-multilib mingw32 tofrodos \

  python-markdown libxml2-utils xsltproc zlib1g-dev:i386 \

  libuuid1:i386

$ sudo ln -s /usr/lib/i386-linux-gnu/mesa/libGL.so.1
/usr/lib/i386-linux-gnu
```

2. Download JDK 6 package by executing the following command:

```
git clone https://github.com/GoWarrior/tools_jdk-6u45-linux-x64.git
```

3. Install JDK 6 by executing the following command:

```
$ chmod a+x jdk-6u45-linux-x64.bin

$ ./jdk-6u45-linux-x64.bin
```

4. Create jdk to the system directory.

```
$ sudo mkdir /usr/lib/jvm

$ sudo tar cvf - jdk1.6.0_45 | (cd /usr/lib/jvm; tar xvf - )
```

5. Set Oracle JDK6 to the default JDK by executing the following commands:

```
$ sudo update-alternatives --install /usr/bin/javac javac
/usr/lib/jvm/jdk1.6.0_45/bin/javac 1

$ sudo update-alternatives --install /usr/bin/java java
/usr/lib/jvm/jdk1.6.0_45/bin/java 1

$ sudo update-alternatives --install /usr/bin/javaws javaws
/usr/lib/jvm/jdk1.6.0_45/bin/javaws 1

$ sudo update-alternatives --install /usr/bin/jar jar
/usr/lib/jvm/jdk1.6.0_45/bin/jar 1

$ sudo update-alternatives --install /usr/bin/javadoc javadoc
/usr/lib/jvm/jdk1.6.0_45/bin/javadoc 1

$ sudo update-alternatives --install /usr/bin/javap javap
/usr/lib/jvm/jdk1.6.0_45/bin/javap 1
```

6. Use the update-alternatives command to confirm that each java command points to Oracle JDK.

```
![Configuring JDK](../../)
```

The above mentioned is a typical example. If both Oracle JDK and Open JDK have been installed on the server, make sure to select Oracle JDK and execute all of the following commands.

```
$ sudo update-alternatives --config javac

$ sudo update-alternatives --config javaws

$ sudo update-alternatives --config java

$ sudo update-alternatives --config jar

$ sudo update-alternatives --config javadoc

$ sudo update-alternatives --config javap
```

7.2 Installing Kernel Toolchains

Procedure:

1. Download the ARM and MIPS cross compiling toolchain packages by executing the following commands:

```
# mkdir -p $HOME/ali_toolchain

# cd $HOME/ali_toolchain

# git clone https://github.com/GoWarrior/tools_arm-linux-gnueabi.git
arm_toolchain

# git clone https://github.com/GoWarrior/mips-sde-elf.git
mips_toolchain

# ls  arm_toolchain

  gcc-linaro-arm-linux-gnueabi-2012.01-20120125_linux.tar.gz

# ls mips_toolchain

  mips-4.3-221-mips-sde-elf-i686-pc-linux-gnu.tar.bz2
```

2. Unzip the package and configure the environment variable PATH by executing the following commands:

```
# cd arm_toolchain

# tar xzvf gcc-linaro-arm-linux-gnueabi-2012.01-20120125_linux.tar.gz

# PATH=$HOME/ali_toolchain/arm_toolchain/arm-linux-gnueabi/bin:$PATH

# export PATH

# cd ..

# cd mips_toolchain

# tar xjvf mips-4.3-221-mips-sde-elf-i686-pc-linux-gnu.tar.bz2

# PATH=$HOME/ali_toolchain/mips_toolchain/mips-4.3/bin:$PATH

# export PATH
```

7.3 Getting PDK Source Code

Procedure:

1. Download the PDK source code by executing the following command:

```
git clone https://github.com/GoWarrior/PDK_GoBian.git
```

2. After downloading, switch to the root directory of PDK, and then change its branch to gobian-1.0 by executing the following command:

```
git checkout gobian-1.0
```

7.4 Building Kernel

The sections below describe how to the GoBian kernel step by step. After this compilation, the output files need to be copied to the SD card with GoBian installed to replace the original kernel files, as instructed in the following sections.

7.4.1 Compiling Bootloader (U-Boot)

Procedure:

1. Access the U-Boot directory.

```
$ cd uboot
```

2. Configure U-Boot version.

Update the U-Boot version number in the file `include/configs/ali_3921.h` if there is any update in U-Boot.

```
$ vim include/configs/ali_3921.h

#define BOOTLOADER_VERSION "ALiBoot1.2.0"
```

Note

The naming rule of U-Boot version is customizable according to actual requirements.

3. Compile.

```
$ make rebuild_3921
```

4. Output.

Replace the following files in the `/boot` folder in your SD card with the ones in the `u-boot/uboot_merger/boot_files` folder:

```
baseparams.abs

bootargs.abs

bootcode.abs

bootmedia.ubo

cmdline.txt

deviceinfo.abs
```

7.4.2 Compiling Audio Engine

Procedure:

1. Switch to the audio engine directory by executing the following command:

```
$ cd /tds/ae_sdk/src/ae/m36f_linux
```

2. Compile the audio engine module by executing the following commands:

```
$ make path

$ make config_3921_release

$ make generate_ae
```

7.4.3 Compiling SEE

Procedure：

1. Switch to the SEE directory by executing the following command:

```
$ cd /tds/see_sdk/src/see/m36f_linux
```

2. Compile the SEE module by executing the following commands:

```
$ make path

$ make config_3921_release

$ make generate_see
```

7.4.4 Compiling Linux Kernel

Procedure：

1. To add or delete a configuration item for kernal, modify the file `pdk/linux/kernel/boards/TIGER_BOARD/defkernelconfig_linux-3.18.16`, or execute the following command:

```
$ cd pdk/linux/

$ make BOARD=TIGER_BOARD KERNELVER=linux-3.18.16 linux-config
```

2. Run the following script to copy the files (`uboot_*.abs`, `ae_bin.abs` and `see_bin.abs`) required by kernel compilation:

```
$ cd pdk/linux

$./ cp_bootabs_release.sh TIGER_BOARD
```

3. Execute the following command to compile the kernel:

```
$ cd pdk/linux

$ make BOARD= TIGER_BOARD KERNELVER=linux-3.18.16
```

Upon completion of compilation, notice that the output files are in the `pdk/linux/install/bin` directory.

4. Replace the following files in the `/boot` folder in your SD card with the ones in `pdk/linux/install/bin` folder:

```
ae_bin.ubo

main_bin.ubo

see_bin.ubo
```

Appendix: Glossary

Name	Description
GoBian	GoBian is the integral Linux-based operating system of GoWarrior platform.

Table 7. List of Glossary

Revision History

Document Change History

Revision	Changes	Date
v1.01	Updated GoBian PDK source code download commands. Updated the Kernel compile steps.	March 18, 2016
v1.0	Initial Release	December 18, 2015

Table 8. Document Change History

Software Changes

Revision	Changes	Date
v1.0	Initial Release	December 11, 2015

Table 9. Software Change History

www.gowarriorosh.com

Headquarters

Tel: +886-2-8752-2000

Fax: +886-2-8751-1001