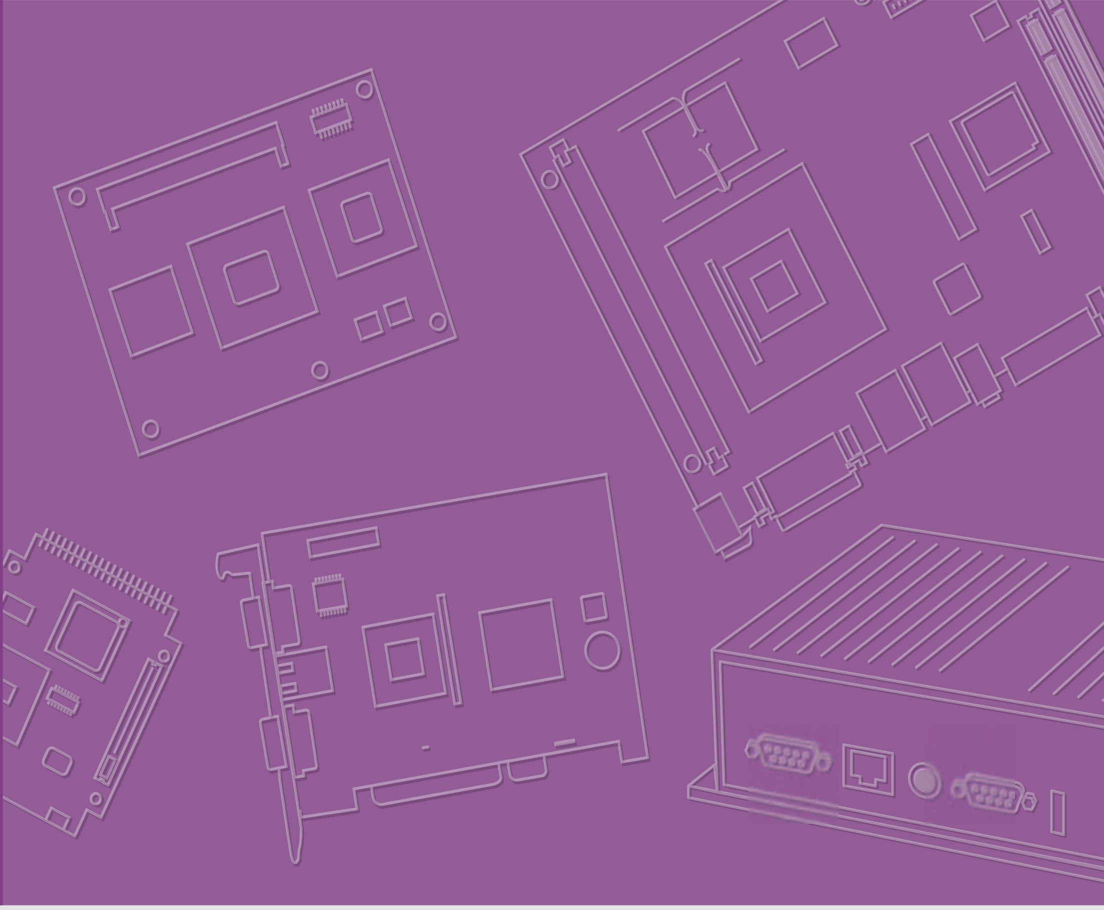


User Manual

EPC-R6410

NXP ARM$^{®}$ Cortex$^{®}$-A9 i.MX6Dual/Quad Computer Box

ADVANTECH

Enabling an Intelligent Planet

Copyright

The documentation and the software included with this product are copyrighted 2017 by Advantech Co., Ltd. All rights are reserved. Advantech Co., Ltd. reserves the right to make improvements in the products described in this manual at any time without notice. No part of this manual may be reproduced, copied, translated or transmitted in any form or by any means without the prior written permission of Advantech Co., Ltd. Information provided in this manual is intended to be accurate and reliable. However, Advantech Co., Ltd. assumes no responsibility for its use, nor for any infringements of the rights of third parties, which may result from its use.

Acknowledgements

ARM is a trademark of the ARM Corporation.

NXP is a trademark of the NXP Corporation.

All other product names or trademarks are properties of their respective owners.

Product Warranty (2 years)

Advantech warrants to you, the original purchaser, that each of its products will be free from defects in materials and workmanship for two years from the date of purchase.

This warranty does not apply to any products which have been repaired or altered by persons other than repair personnel authorized by Advantech, or which have been subject to misuse, abuse, accident or improper installation. Advantech assumes no liability under the terms of this warranty as a consequence of such events.

Because of Advantech's high quality-control standards and rigorous testing, most of our customers never need to use our repair service. If an Advantech product is defective, it will be repaired or replaced at no charge during the warranty period. For out-of-warranty repairs, you will be billed according to the cost of replacement materials, service time and freight. Please consult your dealer for more details.

If you think you have a defective product, follow these steps:

1. Collect all the information about the problem encountered. (For example, CPU speed, Advantech products used, other hardware and software used, etc.) Note anything abnormal and list any onscreen messages you get when the problem occurs.
2. Call your dealer and describe the problem. Please have your manual, product, and any helpful information readily available.
3. If your product is diagnosed as defective, obtain an RMA (return merchandize authorization) number from your dealer. This allows us to process your return more quickly.
4. Carefully pack the defective product, a fully-completed Repair and Replacement Order Card and a photocopy proof of purchase date (such as your sales receipt) in a shippable container. A product returned without proof of the purchase date is not eligible for warranty service.
5. Write the RMA number visibly on the outside of the package and ship it prepaid to your dealer.

Part No. 2006C64130 Edition 1
Printed in Taiwan August 2017

Declaration of Conformity

FCC Class B

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Packing List

Before installation, please ensure the following items have been shipped.

- 1 x EPC-R6410 Computer Box
- 2 x wall mount
- 1 x China rohs
- 1 x user manual_CH

Optional Accessories

Part No.	Description
96PSA-A36W12R1	A/D 100 ~ 240 V, 36 W, 12 V
1700001524	Power cord 3P UL 10A 125 V 180 cm
170203180A	Power Code 3P UK 2.5A/3A 250V 1.83cm
170203183C	Power Code 3P Europe 183cm
IDK-1107WR-40WVA1E	7" LED PANEL 400N with 4WR touch, 800 x 480 (G)
1700025779-01	LVDS cable for IDK-1107WR
1700025767-01	LCD black light cable for IDK-1107WR
SQF-ISDS1-4G-82C	SQFlash SD card 4G
968AD00038	Quectel 4G module EC20-CEFA kit w/ antenna
968AD00039	Wi-Fi/BT module AW-NB136 kit w/ antenna
PCIE-1602C	2-Port RS232/422/485 PCIE card

Ordering Information

Part No.	Description
EPC-R6410CD-PAA1E	NXP i.MX6Dual A9 RISC computer box
EPC-R6410CQ-VAA1E	NXP i.MX6Quad A9 RISC computer box

Safety Instructions

1. Read these safety instructions carefully.
2. Keep this User Manual for later reference.
3. Disconnect this equipment from any AC outlet before cleaning. Use a damp cloth. Do not use liquid or spray detergents for cleaning.
4. For plug-in equipment, the power outlet socket must be located near the equipment and must be easily accessible.
5. Keep this equipment away from humidity.
6. Put this equipment on a reliable surface during installation. Dropping it or letting it fall may cause damage.
7. The openings on the enclosure are for air convection. Protect the equipment from overheating. DO NOT COVER THE OPENINGS.
8. Make sure the voltage of the power source is correct before connecting the equipment to the power outlet.
9. Position the power cord so that people cannot step on it. Do not place anything over the power cord.
10. All cautions and warnings on the equipment should be noted.
11. If the equipment is not used for a long time, disconnect it from the power source to avoid damage by transient overvoltage.
12. Never pour any liquid into an opening. This may cause fire or electrical shock.
13. Never open the equipment. For safety reasons, the equipment should be opened only by qualified service personnel.
14. If one of the following situations arises, get the equipment checked by service personnel:
    - The power cord or plug is damaged.
    - Liquid has penetrated into the equipment.
    - The equipment has been exposed to moisture.
    - The equipment does not work well, or you cannot get it to work according to the user's manual.
    - The equipment has been dropped and damaged.
    - The equipment has obvious signs of breakage.

DISCLAIMER: This set of instructions is given according to IEC 704-1. Advantech disclaims all responsibility for the accuracy of any statements contained herein.

Safety Precaution – Static Electricity

Follow these simple precautions to protect yourself from harm and the products from damage.

- To avoid electrical shock, always disconnect the power from your PC chassis before you work on it. Don't touch any components on the CPU card or other cards while the PC is on.

Contents

Chapter 1

General Introduction

This chapter gives background information on the EPC-R6410.

Sections include:

- Introduction
- Specifications

1.1 Introduction

EPC-R6410 is a RISC based box computer with integrated NXP i.MX6D/Q Cortex-A9 1.0GHz processor. It is designed for applications that require high performance and rich I/O, but low power consumption.

The EPC-R6410 offers rich I/O connectivity with 6 USB 2.0, 5 COM, 8 GPIOs, and 1 CAN. The system also supports dual display of HDMI and VGA. EPC-R6410 also features wireless connectivity, including 1 M.2 E-key slot for Wi-Fi and BT module, and 1 Mini-PCIe for 3G/4G module.

EPC-R6410 is not only the best hardware solution for industrial market, but also be flexible enough to be applied to variant applications such as kiosks, ATM/VTMs, industrial controllers and IoT device related applications.

1.2 Product Features

1.2.1 Key Features

Processor: Freescale i.mx6 Series

- ARM Cortex™-A9 high performance processor, up to 1 GHz
- Supports 1 IPU, OpenGL ES 2.0 for 3D BitBLT for 2D and OpenVG™ 1.1
- Video decoder: MPEG-4 ASP, H.264 HP, H.263, MPEG-2 MP, MJPEG BP
- Video encoder: MPEG-4 SP, H.264 BP, H.263, MJPEG BP

System Memory Support

- DDR3 800 MHz
- Capacity: on board DDR3 1/2G GB

Gigabit Ethernet

- Chipset: NXP i.MX6 integrated RGMII
- 1 x10/100/1000 Mbps

Peripheral Interface

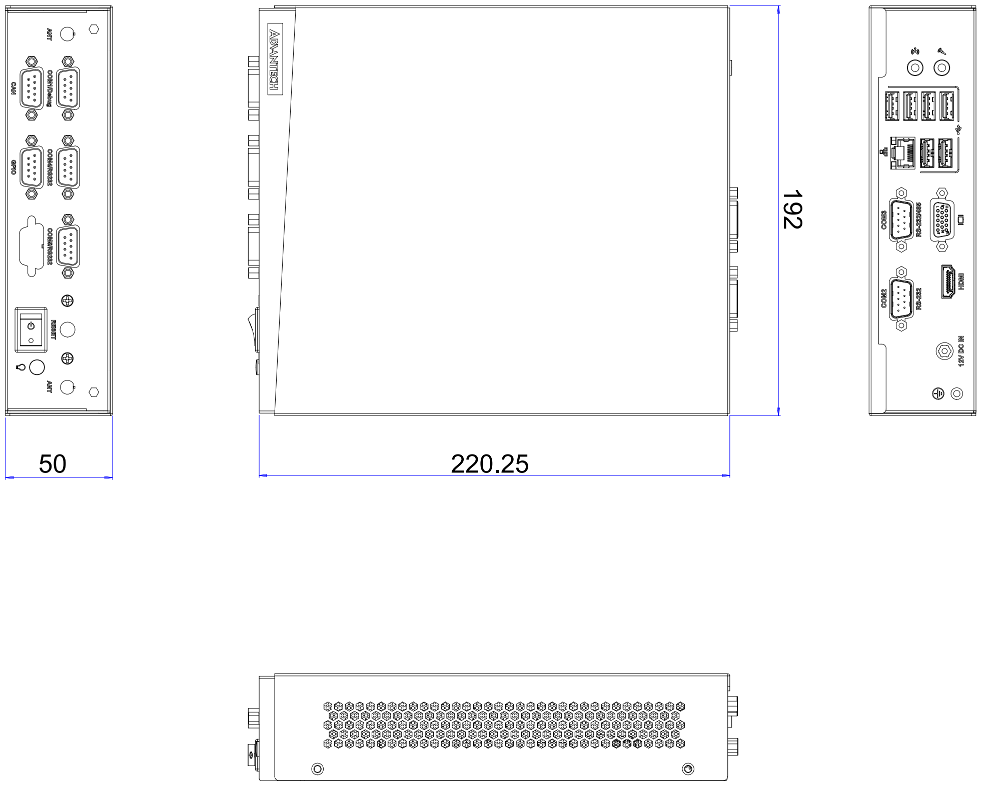
- 1 x VGA
- 1 x HDMI
- 6 x USB2.0 Type A
- 1 x SD Slot
- 5 x serial (3 of RS-232 w/ 4wire,1 of RS-232/422/485 w/ 4wire, 1 of RS232(default)/Debug port by Jumper option)
- 1 x miniPCIe slot
- 1 x SIM Slot
- 1 x M.2 E-Key

OS Support

- Android V4.4.2, Kernel V3.0.35
- Linux Yocto1.7, Kernel V3.14.28

1.3 Mechanical Specifications

- **Dimension:** 210 x 195 x 50 mm
- **Reference Weight:** 2.26kg (including whole package)



1.4 Electrical Specifications

- **Power supply type:** DC-in 12V
- **RTC battery:**
  - Typical voltage: 3V
  - Normal discharge capacity: 240 mAh

1.5 Environmental Specifications

- **Operating temperature:** 0~60°C (32~140°F)
- **Operating humidity:** 0% ~ 90% relative humidity, non-condensing
- **Storage temperature:** -40~85°C (-40~185°F)
- **Storage humidity:** 60°C @ 95% RH Non-condensing

Chapter 2

H/W Installation

This chapter gives mechanical and connector information on the EPC-R6410.

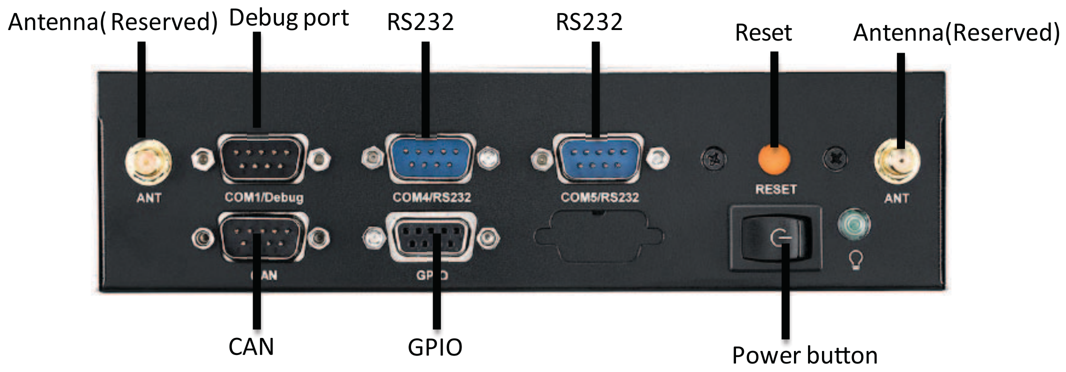
Sections include:

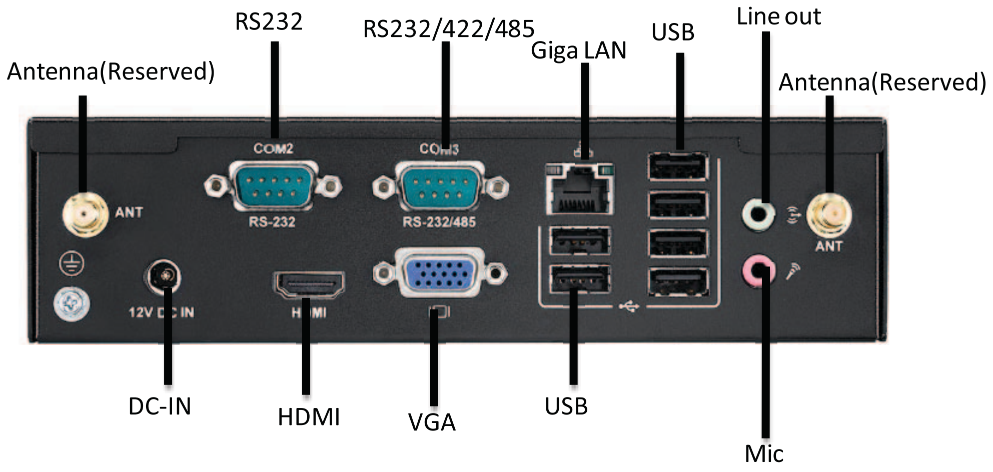
- ■ Connector Information
- ■ Mechanical Drawing
- ■ Quick Start Guide

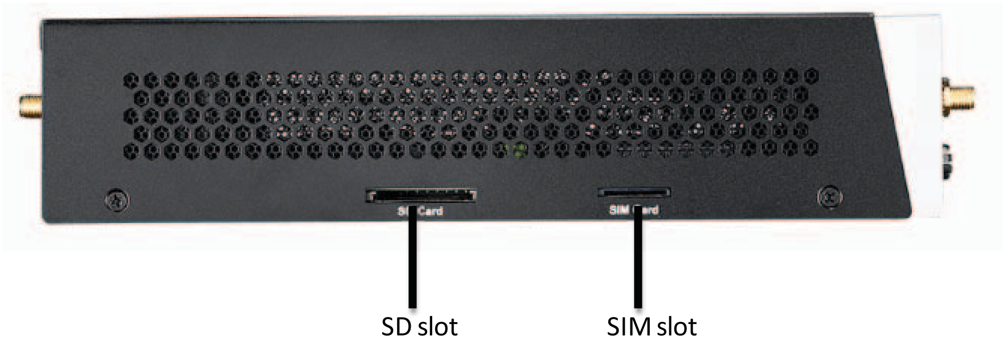
2.1 Introduction

The following sections show the external connectors and pin assignments for applications.

2.2 EPC-R6410 IO Overview



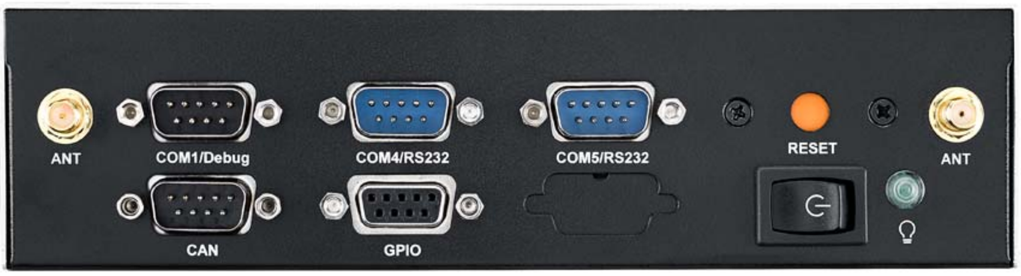




2.3 Connectors

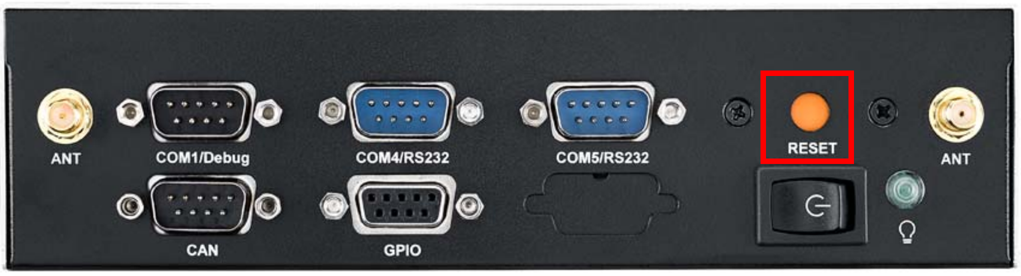
2.3.1 Power button

EPC-R6410 supports ATX mode and there is a power button for system power on/off. Please press over 1 second to power on/off system.



2.3.2 Reset button

Here is reset button for system reset.



2.3.3 COM1/Debug

COM1 of EPC-R6410 can be used as RS232 and Debug port by jumper select and default setting as debug port. You can set CN18 on board to configure the work mode. Pins are defined below

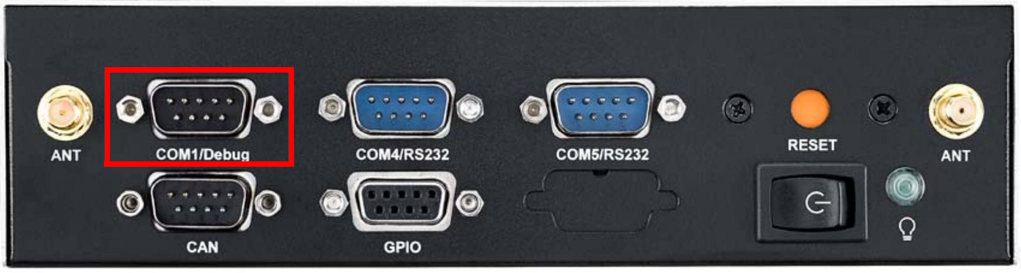


Table 2.1: COM1 pin define

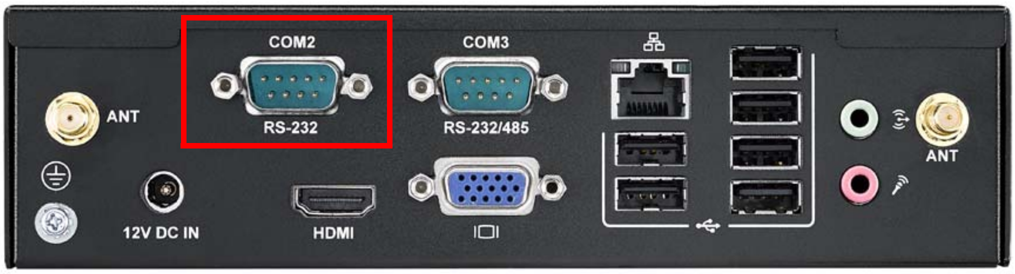
Pin	Description
1	NC
2	RX
3	TX
4	NC
5	GND
6	NC

7	NC
8	NC
9	NC

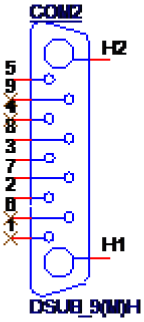
CN18	COM0 RS232 and debug mode select
Part Number	1653003101
Footprint	HD_3x1P_79_D
Description	PIN HEADER 3x1P 2.0mm 180D(M) DIP 2000-13 WS
Setting	function
(1-2)	COM0 is debug port (default)
(2-3)	COM0 is RS232 COM port

2.3.4 COM4/COM5/COM2 RS232

EPC-R6410 supports 3 x RS232 with 4wires, as COM2, COM4, COM5. Pins are defined below:



Pin	Description
1	NC
2	COM2_RX
3	COM2_TX
4	NC
5	GND
6	NC
7	COM2_RTS
8	COM2_CTS
9	NC



COM port

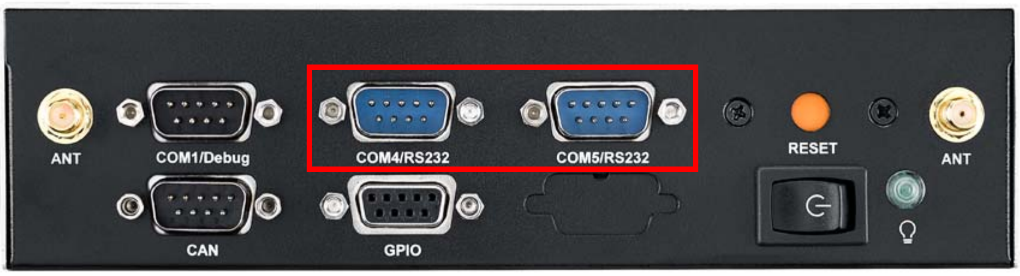


Table 2.2: COM4 pin define

Pin	Description
1	NC
2	RXD
3	TXD
4	NC
5	GND
6	NC
7	RTS
8	CTS
9	NC

Table 2.3: COM5 pin define

Pin	Description
1	NC
2	RXD
3	TXD
4	NC
5	GND
6	NC
7	RTS
8	CTS
9	PWR

2.3.5 CAN

EPC-R6410 supports 1xCAN; pins defined below:

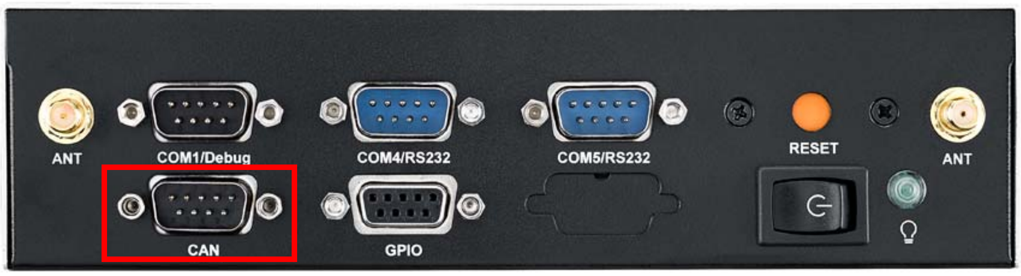


Table 2.4: CAN

Pin	Description
1	NC
2	CAN_D+
3	NC
4	NC
5	GND
6	NC
7	CAN_D-
8	NC
9	NC

2.3.6 GPIO

EPC-R6410 supports 8 x GPIO via a DB9 connector. Pins are defined below:

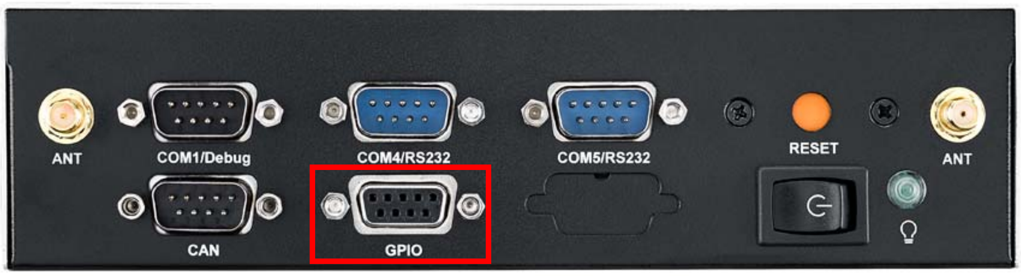
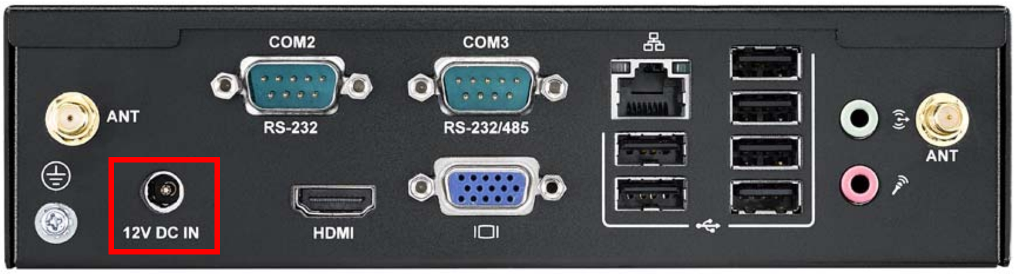


Table 2.5: GPIO

Pin	Description
1	GPIO0
2	GPIO1
3	GPIO2
4	GPIO3
5	GPIO4
6	GPIO5
7	GPIO6
8	+V_GPIO
9	GND

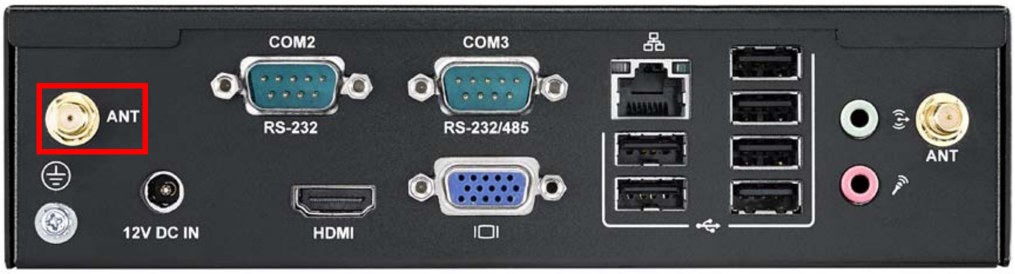
2.3.7 DC-In

EPC-R6410 support 12V DC in by DC-Jack.



2.3.8 Antenna

Here are total four reserved openings for antenna to support Wi-Fi/3G/4G modules.



2.3.9 COM3 RS232/422/485

COM3 can be configure to RS232 or RS422 or RS485 by jumper. You can set CN26 and JSETCOM1 and J422T1 and J384T1 to select the work mode.

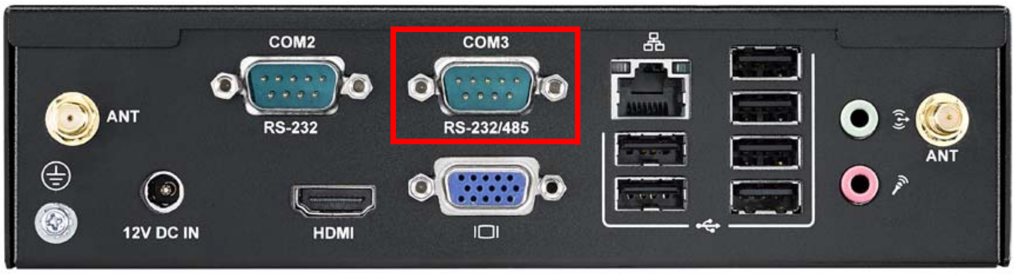
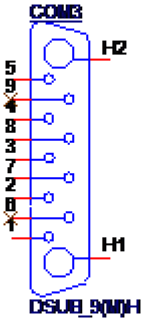


Table 2.6: COM3 pin define

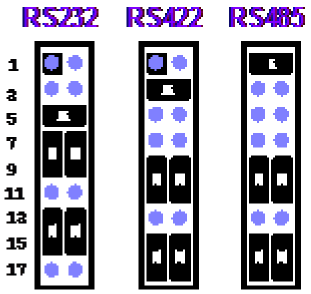
Pin	Description
1	COM3_DCD/RS485 D-
2	COM3_RX/RS485 D+
3	COM3_TX
4	COM3_DTR
5	GND
6	NC
7	COM3_RTS
8	COM3_CTS
9	NC

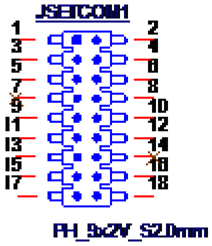


COM port pin header

2.3.9.1 Jumper setting: UART3 RS232, RS422, RS485 signal select

JSETCCOM1	UART3 RS232, RS422, RS485 signal select
Part number	1653009260
Footprint	HD_9x2P_79
Description	PIN HEADER 9x2P 2.0mm 180D(M) SMD 21N22050
Setting	Function
(5-6 7-9 8-10 13-15 14-16)	RS232 mode (CN26 must be set RS232 at the same time) (default)
(3-4 9-11 10-12 15-17 16-18)	RS422 mode
(1-2 9-11 10-12 15-17 16-18)	RS485 mode (CN26 must be set RS485 at the same time)

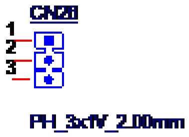




2.3.9.2 Jumper setting: COM3 RS232 and RS485mode detect

CN26	COM3 RS232 and RS485mode detect
Part Number	1653003101
Footprint	HD_3x1P_79_D
Description	PIN HEADER 3x1P 2.0mm 180D(M) DIP 2000-13 WS
Setting	function
(1-2)	RS232 mode (default)
(2-3)	RS485 mode

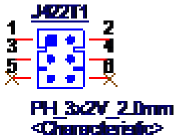
This pin header is designed for selecting COM3 RS232 and RS485 modes.



2.3.9.3 Jumper setting: RS422/RS485 impedance detect

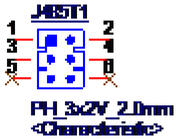
J422T1	COM3 RS-422 impedance select
Part Number	1653003201-01
Footprint	HD_3x2P_79_D_PRX
Description	PIN HEADER 2X3P 2.00mm 180D(M) DIP 1140-010-06SN
Setting	Function
(1-3,2-4)	Impedance on (default)
(3-5,4-6)	Impedance off

This pin header is designed for RS232 impedance on/off.



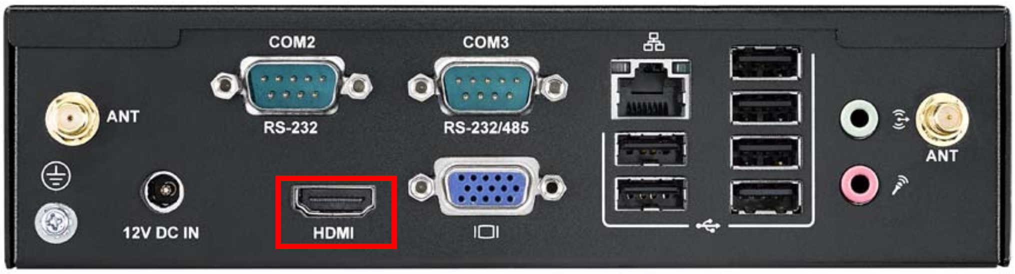
J485t1	COM3 RS-485 impedance select
Part Number	1653003201-01
Footprint	HD_3x2P_79_D_PRX
Description	PIN HEADER 2X3P 2.00mm 180D(M) DIP 1140-010-06SN
Setting	Function
(1-3,2-4)	Impedance On (default)
(3-5,4-6)	Impedance off

This pin header is designed for RS485 impedance on/off.

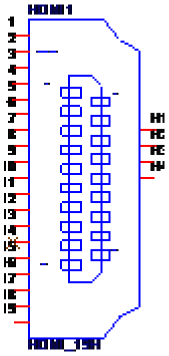


2.3.10 HDMI

EPC-R6410 supports HDMI; pin definition below:

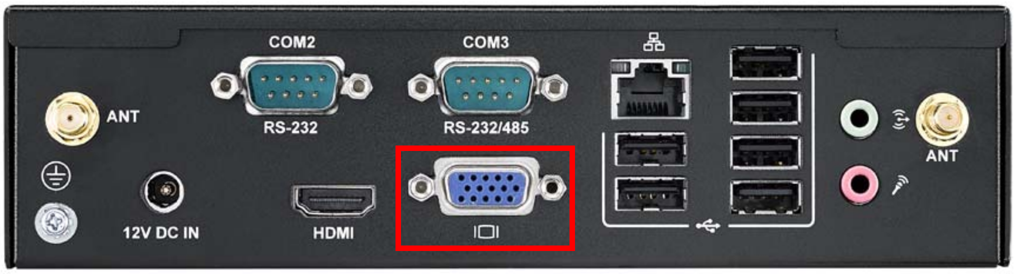


Pin	Description	Pin	Description
1	HDMI_Z_TD2_P	11	GND
2	GND	12	HDMI_Z_CLK_N
3	HDMI_Z_TD2_N	13	HDMI_CEC_OUT
4	HDMI_Z_TD1_P	14	NC
5	GND	15	HDMI_DDC_CLK
6	HDMI_Z_TD1_N	16	HDMI_DDC_DAT
7	HDMI_Z_TD0_P	17	GND
8	GND	18	+5V_HDMI
9	HDMI_Z_TD0_N	19	HDMI_HPD_OUT
10	HDMI_Z_CLK_P		

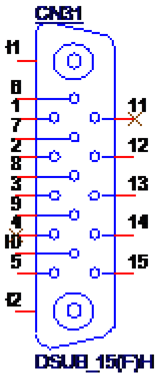


2.3.11 VGA

EPC-R6410 supports VGA; pin definitions below:

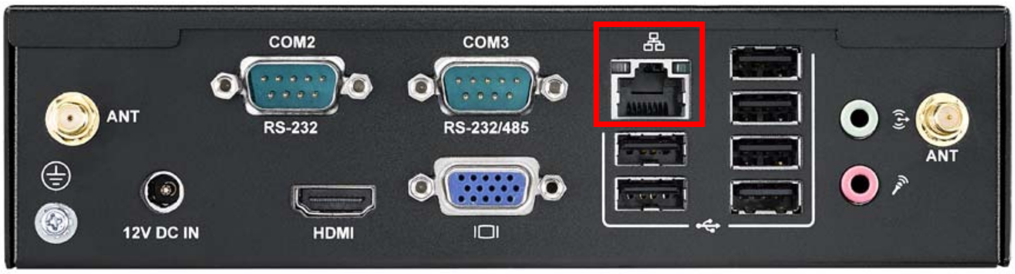


Pin	Description	Pin	Description
1	VGA_b_R	9	+5V_CRT
2	VGA_b_G	10	GND
3	VGA_b_B	11	NC
4	NC	12	VGA_b_DDC_DAT
5	GND	13	VGA_b_HSYNC
6	GND	14	VGA_b_VSYNC
7	GND	15	VGA_b_DDC_CLK
8	GND		

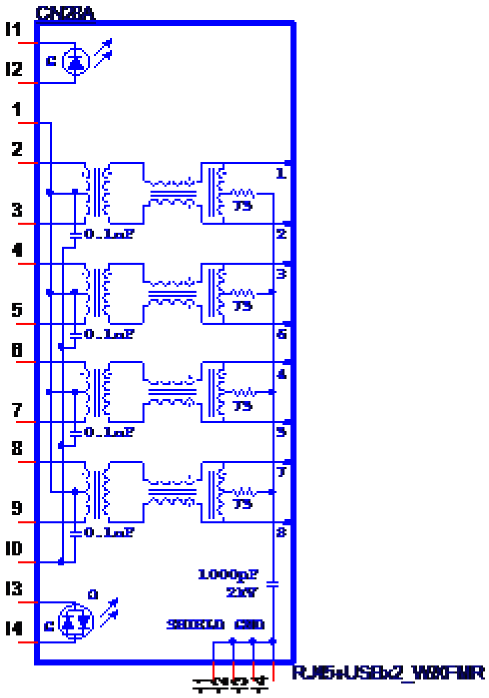


2.3.12 LAN

EPC-R6410 supports Ethernet; pin definitions below



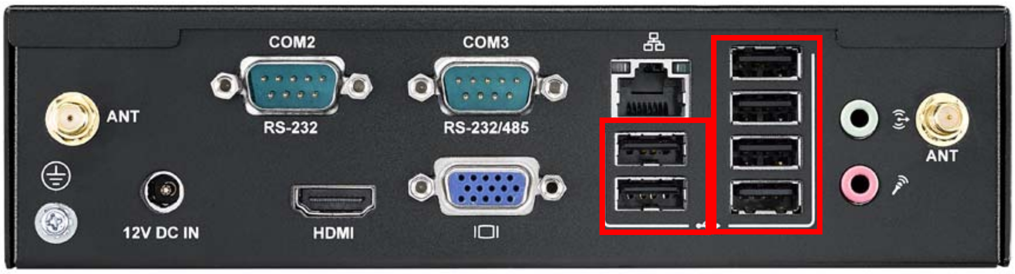
Pin	Description
1	NC
2	GBE_MDI0_P
3	GBE_MDI0_N
4	GBE_MDI1_P
5	GBE_MDI1_N
6	GBE_MDI2_P
7	GBE_MDI2_N
8	GBE_MDI3_P
9	GBE_MDI3_N
10	GND
11	LAN1_ACT
12	+3.3V
13	LAN1_1000_LINK
14	LAN1_100_LINK



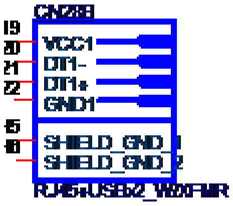
Ethernet Connector

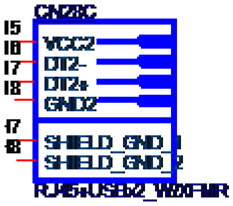
2.3.13 USB

EPC-R6410 supports 6 x USB2.0 type A connectors



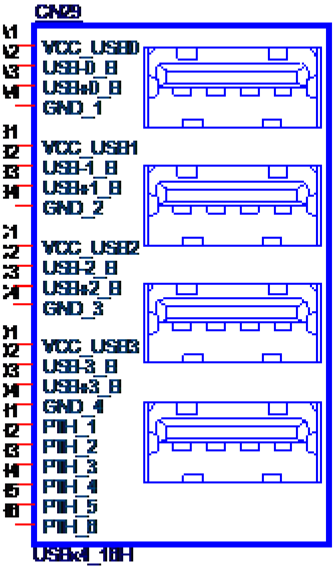
Pin	Description
15	+5V_USB_P56_B
16	USB_P6L_N
17	USB_P6L_P
18	USBV56_GND
19	+5V_USB_P56_B
20	USB_P5L_N
21	USB_P5L_P
22	USBV56_GND





Pin	Description
A1	+5V_USB_P12_B
A2	USB_P1L_N
A3	USB_P2L_P
A4	USB_12_GND
B1	+5V_USB_P12_B
B2	USB_P2L_N
B3	USB_P2L_P
B4	USB_12_GND
C1	+5V_USB_P34_B
C2	USB_P3L_N
C3	USB_P3L_P

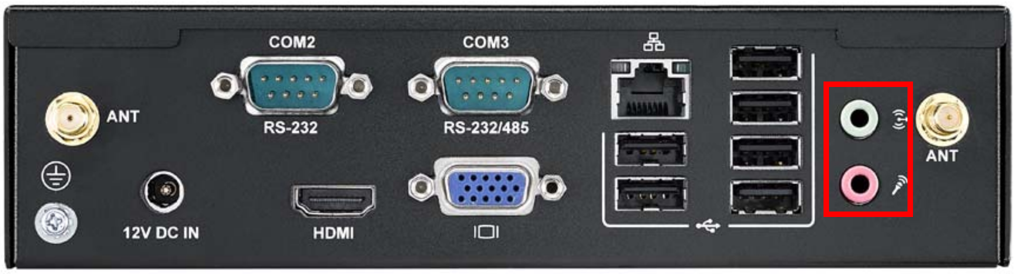
C4	USB_34_GND
D1	+5V_USB_P34_B
D2	USB_P4L_N
D3	USB_P4L_P
D4	USB_34_GND



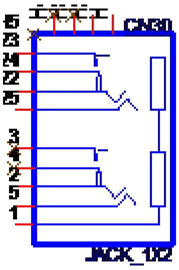
USB port connector

2.3.14 Audio Jack

ECP-R6410 supports 1 Line out and 1 Mic in



Pin	Description
1	GND
2	MIC
3	NC
4	NC
5	NV
22	HPOUT_L
23	lLINEOUT_DETECT
24	GND
25	HPOUT_R



2.4 Quick Start Guide

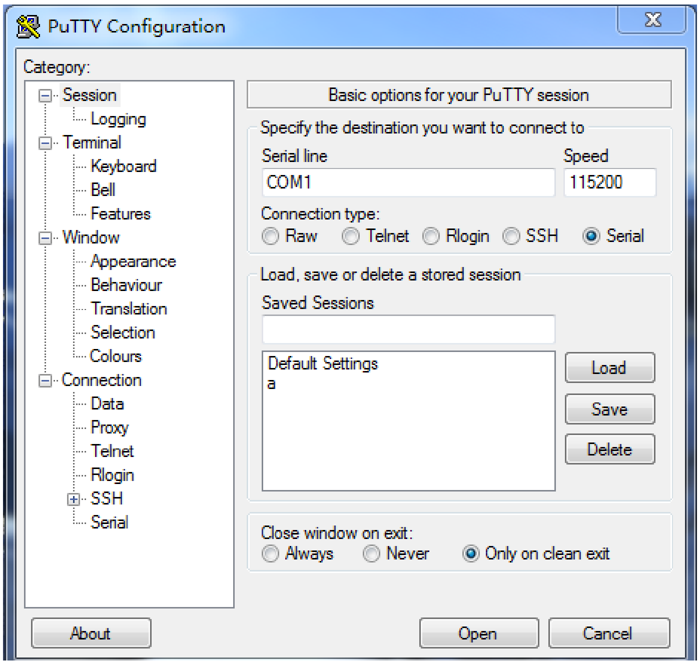
2.4.1 Debug Port Connection

1. Connect debug cable to EPC-R6410 debug port. (refer figure 2.3.1)
2. Connect the other side of debug cable to USB-to-RS232 cable then connect to your PC.

2.4.2 Debug Port Setting

EPC-R6410 can communicate with a host server by using serial cables. Common serial communication programs such as HyperTerminal, Tera Term or PuTTY can be used in this case. The example below describes the serial terminal setup using HyperTerminal on a Windows host:

1. Connect EPC-R6410 with your PC by using a serial cable.
2. Open HyperTerminal on your Windows PC, and select the settings as shown in Figure 2-7.
3. After the bootloader is programmed on SD card, insert power adapter connector to DC jack on EPC-R6410 to power up the board. The bootloader prompt is displayed on the terminal screen.



Specify the destination you want to connect to

Serial line: COM1

Speed: 115200

Connection type: ○ Raw ○ Telnet ○ Rlogin ○ SSH ● Serial

Load, save or delete a stored session

Saved Sessions:

Default Settings
a

Close window on exit: ○ Always ○ Never ● Only on clean exit

Figure 2.1 HyperTerminal Settings for Terminal Setup

Chapter 3

Software Functionality

This chapter details the software programs on the EPC-R6410 platform.

3.1 Test Tools

All test tools must be verified on EPC-R6410 Evaluation kit, please prepare the required test fixtures before verifying each specified I/O. If you have any problem getting the test fixture, please contact your Advantech contact window for help.

3.2 eMMC Test

1. When booting from SD, erase and check

```
# busybox dd if=/dev/zero of=/dev/block/mmcblk0 bs=1024 count=1 seek=1
1+0 records in
1+0 records out

# busybox hexdump -C /dev/block/mmcblk0 -s 1024 -n 16
00000400  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

2. Write and check

```
# echo -n "0123456789ABCDEF" | dd of=/dev/block/mmcblk0 bs=1024 count=1
seek=1
0+1 records in
0+1 records out

# busybox hexdump -C /dev/block/mmcblk0 -s 1024 -n 16
00000400  30 31 32 33 34 35 36 37  38 39 41 42 43 44 45 46  |0123456789ABCDEF|
```

Note! *Insert the SD Card, the eMMC NAND Flash is mmcblk0*

3.3 USB Test

1. Insert USB flash disk then assure that it is in EPC-R6410 device list.
2. Erase and check

```
# dd if=/dev/zero of=/dev/block/sda bs=1024 count=1 seek=25118
1+0 records in
1+0 records out

# busybox hexdump -C /dev/block/sda -s 25720832 -n 16
01887800  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

Note! *This operation may damage the data stored in the USB flash disk. Please make sure there is no critical data in the USB flash disk being used for this test.*

3. Write and check

```
# echo -n "0123456789ABCDEF" | dd of=/dev/block/sda  bs=1024 count=1
seek=25118
0+1 records in
0+1 records out

# busybox hexdump -C /dev/block/sda -s 25720832 -n 16
01887800  30 31 32 33 34 35 36 37  38 39 41 42 43 44 45 46  |0123456789ABCDEF|
```

Note!

1. *NXP i.MX6D/Q has a limitation on USB device collection, we recommend the follow brands: Logitech K120 / Lenovo K5819 LXH-EKB-10YA / RAPOO 1800.Pro / Dell MS111-P / Microsoft Wired Keyboard 200 (Model:1406) and so on.*
2. *This operation may damage the data stored in USB flash disk. Please make sure there is no critical data in the USB flash disk being used for this test. If your U Disk size is small, the seek value needs to be small.*

3.4 SD Test

1. When booting from eMMC, you will only see the below directories:

```
# ls /dev/block/mmcblk*

/dev/block/mmcblk0  /dev/block/mmcblk0boot0  /dev/block/mmcblk0boot1 /dev/block/
mmcblk0p1  /dev/block/mmcblk0p2  /dev/block/mmcblk0p3 /dev/block/mmcblk0p4  /
dev/block/mmcblk0p5  /dev/block/mmcblk0p6 /dev/block/mmcblk0p7  /dev/block/
mmcblk0p8
```

2. Insert SD card to SD card slot (SD1) and check your device again. You should be able to see more directories. /dev/block/mmcblk1 is the SD card storage (Note: SD Card can have 2 partitions).

```
# ls /dev/block/mmcblk*
/dev/block/mmcblk0  /dev/block/mmcblk0boot0  /dev/block/mmcblk0boot1 /dev/block/
mmcblk0p1  /dev/block/mmcblk0p2  /dev/block/mmcblk0p3 /dev/block/mmcblk0p4  /
dev/block/mmcblk0p5  /dev/block/mmcblk0p6 /dev/block/mmcblk0p7  /dev/block/
mmcblk0p8  /dev/block/mmcblk1  /dev/block/mmcblk1p1  /dev/block/mmcblk1p2  /dev/
block/mmcblk1p3 /dev/block/mmcblk1p4  /dev/block/mmcblk1p5  /dev/block/
mmcblk1p6 /dev/block/mmcblk1p7  /dev/block/mmcblk1p8
```

3. Erase and check

```
# dd if=/dev/zero of=/dev/block/mmcblk1 bs=1024 count=1 seek=25118
1+0 records in
1+0 records out

# busybox hexdump -C /dev/block/mmcblk1 -s 25720832 -n 16
01887800  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

4. Write and check

```
# echo -n "0123456789ABCDEF" | dd of=/dev/block/mmcblk1 bs=1024 count=1
seek=25118
0+1 records in
0+1 records out

# busybox hexdump -C /dev/block/mmcblk1 -s 25720832 -n 16
01887800  30 31 32 33 34 35 36 37  38 39 41 42 43 44 45 46  |0123456789ABCDEF|
```

Note! *Please make sure parameter “seek” is equal to 25118 as indicated in the above codes. If you create the file to a wrong sector, it may damage the system.*

3.5 GPIO Test

EPC-R6410 GPIO default setting

EPC-R6410	Android OS /sys/class/gpio/	Default direction
GPIO0	gpio2	in
GPIO1	gpio3	in
GPIO2	gpio4	in
GPIO3	gpio 5	in
GPIO4	gpio6	in
GPIO5	gpio7	in
GPIO6	gpio8	in
GPIO7	gpio9	in
GPIO8	gpio10	in
GPIO9	gpio11	in
GPIO10	gpio12	out
GPIO11	gpio13	out
GPIO12	gpio14	out
GPIO13	gpio15	out
GPIO14	gpio16	out
GPIO15	gpio17	out
GPIO16	gpio18	out
GPIO17	gpio19	out

```
# cd /sys/class/gpio
```

You can use "ls" to list all GPIO devices, and you should also see GPIO ports in above table.

Example of testing GPIO

Set gpio2 GPI(in)

```
# echo in > ./gpio2/direction
# cat ./gpio2/direction
in
```

Set gpio12 GPO (out)

```
# echo out > ./gpio12/direction
# cat ./gpio12/direction
out
```

Set gpio12 GPO value “1”

```
# echo 1 > ./gpio12/value
```

Get gpio2 GPI value

```
# cat ./gpio2/value
1
```

As you can see in above procedure A and B, we set gpio2 as GPI and gpio12 as GPO, once we send data out from gpio12, it should be able to receive the same data from gpio2.

3.6 I2C Test

There is one I2C bus in EPC-R6410.

```
# ls /sys/class/i2c-dev
i2c-0
i2c-1
i2c-2

# i2cdetect -l
i2c-0   i2c          imx-i2c                        I2C adapter
i2c-1   i2c          imx-i2c                        I2C adapter
i2c-2   i2c          imx-i2c                        I2C adapter
```

Please try the command below to check if there is any device connected to i2c bus 1.

```
# i2cdetect -yr 2
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- 29 -- -- -- -- -- --
30: -- -- -- -- -- -- -- 37 -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- 49 -- -- -- -- -- --
50: UU 51 52 53 54 55 56 57 -- 59 -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- UU --
```

The 0x29 is the 8051 address. Try the command below to check if the I2C bus is working or not.

```
#  i2cdump -yf 2 0x29
No size specified (using byte-data access)
No size specified (using byte-data access)
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
10: ff ff ff ff ff 2c c8 ff ff ff ff ff ff ff ff ff    .....,?.........
20: ff ff ff ff ff 28 c4 00 00 ff ff ff ff ff ff ff    .....(?.........
30: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
40: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
50: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
60: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
70: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
80: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
90: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
a0: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
b0: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
c0: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
d0: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
e0: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
f0: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
```

3.7 LAN Test

EPC-R6410 sets DHCP as the default network protocol.

```
# busybox ifconfig -a
can0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          UP RUNNING NOARP  MTU:16  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:10
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
          Interrupt:142

eth0      Link encap:Ethernet  HWaddr 00:04:9F:01:30:E0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  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:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          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)
```

If you would like to config IP manually, please use the command below:

```
# busybox ifconfig eth0 xxx.xxx.xxx.xxx up
```

Here is a real case for your reference. The host (EPC-R6410) IP is 192.168.0.10; the target (a desktop computer) IP is 192.168.0.12

```
#busybox ifconfig eth0 down
# busybox ifconfig eth0 192.168.0.10 up
# busybox ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:04:9F:01:30:E0
          inet addr:192.168.0.10  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  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:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

The target computer (Client) IP address is 192.168.0.12, so we can use the command below to see if we can get any response from the client.

```
# ping 192.168.0.12

PING 192.168.0.12 (192.168.0.12): 56 data bytes
64 bytes from 192.168.0.12: seq=0 ttl=128 time=7.417 ms
64 bytes from 192.168.0.12: seq=1 ttl=128 time=0.203 ms
64 bytes from 192.168.0.12: seq=2 ttl=128 time=0.300 ms

--- 172.17.20.192 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
```

Note! *The target computer (Client) firewall need close.*

3.8 UART Test

As you can see below, there are 5 UARTs supported by EPC-R6410. /dev/ttymxc0 is reserved for EPC-R6410 debug port, the rest of the UART ports can be applied by the user. UART1/UART2/UART3/UART4 connect to loopback.

HW	SW	DEVICE
COM1	UART0	/dev/ttymxc0
COM2	UART1	/dev/ttymxc1
COM3	UART2	/dev/ttymxc2
COM4	UART3	/dev/ttymxc3
COM5	UART4	/dev/ttymxc4

Test RSB232

```
# AutoRun_uart232

=====test rs232!=====
rs232 number: 5
/dev/ttymxc1 PASS!
/dev/ttymxc2 PASS!
/dev/ttymxc3 PASS!
/dev/ttymxc4 PASS !

+--------------------+
| [RS232] Test Pass!|
+--------------------+
```

Note! *Only UART2 supports RSB232/RSB422/RSB485, test rsb232 mode, please jump rsb232 mode, you can refer to detailed jumper information from the 2.1.2 Jumper List.*

UART2 Test RSB422

```
# AutoRun_uart422 -p /dev/ttymxc2 -t 1 -s 115200
==========test RS422============
 Open uart /dev/ttymxc2 PASS ....

 ->Writing : hello world!
 ->Reading : hello world!
 ->TX/RX Signal pass
+-------------------------+
| UART RS422 Testing PASS |
+-------------------------+
```

Note! *UART2 Jump RSB422 mode and Connect to loopback, you can refer to detailed jumper information in the 2.1.2 Jumper List.*

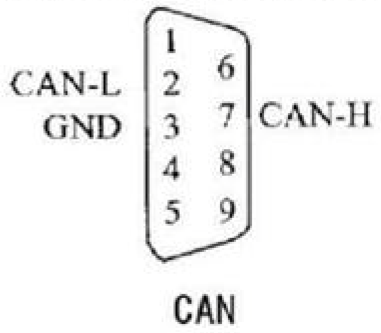
UART2 Test RS485

EPC-R6410 only supports one RS485 port; please find another device that supports RS485 port and connect to COM3 using the Serial Port Tool from 4.8 Serial Port Test.

Note! *For COM3 Jump RS485 mode, you can refer to detailed jumper information in the 2.1.2 Jumper List.*

3.9 Can Test

1. Prepare one Advantech PCI-1680U Can card and install the driver.
2. Connect EPC-R6410 CAN port CAN1_D+ /CAN1_D- and GND with Advantech PCI-1680U Can card.



3. Use the following command under EPC-R6410 to send data through to the Can card.

```
# cansend can0 0x01 0x02 0x03
interface = can0, family = 29, type = 3, proto = 1
```

4. Use the following command under EPC-R6410 to receive data form the opposite side:

```
# candump
interface = can0, family = 29, type = 3, proto = 1
```

3.10 Display Output Setting

Please set the U-boot environment as shown below

3.10.1 Single Display Settings

1. HDMI out, please set U-boot as shown below:
   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=extension
   video=mxcfb0:dev=hdmi,1920x1080M@60,bpp=32 video=mxcfb1:off
   video=mxcfb2:off video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```
2. VGA out, please set U-boot as shown below:
   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=extension
   video=mxcfb0:dev=lcd,1920x1080M@60,bpp=32 video=mxcfb1:off
   video=mxcfb2:off video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```
3. LVDS (Single Channel 7inch) out, please set U-boot as shown below:
   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=displayLSC
   video=mxcfb0:dev=ldb,LDB-800X480,if=RGB666 bpp=24 ldb=sin0
   video=mxcfb1:off video=mxcfb2:off video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```
4. LVDS (Single Channel 15inch) out, please set U-boot as shown below:
   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=displayLSC
   video=mxcfb0:dev=ldb,LDB-1024X768,if=RGB24 bpp=32 ldb=sin0
   video=mxcfb1:off video=mxcfb2:off video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```
5. LVDS (Dual Channel 21inch) out, please set U-boot as shown below:
   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=displayLDC
   video=mxcfb0:dev=ldb,LDB-1080P60,if=RGB24 bpp=32 ldb=spl0
   video=mxcfb1:off video=mxcfb2:off video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

3.10.2 Dual Display Settings

When you want to display dual LVDS, VGA and HDMI output, please set U-boot parameters follows. This is the default setting in U-boot.

1. HDMI and VGA clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=extension
   video=mxcfb0:dev=lcd,1920x1080M@60,bpp=32
   video=mxcfb1:dev=hdmi,1920x1080M@60,bpp=32 video=mxcfb2:off
   video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

   Note! *This is the default setting.*

2. LVDS (Single Channel 15-inch * 2pieces) clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=displayLSCC
   video=mxcfb0:dev=ldb,LDB-1024X768,if=RGB24 bpp=32 ldb=sep0
   video=mxcfb1:dev=ldb,LDB-1024X768,if=RGB24 bpp=32 ldb=sep1
   video=mxcfb2:off video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

3. LVDS (Single Channel 15-inch) and VGA clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=displayLSCV
   video=mxcfb0:dev=ldb,LDB-1024X768,if=RGB24 bpp=32 ldb=sin0
   video=mxcfb1:dev=lcd,1920x1080M@60,bpp=32 video=mxcfb2:off
   video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

4. LVDS (Single Channel 15inch) and HDMI clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=extension
   video=mxcfb0:dev=ldb,LDB-1024X768,if=RGB24,bpp=32 ldb=sin0
   video=mxcfb1:dev=hdmi,1920x1080M@60,bpp=32 video=mxcfb2:off
   video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

5. LVDS (Dual Channel 21inch) and VGA clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=extension
   video=mxcfb0:dev=ldb,LDB-1080P60,if=RGB24,bpp=32 ldb=spl0
   video=mxcfb1:dev=lcd,1920x1080M@60,bpp=32 video=mxcfb2:off
   video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

6. LVDS (Dual Channel 21inch) and HDMI clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=displayLDCH
   video=mxcfb0:dev=ldb,LDB-1080P60,if=RGB24,bpp=32 ldb=spl0
   video=mxcfb1:dev=hdmi,1920x1080M@60,bpp=32 video=mxcfb2:off
    video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

7. LVDS (Single Channel 7inch) and HDMI clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init video_mode=displayL7
   video=mxcfb0:dev=ldb,LDB-800X480,if=RGB666,bpp=24
   video=mxcfb1:dev=hdmi,1920x1080M@60,bpp=24 video=mxcfb2:off
   video=mxcfb3:off fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

3.10.3 Triple Display Settings

1. LVDS (Single Channel 7inch), HDMI and VGA clone out, please set U-boot as shown below:

   ```
   setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0
   vmalloc=400M init=/init
   video=mxcfb0:dev=ldb,LDB-800X480,if=RGB666,bpp=24
   video=mxcfb1:dev=hdmi,1920x1080M@60,bpp=24 fbmem=28M,28M
   androidboot.hardware=freescale pcie_testmode=off
   ```

Chapter 4

Android OS

This chapter introduces how to build an Android system and develop it, based on our Android system.

4.1 Introduction

The purpose of this chapter is to introduce EPC-R6410 software development so that you can develop your own applications. EPC-R6410 supports Linux host only, so trying to develop an EPC-R6410 app on Windows/Android host PC will fail. For now the official supported host version is Ubuntu 14.04 64-bit; host PCs in any other version may have compatibility issues. We strongly recommend having Ubuntu 14.04 LTS 64-bit installed to your host PC before starting EPC-R6410 evaluation/development.

4.2 Board Support Package

We offer you two different kinds of Android packages for EPC-R6410. One is a prebuilt system image; the other is a source code package (BSP).

4.2.1 Pre-built System Image

You can find the pre-built image EPC-R6410AIVxxxx_yyyy-mm-dd.tar.gz from EPC-R6410 evaluation kit DVD image downloaded from the Advantech website. EPC-R6410 supports booting from SD card so you can extract the image to an SD card then dump the image file to the onboard eMMC to complete the system. Some folders and files are described below:

image:

This folder contains uboot image, kernel image and android root filesystem tar file and so on. These files are used to burn to sd card or nand flash.

scripts:

This folder contains scripts for easy image flashing.

mksd-android.sh

To create a bootable SD card. mkmmc-android.sh

Use the SD card to burn android firmware to eMMC flash

4.2.2 Source Code Package

EPC-R6410 board support package (BSP) contains cross compiler, linux kernel source code, U-boot source code, android root file system and some scripts used in OS development. Some of above components are developed by Advantech and the others are developed by the open source community.

Some folders described below:

android/bootable/bootloader/uboot-imx/

U-Boot source code

android/device/fsl/rsb_6410/

Android related settings for EPC-R6410

android/hardware/imx/

HAL (Hardware Abstraction Layer)

android/kernel_imx/

Android kernel source code

android/prebuilts/

Android toolchain, sdk and so on

4.3 Set Up Build Environment

All instructions in this guide are based on Ubuntu 14.04 64-bit only. Please install Ubuntu 14.04 64bit with minimum 2GB DRAM in advance, login to the installed system and perform the following:

4.3.1 Installing JDK

root@PcName:~# sudo apt-get update

root@PcName:~# sudo apt-get install python-software-properties

1. Get jdk-6u45-linux-x64.bin from android-source-code-directory/tools/
2. Follow these steps to install JDK 6

```
$ chmoda+x jdk-6uXX-linux-x64.bin
$ ./jdk-6uXX-linux-x64.bin
$ sudomkdir -p /usr/lib/jvm
$ sudo mv jdk1.6.0_XX /usr/lib/jvm/
$ sudo update-alternatives --install "/usr/bin/java" "java" "/usr/lib/jvm/jdk1.6.0_XX/
bin/java" 1
$ sudo update-alternatives --install "/usr/bin/javac" "javac" "/usr/lib/jvm/
jdk1.6.0_XX/bin/javac" 1
$ sudo update-alternatives --config java
$sudo update-alternatives --config javac
```

4.3.2 Installing Required Packages

sudo apt-get install bison build-essential ccache curl dpkg flex gcc g++ gettext git-core gnupg gperf g++-multilib ia32-libs intltool lib1g-dev lib32ncurses5-dev lib32readline5-dev lib32z-dev libarchive-zip-perl libc6-dev libdbus-glib-1-dev libfreetype6-dev libgl1-mesa-dev libgtk2.0-dev liblzo2-2 liblzo2-dev libncurses5-dev liborbit2-dev libtool libx11-dev libxml2-utils libxml2-dev m4 mingw32 patch python-markdown rpm tcl uboot-mkimage uuid uuid-dev tofrodos x11proto-core-dev xsltproc zip zlib1g zlib1g-dev

4.3.3 Toolchain Setup

Before building an Android system, you need setup the toolchain.

Android toolchain is under android/prebuilds. Please refer to 4.2.1

Setup the toolchain path to point to arm-eabi- tools in android/prebuilds/gcc/linux-x86/arm/arm-eabi-4.6/bin,the command is:

root@PcName:~# export

PATH=$ANDROID_DIR/android/prebuilts/gcc/linux-x86/arm/arm-eabi-4.6/bin:$PATH

4.4 Build Instructions

1. Open one terminal console and change directory to BSP scripts folder.
2. Perform one of the following commands:

4.4.1 Build android

1. Open a terminal console and change directory to BSP scripts folder
2. Perform the following command:

   root@PcName:~# ./mk_android.sh <product name>

   where <product name> is:

   • **EPC-R6410** for EPC-R6410 board
3. Then you can get android all image file under image folder, include as follows **u-boot_crc.bin**, **u-boot_crc.bin.crc**, **boot.img**, **system.img**, **recovery.img**.

4.4.2 Build android kernel

1. Open a terminal console and change directory to BSP scripts folder
2. Perform the following command:
   root@PcName:~# ./mk_bootimg.sh <product name>
   where <product name> is:
   • **EPC-R6410** for EPC-R6410 board
3. Then you can get the android kernel image file named boot.img under **image** folder.

4.4.3 Build android system image

1. Open a terminal console and change directory to BSP scripts folder
2. Perform the following command:

   root@PcName:~# ./mk_system.sh <product name>

   where <product name> is:

   • **EPC-R6410** for EPC-R6410 board
3. Then you can get the android kernel image file named system.img under **image** folder

4.4.4 Build android recovery image

1. Open a terminal console and change directory to BSP scripts folder
2. Perform the following command:

   root@PcName:~# ./mk_recovery.sh <product name>

   where <product name> is:

   • **EPC-R6410** for EPC-R6410 board
3. Then you can get the android kernel image file named recovery.img under **image** folder

4.4.5 Build android OTA package

1. Open a terminal console and change directory to BSP scripts folder
2. Perform the following command:

   root@PcName:~# ./mk_otapackage.sh <product name>

   where <product name> is:

   • **EPC-R6410** for EPC-R6410 board
3. Then you can get the android kernel image file named update.zip under **image** folder

4.5 Debug Console

Connect the 9-pin D-SUB of debug console cable with the host computer serial port using a serial port terminal application (like minicom, putty or teraterm) then config serial port as 115200 8N1.

Then plug in the 4-pin connector debug console cable into EPC-R6410's debug port (also known as /dev/ttymxc0 in Android)

4.6 Boot up from SD card or eMMC

EPC-R6410 supports boot from SD card or onboard flash. This section will guide you how to build an image for EPC-R6410 Android system boot media.

4.6.1 The storages devices name as following:

Device	Name
SD card	/dev/block/mmcblk1
eMMC	/dev/block/mmcblk0

4.6.2 Create a bootable SD card

You are able to find the pre-built image from Advantech website. Please follow the steps below to create an SD card for boot up.

1. Copy “EPC-R6410AIVxxxx_yyyy-mm-dd.tar.gz” package to your /root/.
2. Open "Terminal" on Ubuntu 14.04 LTS..
3. **$**sudo su (Change to “root” authority)
4. Input your password.
5. **root@PcName:~#** cd /root/
6. **root@PcName:~#** tar xzvf EPC-R6410AIVxxxx_yyyy-mm-dd.tar.gz (Unzip files)
7. Insert one SD card into your development computer
8. Check the SD card location, like /dev/sdb
9. **root@PcName:~#** cd ./ EPC-R6410AIVxxxx_yyyy-mm-dd/scripts
10. **root@PcName:~#**./mksd-android.sh /dev/sdb

Then insert the SD card to EPC-R6410 and power up, it should boot up with an Android environment.

4.6.3 Transfer the whole system to onboard eMMC

1. Open a debug console
2. Boot up from the SD card

   Perform the following command:

   root@PcName:~# cd /data/mkimage /scripts

   root@PcName:~# ./mkmmc?android.sh /dev/block/mmcblk0
3. Remove the SD card, then EPC-R6410 can boot up from the onboard eMMC.

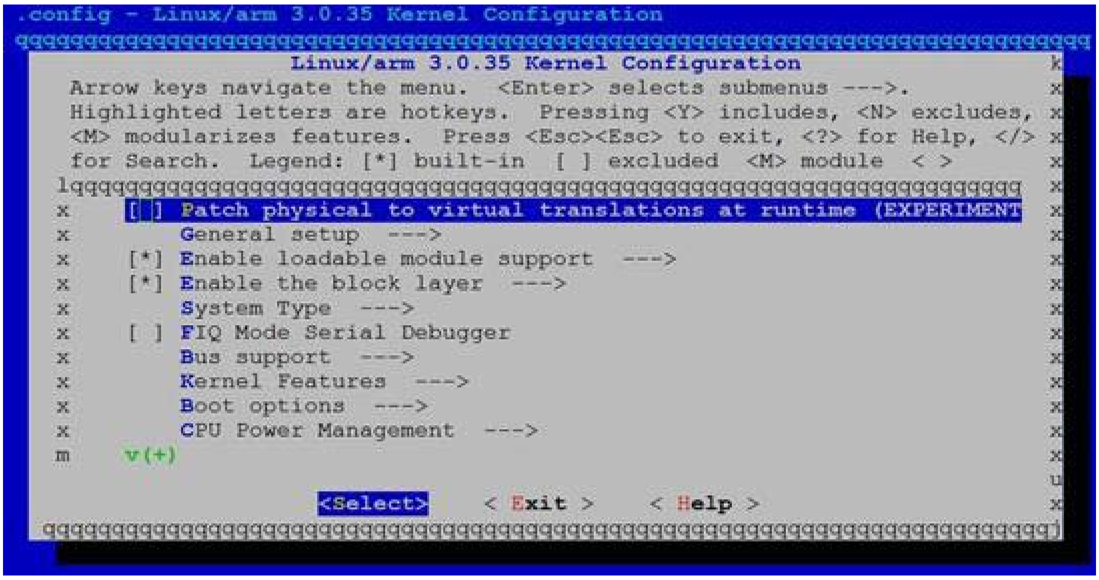
4.7 Customization

4.7.1 Config Android Kernel

1. Open a terminal console and change directory to BSP scripts folder
2. Perform the following command:

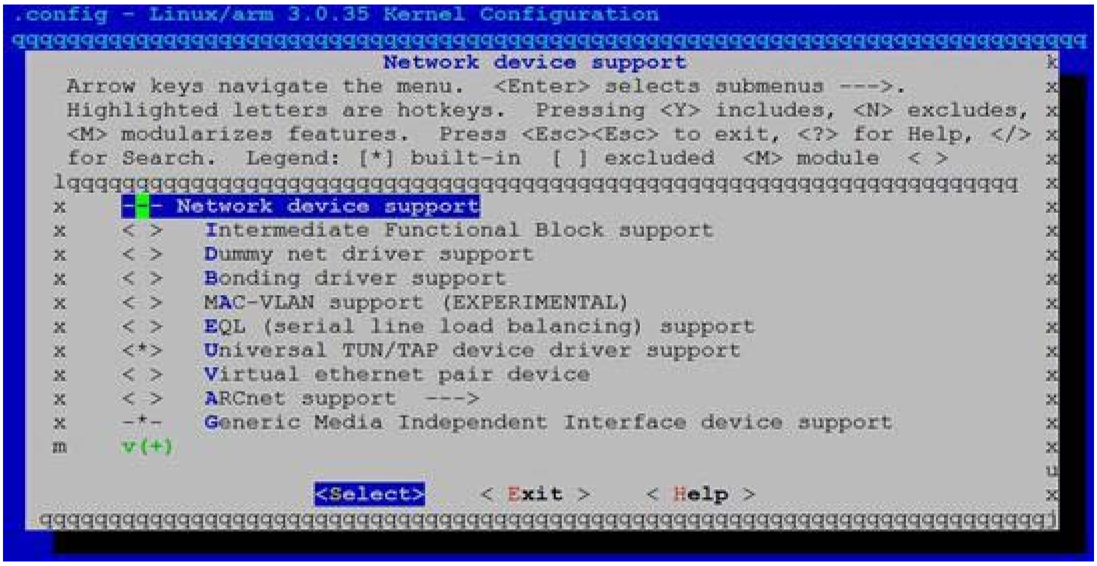
   root@PcName:~# cd android/kernel_imx

   root@PcName:~# make ARCH=arm menuconfig
3. Linux Kernel configuration is shown below:



```
.config - Linux/arm 3.0.35 Kernel Configuration
qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
  x                   Linux/arm 3.0.35 Kernel Configuration                    k
  x  Arrow keys navigate the menu.  <Enter> selects submenus --->.            x
  x  Highlighted letters are hotkeys.  Pressing <Y> includes, <N> excludes,   x
  x  <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </>   x
  x  for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < >         x
  x lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq    x
  x x    [ ] Patch physical to virtual translations at runtime (EXPERIMENT    x
  x x        General setup  --->                                              x
  x x    [*] Enable loadable module support  --->                             x
  x x    [*] Enable the block layer  --->                                     x
  x x        System Type  --->                                                x
  x x    [ ] FIQ Mode Serial Debugger                                         x
  x x        Bus support  --->                                                x
  x x        Kernel Features  --->                                            x
  x x        Boot options  --->                                               x
  x x        CPU Power Management  --->                                       x
  x m    v(+)                                                                 x
                                                                              u
  x                   <Select>    < Exit >    < Help >                        x
  mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj
```

4. In the Device Drivers / Network device support menu, you can select device(s) you want to build into your Linux kernel.



```
.config - Linux/arm 3.0.35 Kernel Configuration
qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
                        Network device support                              k
    Arrow keys navigate the menu.  <Enter> selects submenus --->.            x
    Highlighted letters are hotkeys.  Pressing <Y> includes, <N> excludes,   x
    <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </>   x
    for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < >         x
   lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq    x
   x    --- Network device support                                           x
   x    < >   Intermediate Functional Block support                          x
   x    < >   Dummy net driver support                                       x
   x    < >   Bonding driver support                                         x
   x    < >   MAC-VLAN support (EXPERIMENTAL)                                x
   x    < >   EQL (serial line load balancing) support                       x
   x    <*>   Universal TUN/TAP device driver support                        x
   x    < >   Virtual ethernet pair device                                   x
   x    < >   ARCnet support  --->                                           x
   x    -*-   Generic Media Independent Interface device support             x
   m    v(+)                                                                 x
                                                                             u
                       <Select>    < Exit >    < Help >                      x
  qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj
```

5. Refer to section 4.4.1 to rebuild everything.

4.7.2 Put source code into system

1. Put whole package directory to BSP’s folder:
   ./android/packages/apps/
2. Create a file, Android.mk, at package directory; the contents look like below:

```
LOCAL_PATH:= $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE_TAGS := optional
LOCAL_SRC_FILES := $(call all-java-files-under, src)
LOCAL_PACKAGE_NAME := PROJECT_NAME_HERE
LOCAL_CERTIFICATE := platform
include $(BUILD_PACKAGE)
# Use the following include to make our test apk. include $(call all-makefiles-under,$(LOCAL_PATH))
```

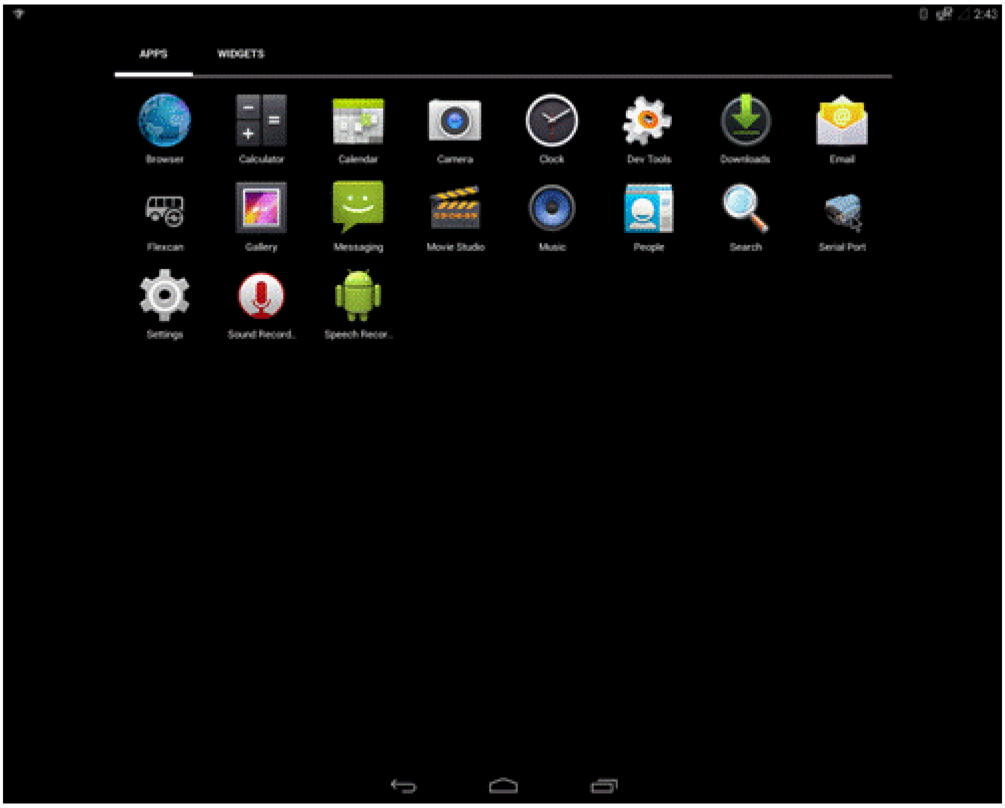
3. Edit following file:
   ./android/device/Advantech/EPC-R6410/device.mk
   to insert correct project name:

```
PRODUCT_PACKAGES += \
PROJECT_NAME_HERE\                    \
librxtxSerial\                        \
SerialJ
```

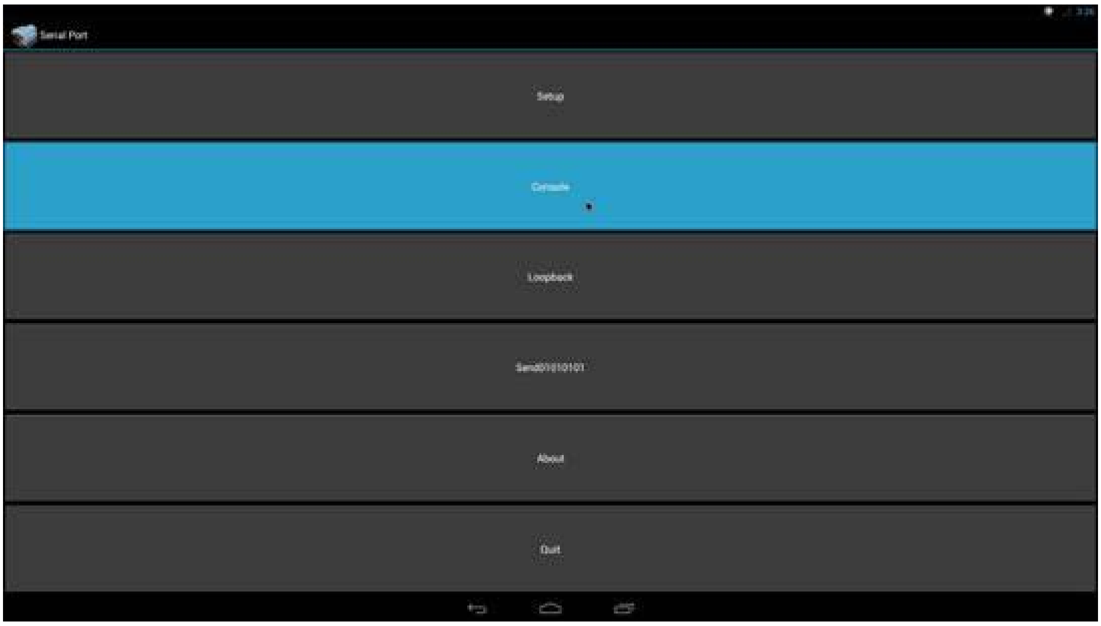
4.8 Serial Port Test

4.8.1 Setup Serial Port

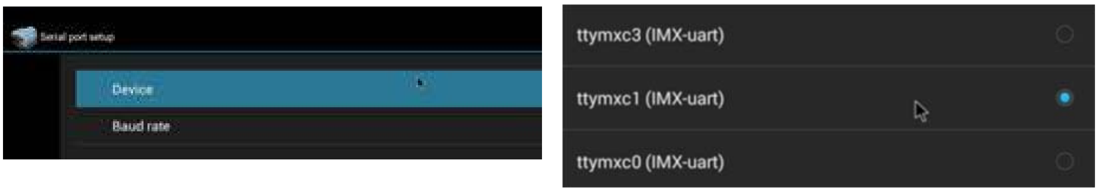
1. Click on the Serial Port.



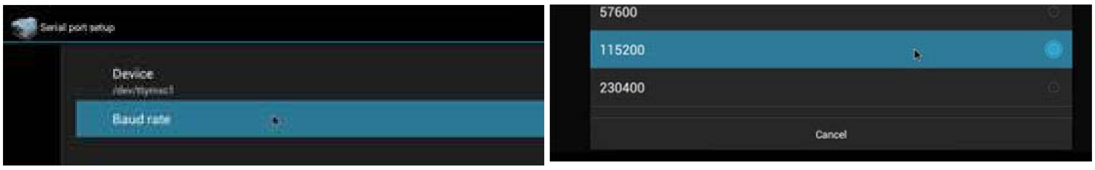
2. Click **Setup.**



3. Click Device, and choose used device (e.g., ttymxc1)

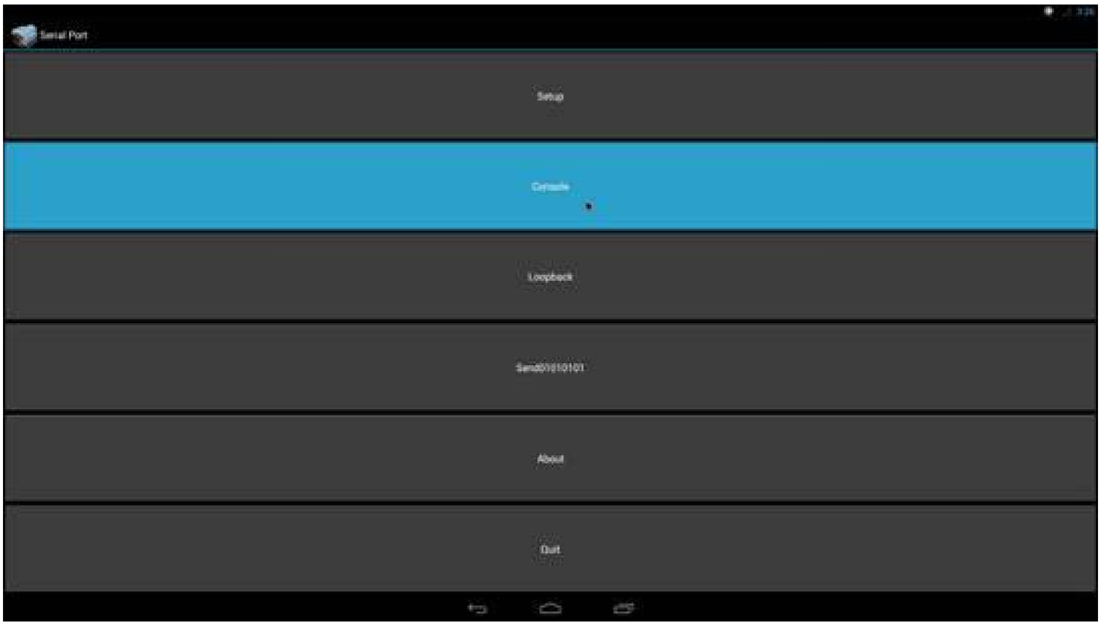


4. Click Baud rate, and choose used baudrate (e.g., 115200)

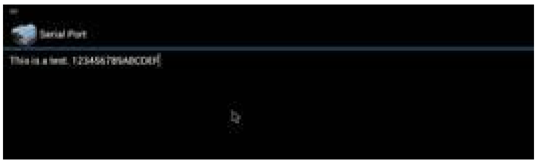


4.8.2 Console Test

1. Using serial cable to connec t ttymxc1 with PC/NB's serial port that should be configured to 115200 8N1, and opening serial terminal AP (like minicom, putty or teraterm)
2. Click Console.

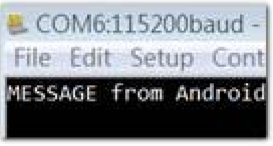


After typing some characters (e.g., "This is a test. 123456789ABCDEF") and pressing Enter in serial terminal AP, an identical message will be displayed in the reception block, as shown below:



```
This is a test. 123456789ABCDEF
```

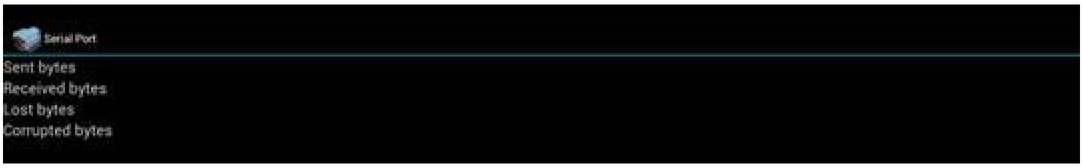
If you type "MESSAGE from Android" in emission block, an identical message will be displayed in the serial terminal AP, as shown below:



```
MESSAGE from Android
```

4.8.3 Loopback Test

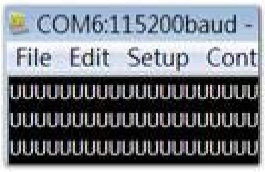
1. Plug loopback device into ttymxc1
2. Click Loopback.
   The loopback test result will be displayed as shown below



Sent bytes
Received bytes
Lost bytes
Corrupted bytes

4.8.4 Send01010101 Test

1. Use a serial cable to connect ttymxc1 with PC/NB's serial port that should be configured to 115200 8N1, and open a serial terminal AP (like minicom, putty or teraterm)
2. Click Send01010101.
   The character "U" (b'01010101) will be displayed continuously, as shown below:

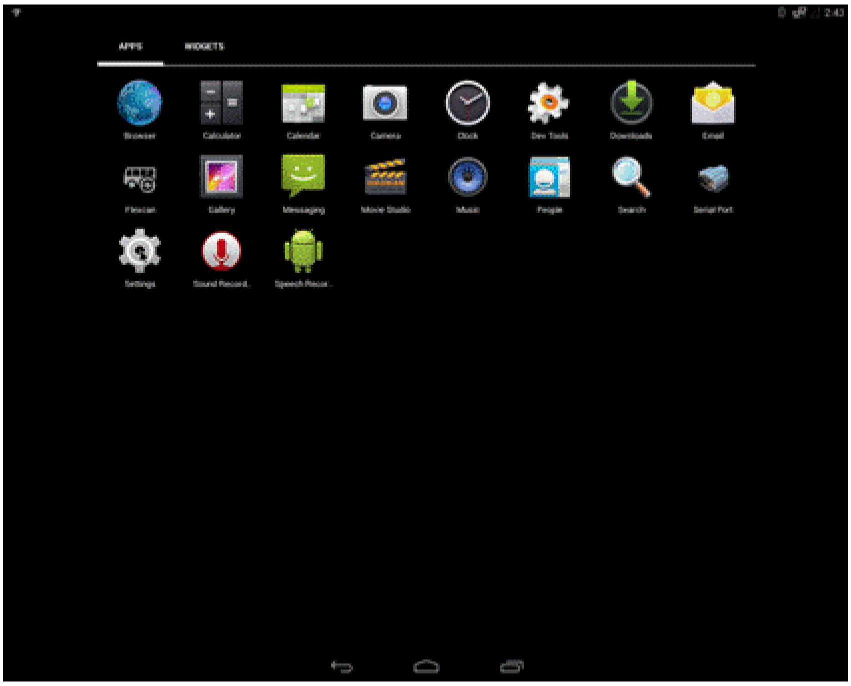


```
UUUUUUUUUUUUUUUUUUUU
UUUUUUUUUUUUUUUUUUUU
UUUUUUUUUUUUUUUUUUUU
```

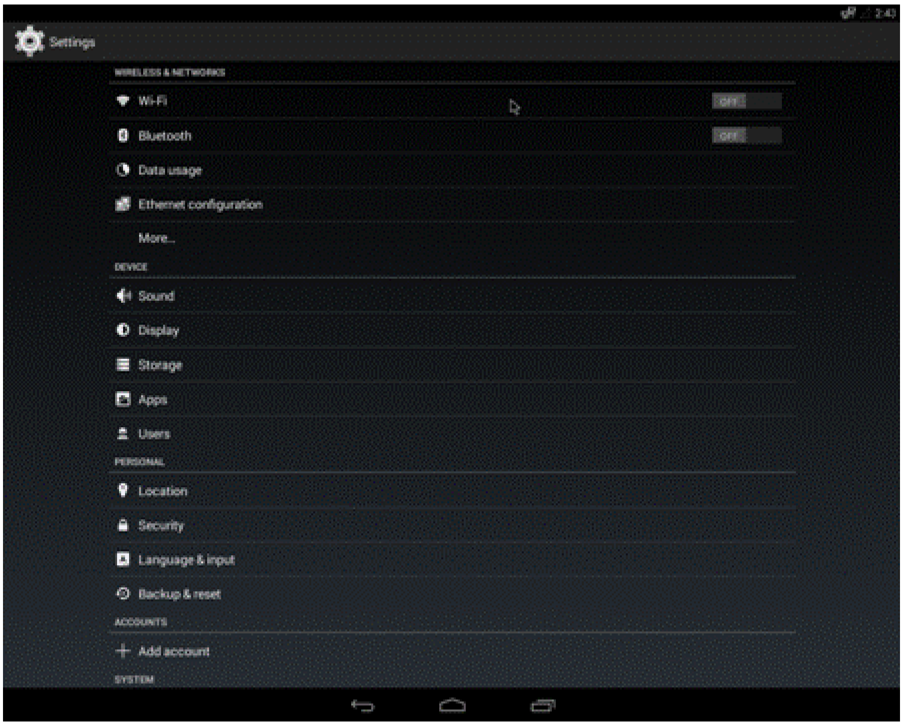
4.9 Network Setup

4.9.1 Wi-Fi

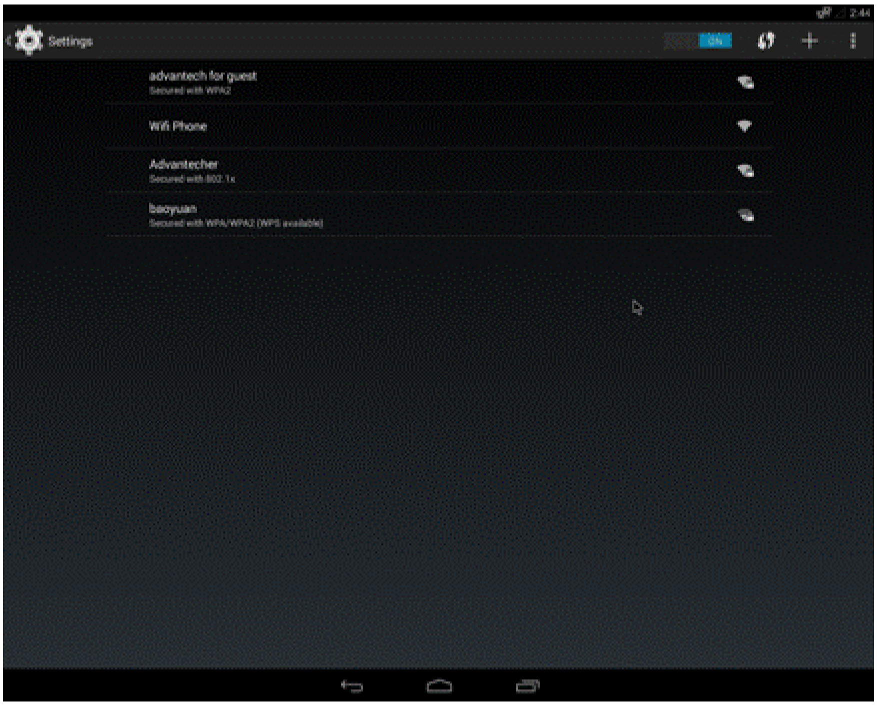
1. Click Settings.



2. Turn Wi-Fi on.



3. Choose ESSID (for example, Advantech for guest Testing)



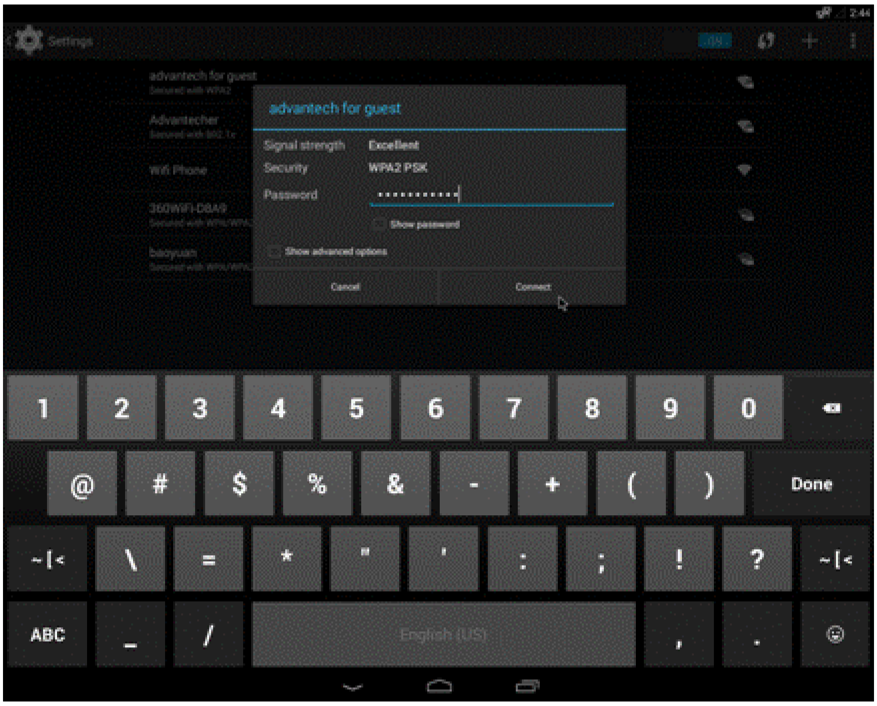
advantech for guest
Secured with WPA2

Wifi Phone

Advantecher
Secured with 802.1x

baoyuan
Secured with WPA/WPA2 (WPS available)

4. Input correct password.



advantech for guest

Signal strength Excellent

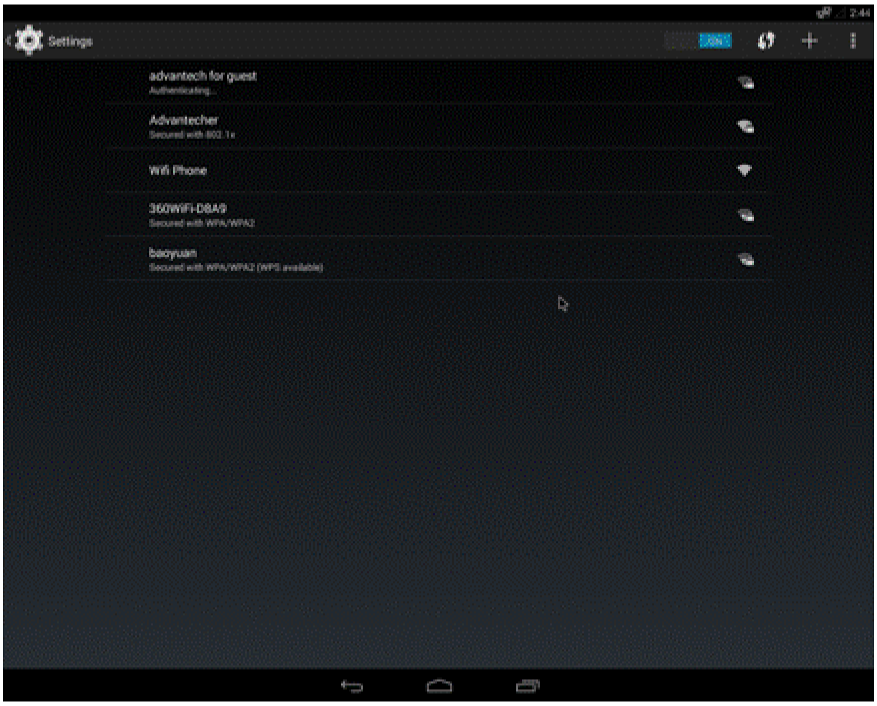
Security WPA2 PSK

Password ••••••••••••

□ Show password

□ Show advanced options

5. Wi-Fi Authenticating/Connecting/Obtaining IP address



2:44

Settings

ON

advantech for guest
Authenticating...

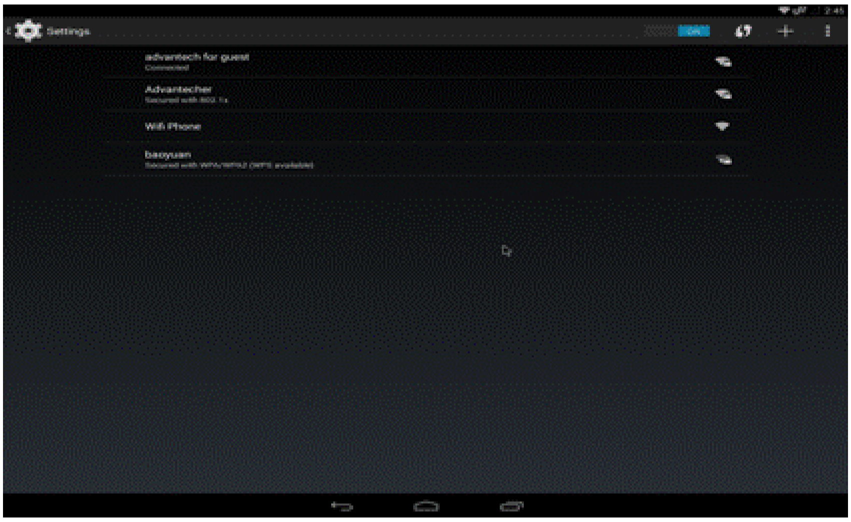
Advantecher
Secured with 802.1x

Wifi Phone

360WiFi-DBA9
Secured with WPA/WPA2

baoyuan
Secured with WPA/WPA2 (WPS available)

6. Wi-Fi connected



2:45

Settings

ON

advantech for guest
Connected

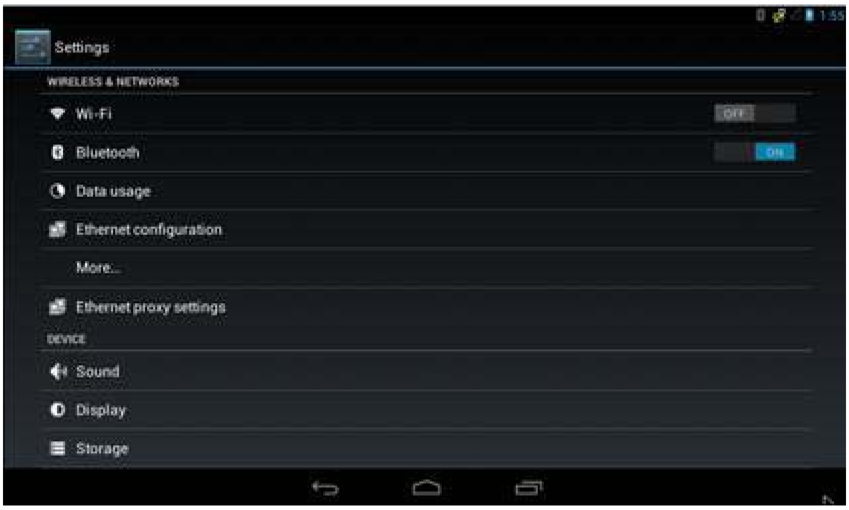
Advantecher
Secured with 802.1x

Wifi Phone

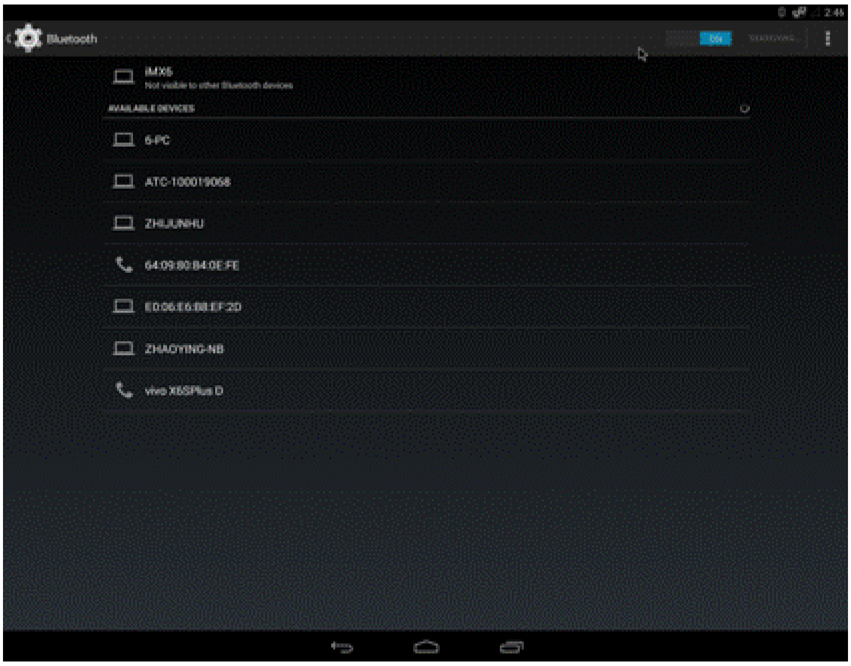
baoyuan
Secured with WPA/WPA2 (WPS available)

4.9.2 Bluetooth

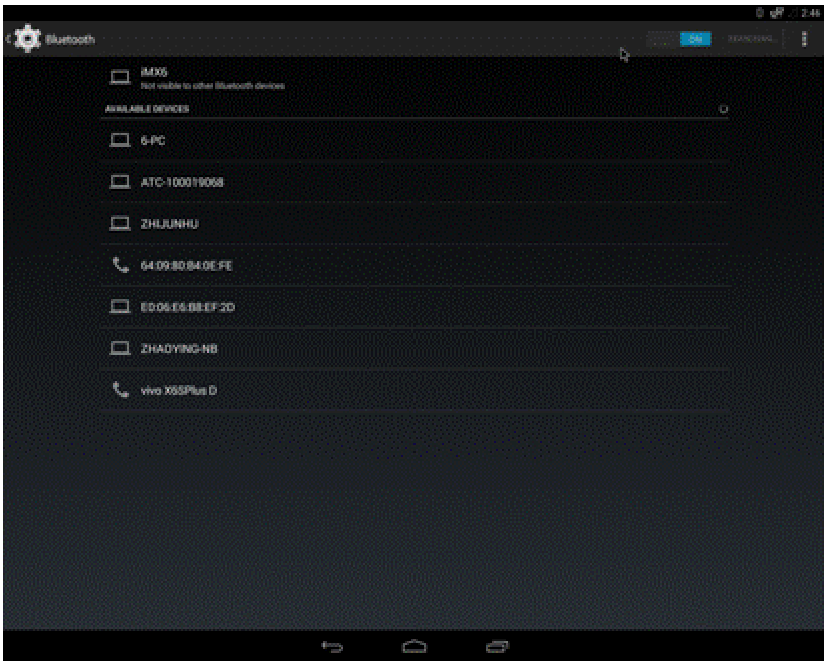
1. Click Settings, switch the Bluetooth switch to ON to Turn on Bluetooth:



2. Click Settings / Bluetooth for bluetooth main interface:



3. Click "iMX6" to let EPC-R6410 bluetooth be visible to other Bluetooth devices

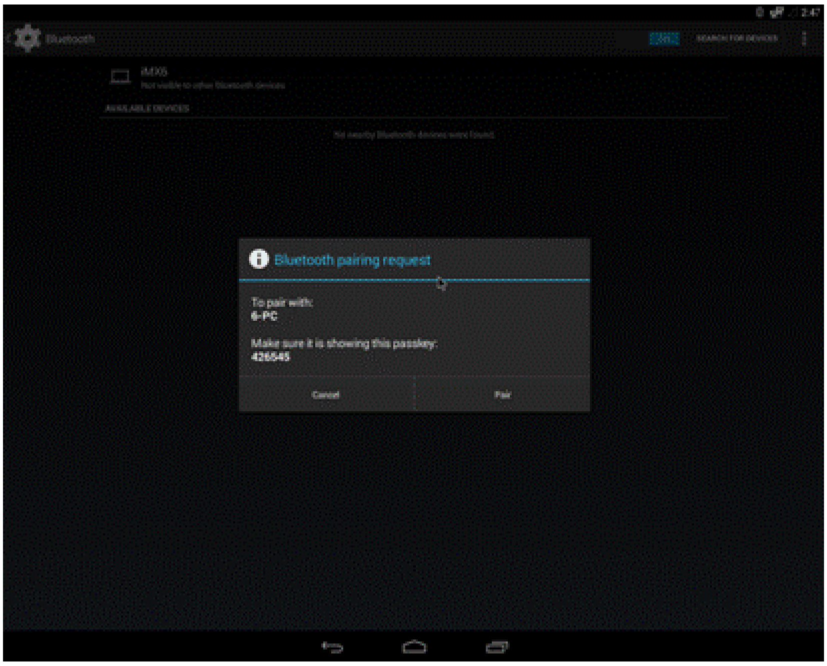


iMX6
Not visible to other Bluetooth devices

AVAILABLE DEVICES

- 6-PC
- ATC-100019068
- ZHIJUNHU
- 64:09:80:84:0E:FE
- E0:06:E6:B8:EF:2D
- ZHAOYING-NB
- vivo X6SPlus D

4. Click any available devices to pair with it:



iMX6
Not visible to other Bluetooth devices

AVAILABLE DEVICES

No nearby Bluetooth devices were found.

Bluetooth pairing request

To pair with:
6-PC

Make sure it is showing this passkey:
426545

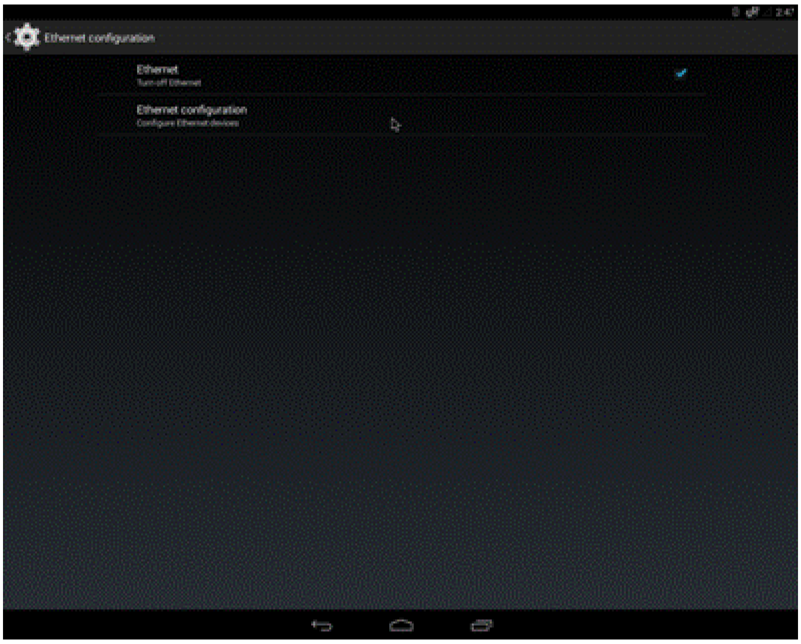
5. After pairing successfully with another Bluetooth device, you can communicate with it.

4.9.3 3G/4G

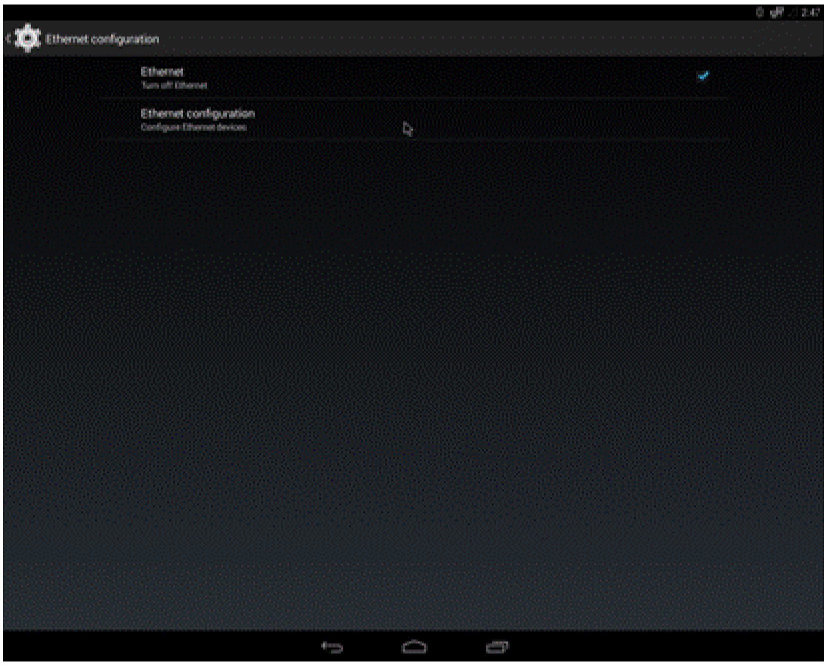
1. Insert Sim card and restart
2. If you can't connect to the network, please check the following settings:
   A. Settings/More/Mobile networks/Data enabled, then Enabled.
   B. Settings/More/Mobile networks/Access Point Names, then Correct.

4.9.4 Ethernet

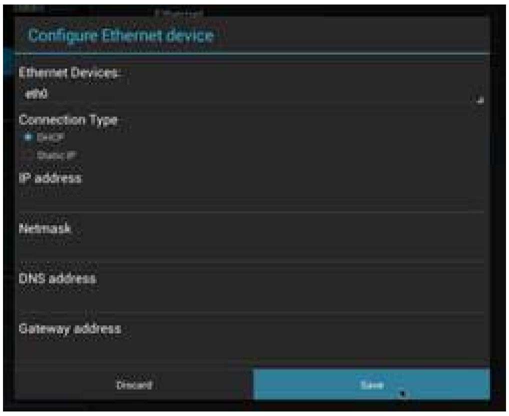
1. Click Settings / Ethernet configuration, then turn on Ethernet



2. Click Ethernet configuration.



3. Choose Connection Type (DHCP or Static IP)


Configure Ethernet device

Ethernet Devices:
eth0

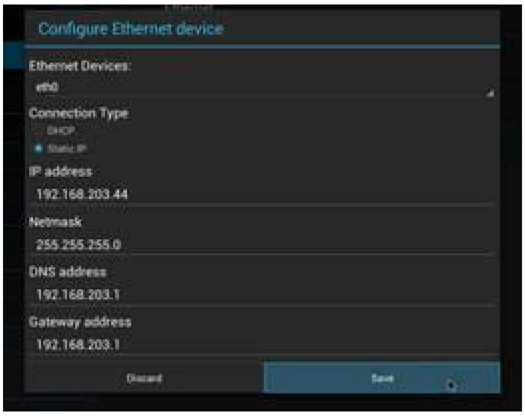
Connection Type
● DHCP
○ Static IP

IP address

Netmask

DNS address

Gateway address


Configure Ethernet device

Ethernet Devices:
eth0

Connection Type
○ DHCP
● Static IP

IP address
192.168.203.44

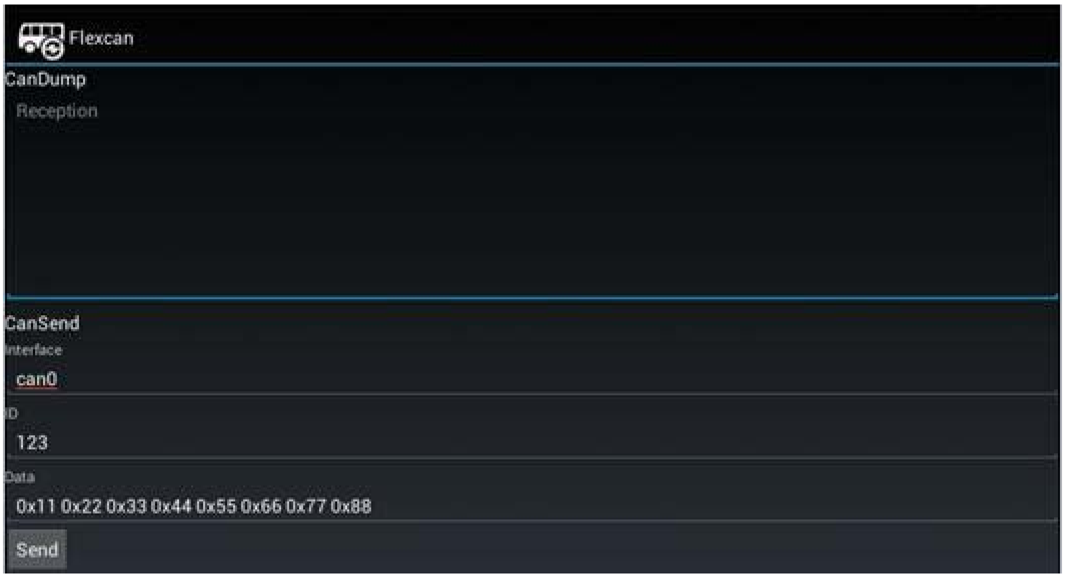
Netmask
255.255.255.0

DNS address
192.168.203.1

Gateway address
192.168.203.1

4.10 Can Test

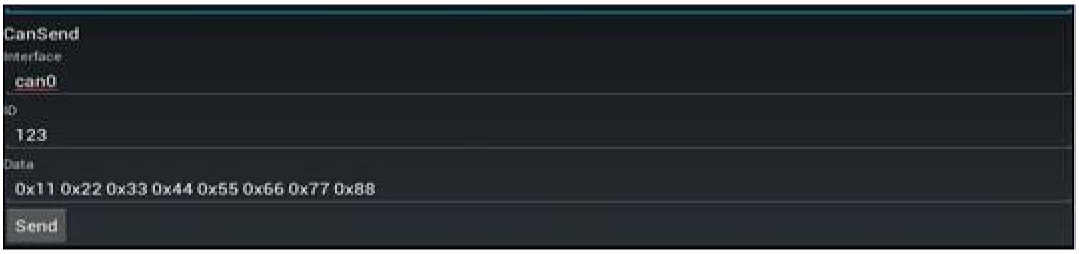
1. Click Flexcan.


Flexcan

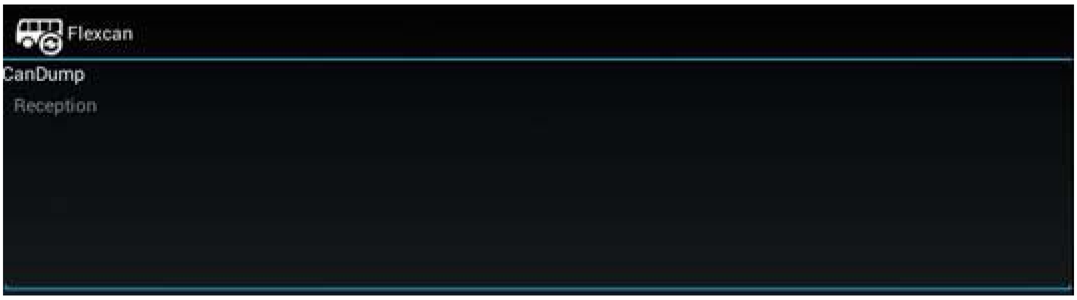
CanDump
Reception

CanSend
Interface
can0
ID
123
Data
0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88

2. Edit "interface","ID"and"Data", then click "Send" to send data through the Can port.


CanSend
Interface
can0
ID
123
Data
0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88

3. Received Data is displayed under the CanDump area.

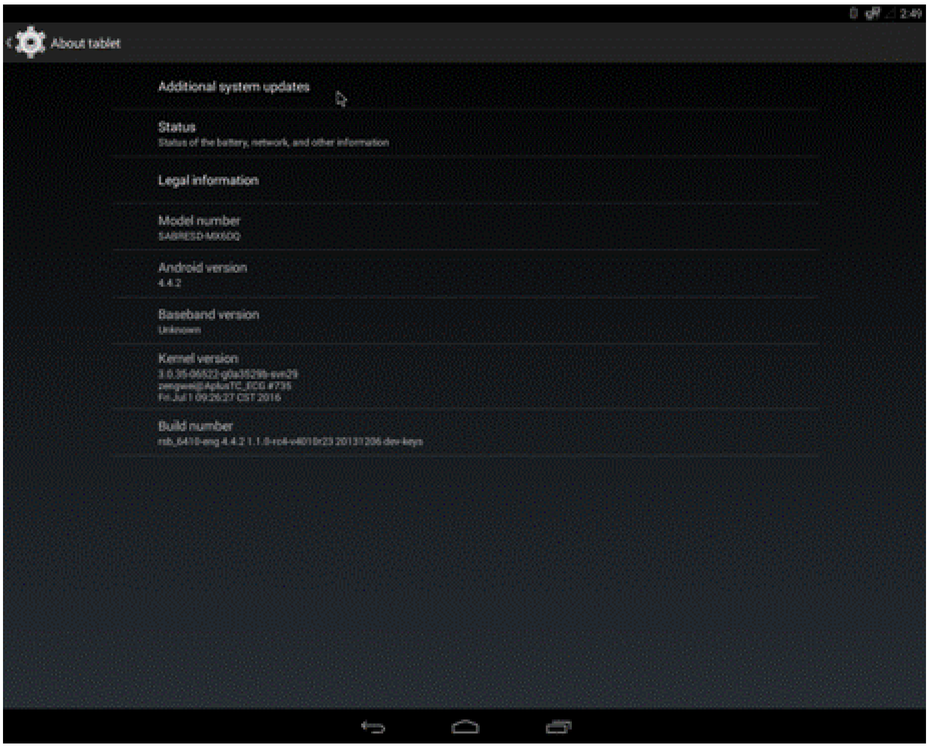


4.11 System Update

1. Refer to section 4.4.5 to build OTA package.
2. Insert your u disk.

Note! *Make sure your u disk is FAT32 and has an update.zip file under the first partition, such as /dev/sdb1.*

3. Click Settings / About tablet / Addition system updates.



Additional system updates

Status
Status of the battery, network, and other information

Legal information

Model number
SABRESD-MX6DQ

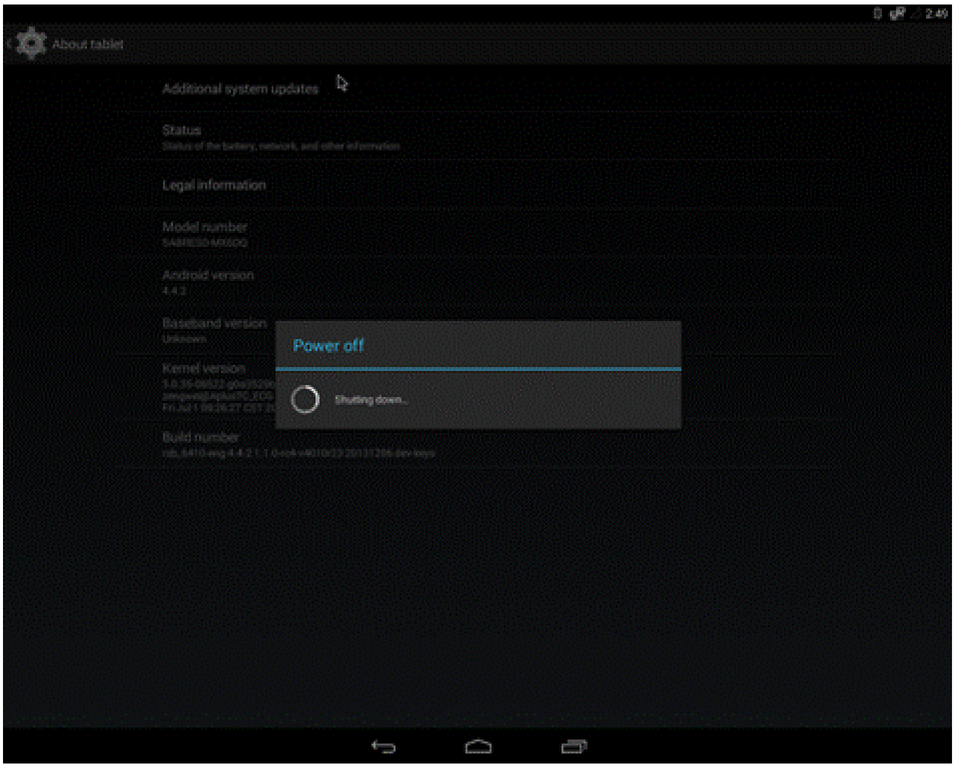
Android version
4.4.2

Baseband version
Unknown

Kernel version
3.0.35-06522-g0a3529b-svn29
zengwei@AplusTC_ECG #735
Fri Jul 1 09:26:27 CST 2016

Build number
rsb_6410-eng 4.4.2 1.1.0-rc4-v4010r23 20131206 dev-keys

4. Wait for a moment; the system will reboot for updating



About tablet

Additional system updates

Status
Status of the battery, network, and other information

Legal information

Model number
SABRESD-MX6DQ

Android version
4.4.2

Baseband version
Unknown

Kernel version
[illegible]

Build number
[illegible]

Power off

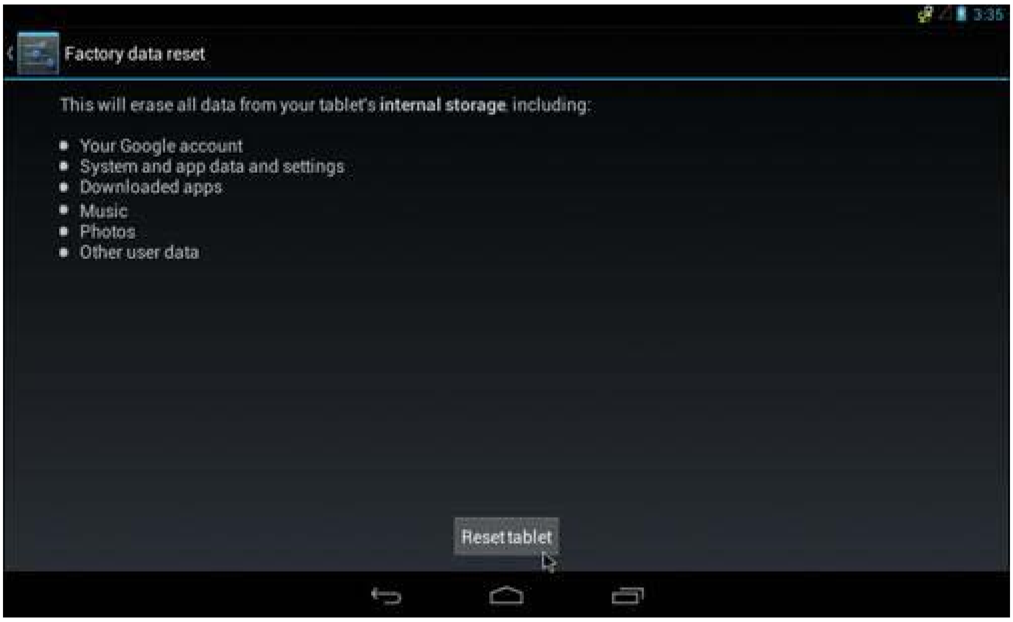
Shutting down...

5. It will take some time to update.

4.12 System Reset

Factory data reset

1. Click settings / backup & reset, then enter into main interface.
2. Click Factory data reset.

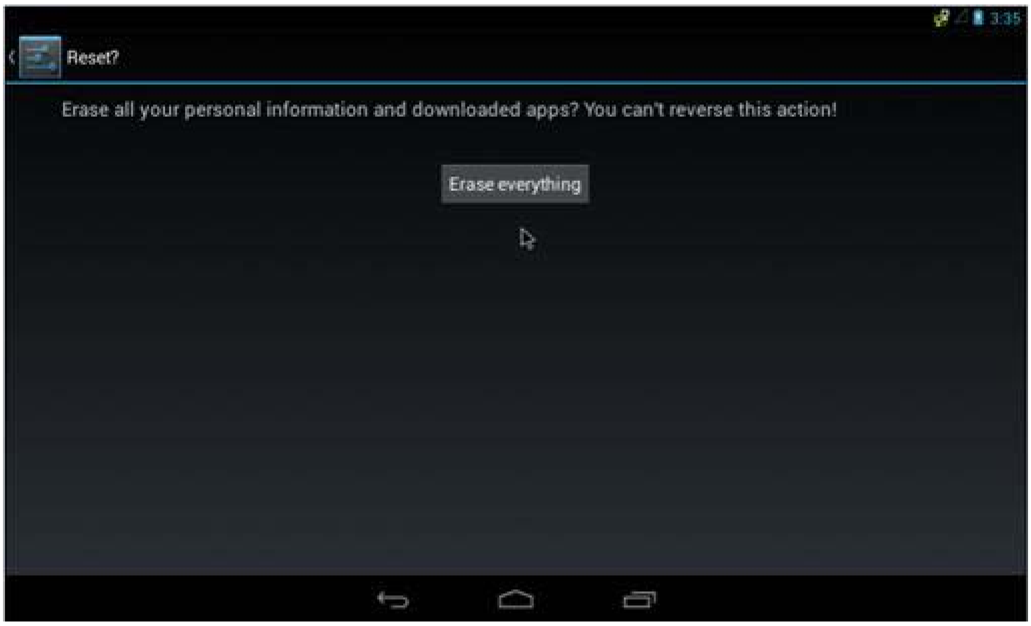


Factory data reset

This will erase all data from your tablet's **internal storage**, including:

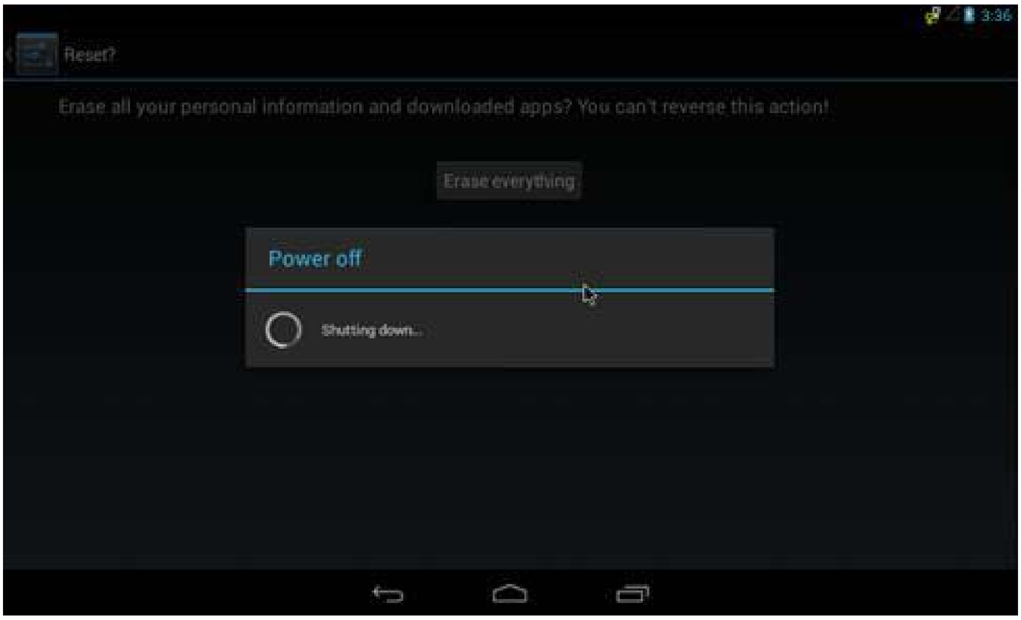
- Your Google account
- System and app data and settings
- Downloaded apps
- Music
- Photos
- Other user data

3. Click Reset to erase all data on the tablet.



Erase all your personal information and downloaded apps? You can't reverse this action!

4. Click Erase everything.



Erase all your personal information and downloaded apps? You can't reverse this action!

Shutting down...

5. Wait for a moment; the system will reboot and erase all user data.

4.13 Watchdog Function

The watchdog program auto starts when Android boots up. The default timeout time is 30 seconds and the watchdog keeps checking every 2 seconds. You can modify these two values below.

1. Edit init.rc(you can find it under **android_source _code/ device/fsl/EPC-R6410/init.rc**) ,find follow line:

   ```
   # [Advantech]Set watchdog timer to 30 seconds and pet it every 10 seconds to get a 20 seconds margin
   service watchdogd /sbin/watchdogd 10 20
   class core
   ```

Note! *The "10" means check time is every 10 secs!*

The "20" plus "10" means watchdog timeout is 30 secs!

So you can modify these two values to fit your requirements.

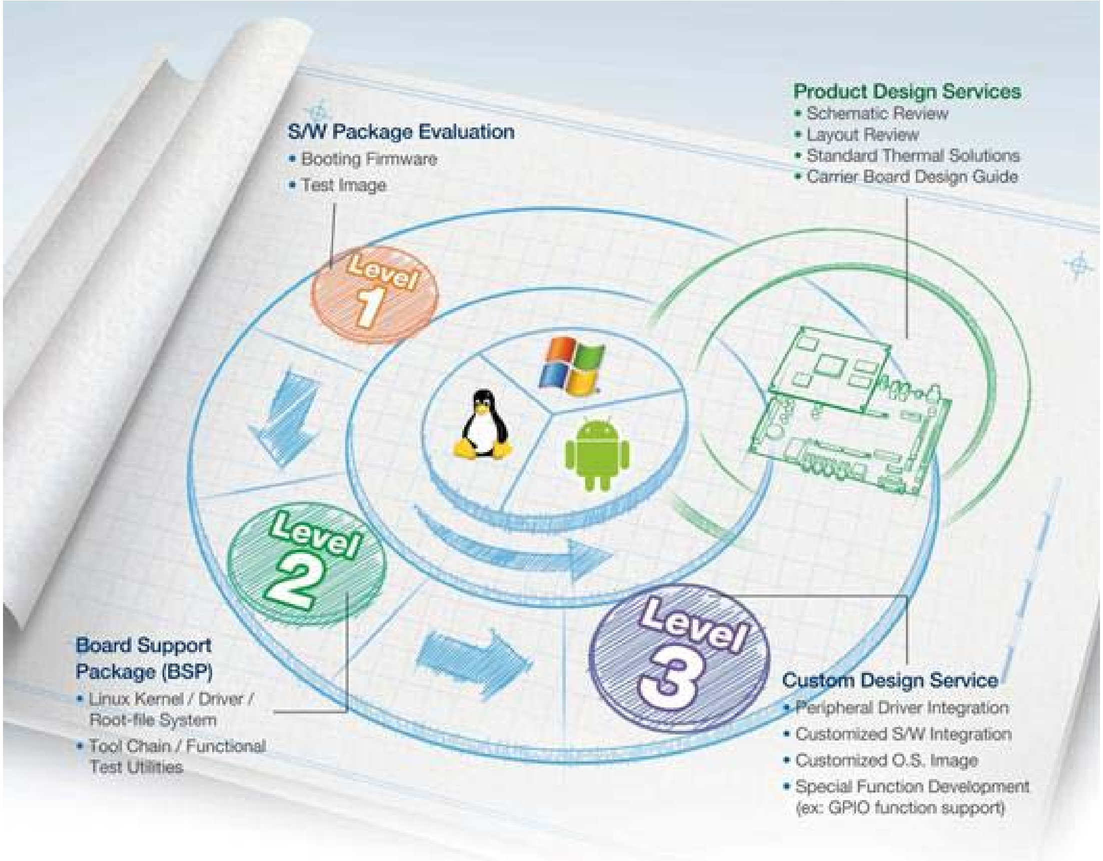
2. Rebuild the kernel image with your modification (refer to 4.4.2 Build android kernel) and create a bootable sd card to test.

Chapter 5

Advantech Services

This chapter introduces Advantech Design-in service, technical support and warranty policy for EPC-R6410 evaluation kit.

5.1 RISC Design-in Services


S/W Package Evaluation
• Booting Firmware
• Test Image
Product Design Services
• Schematic Review
• Layout Review
• Standard Thermal Solutions
• Carrier Board Design Guide
Level 1
Level 2
Level 3
Board Support Package (BSP)
• Linux Kernel / Driver / Root-file System
• Tool Chain / Functional Test Utilities
Custom Design Service
• Peripheral Driver Integration
• Customized S/W Integration
• Customized O.S. Image
• Special Function Development (ex: GPIO function support)

Advantech RISC Design-in Services help customers reduce the time and effort involved with designing new carrier boards. We handle the complexities of technical research and greatly minimize the development risk associated with carrier boards.

Easy Development

Advantech offers firmware support, root file-system, BSP and other develop tools. It helps customers to easily develop their carrier board and differentiate their embedded products and applications.

- Full Range of RISC Product Offerings
- Comprehensive Document Support

Design Assistance Service

Advantech provides a check list for engineers to check customer’s carrier board schematics and compatibility. These services are preventative, and help to catch design errors before they happen. It helps to save a lot of time and cost with regard to developing carrier boards.

- Schematic Review
- Placement and Layout Review
- Debugging Assistance Services
- General/Special Reference Design Database.

Thermal Solution Services

In order to provide quicker and more flexible solutions for customer's thermal designs. Advantech provides thermal solution services including:

- Standard Thermal Solutions
- Customized Thermal Solutions

Embedded Software Services

We support driver development, software integration or customized firmware.

- Embedded Linux/ Android OS
- Advantech boot loader Customization

With the spread of industrial computing, a whole range of new applications have been developed, resulting in a fundamental change in the IPC industry. In the past System Integrators (SI) were used to completing projects without outside assistance but now such working models have moved on. Due to diverse market demands and intense competition, cooperation for (both upstream and downstream) vertical integration has become a much more effective way to create competitive advantage. ARM-based CPU modules were born out of this trend. Concentrating all necessary components on the CPU module and placing other parts on the carrier board in response to market requirements for specialization, provides greater flexibility while retaining low power consumption credentials.

Advantech has been involved in the industrial computer industry for many years and found that customers usually have the following questions when implementing modular designs.

General I/O design capability

Although customers possess the ability for vertical integration and have enough know-how and core competitiveness in the application field, the lack of expertise and experience in general power and I/O design causes many challenges for them, especially integrating CPU modules into their carrier board.

The acquisition of information

Even if an individual client is able to obtain sufficient information to make the right decision for a specialized vertical application, some customers encounter problems dealing with platform design in general and communication with the CPU or chipset, thereby increasing carrier board design difficulties and risk as well as seriously impacting on time-to-market and lost market opportunities.

Software development and modification

Compared to x86 architectures, RISC architectures use simpler instruction sets, therefore the software support for x86 platforms cannot be used on RISC platforms. System Integrators (SI) need to develop software for their system and do the hardware and software integration themselves. Unlike x86 platforms, RISC platforms have less support for Board Support Packages (BSP) and drivers and even though driver support is provided, SI still have to apply a lot of effort to integrate it into the system core. Moreover, the BSP provided by CPU manufacturers are usually for carrier board design, so it’s difficult for SIs to have an environment for software development.

In view of this, Advantech proposed the concept of Streamlined Design-in Support Services for RISC-based Computer On Modules (COM). With a dedicated professional design-in services team, Advantech actively participates in carrier board design and problem solving. Our services not only enable customers to effectively distribute their resources but also reduce R&D manpower cost and hardware investment.

By virtue of a close interactive relationship with leading original manufacturers of CPUs and chipsets such as ARM, TI and Freescale, Advantech helps solve communication and technical support difficulties and that can reduce the uncertainties of product development too. Advantech’s professional software team focuses on providing a complete Board Support Package and assists customers to build up a software development environment for their RISC platforms.

Advantech RISC design-in services helps customers overcome their problems to achieve the most important goal of faster time to market through a streamlined RISC Design-in services.

Along with our multi-stage development process which includes: planning, design, integration, and validation, Advantech’s RISC design-in service provides comprehensive support to the following different phases:

Planning stage

Before deciding to adopt Advantech RISC COM, customers must go through a complete survey process, including product features, specification, and compatibility testing with software. So, Advantech offers a RISC Customer Solution Board (CSB) as an evaluation tool for carrier boards which are simultaneously designed when developing RISC COMs. In the planning stage, customers can use this evaluation board to assess RISC modules and test peripheral hardware. What’s more, Advantech provides a standard software Board Support Package (BSP) for RISC COM, so that customers can define their product’s specifications as well as verifying I/O and performance at the same time. We not only offer hardware planning and technology consulting, but also software evaluation and peripheral module recommendations (such as Wi-Fi, 3G, BT). Resolving customer concerns is Advantech’s main target at this stage. Since we all know that product evaluation is the key task in the planning period, especially for performance and specification, so we try to help our customers conduct all the necessary tests for their RISC COM.

Design stage

When a product moves into the design stage, Advantech will supply a design guide of the carrier board for reference. The carrier board design guide provides pin definitions of the COM connector with limitations and recommendations for final design, so customers can have a clear guideline to follow during their carrier board development. Regarding different form factors, Advantech offers a complete pin-out check list for different form factors such as Q7, ULP and RTX2.0, so that customers can examine the carrier board signals and layout design accordingly. In addition, our team is able to assist customers to review the placement/layout and schematics to ensure the carrier board design meets their full requirements. For software development, Advantech RISC software team can assist customers to establish an environment for software development and evaluate the amount of time and resources needed. If customers outsource software development to a 3rd party, Advantech can also cooperate with the 3rd party and provide proficient consulting services. With Advantech’s professional support, the design process becomes much easier and product quality will be improved to meet their targets.

Integration stage

This phase comprises of HW/SW integration, application development, and peripheral module implementation. Due to the lack of knowledge and experience on platforms, customers need to spend a certain amount of time on analyzing integration problems. In addition, peripheral module implementation has a lot to do with driver designs on carrier boards, RISC platforms usually have less support for ready-made drivers on the carrier board, therefore the customer has to learn from trial and error and finally get the best solution with the least effort. Advantech's team has years of experience in customer support and HW/SW development knowledge. Consequently, we can support customers with professional advice and information as well as shortening development time and enabling more effective product integration.

Validation stage

After customer's ES sample is completed, the next step is a series of verification steps. As a supportive role, Advantech primarily helps customers solve their problems in the testing process and will give suggestions and tips as well. Through an efficient verification process backed by our technical support, customers are able to optimize their applications with less fuss. Furthermore, Advantech's team can provide professional consulting services about further testing and equipment usage, so customers can find the right tools to efficiently identify and solve problems to further enhance their products quality and performance.

5.2 Contact Information

Below is the contact information for Advantech customer service.

Region/Country	Contact Information
North America	1-888-576-9688
Brazil	0800-770-5355
Mexico	01-800-467-2415
Europe (Toll Free)	00800-2426-8080
Singapore & SAP	65-64421000
Malaysia	1800-88-1809
Australia (Toll Free)	1300-308-531
China (Toll Free)	800-810-0345 800-810-8389 Sales@advantech.com.cn
India (Toll Free)	1-800-425-5071
Japan (Toll Free)	0800-500-1055
Korea (Toll Free)	080-363-9494 080-363-9495
Taiwan (Toll Free)	0800-777-111
Russia (Toll Free)	8-800-555-01-50

Also, you can reach our service team through the website below; a technical support engineer will provide a quick response once the form is filled out:

http://www.advantech.com.tw/contact/default.aspx?page=contact_form2&subject=Technical+Support

5.3 Global Service Policy

5.3.1 Warranty Policy

Below is the warranty policy of Advantech products:

5.3.1.1 Warranty Period

Advantech branded off-the-shelf products and 3rd party off-the-shelf products used to assemble Advantech Configure to Order products are entitled to a 2 years complete and prompt global warranty service. Product defect in design, materials, and workmanship, are covered from the date of shipment.

All customized products will by default carry a 15 months regional warranty service. The actual product warranty terms and conditions may vary based on sales contract.

All 3rd party products purchased separately will be covered by the original manufacturer's warranty and time period, and shall not exceed one year of coverage through Advantech.

5.3.1.2 Repairs under Warranty

It is possible to obtain a replacement (Cross-Shipment) during the first 30 days of the purchase, thru your original ADVANTECH supplier to arrange DOA replacement if the products were purchased directly from ADVANTECH and the product is DOA (Dead-on-Arrival). The DOA Cross-Shipment excludes any shipping damage, customized and/or build-to-order products.

For those products which are not DOA, the return fee to an authorized ADVANTECH repair facility will be at the customers' expense. The shipping fee for reconstructive products from ADVANTECH and back to the customers' site will be at ADVANTECH's expense.

5.3.1.3 Exclusions from Warranty

The product is excluded from warranty if

- The product has been found to be defective after expiry of the warranty period.
- Warranty has been voided by removal or alternation of product or part identification labels.
- The product has been misused, abused, or subjected to unauthorized disassembly/modification; placed in an unsuitable physical or operating environment; improperly maintained by the customer; or failure caused which ADVANTECH is not responsible whether by accident or other cause. Such conditions will be determined by ADVANTECH at its sole unfettered discretion.
- The product is damaged beyond repair due to a natural disaster such as a lighting strike, flood, earthquake, etc.
- Product updates/upgrades and tests upon the request of customers who are without warranty.

5.3.2 Repair Process

5.3.2.1 Obtaining an RMA Number

All returns from customers must be authorized with an ADVANTECH RMA (Return Merchandise Authorization) number. Any returns of defective units or parts without valid RMA numbers will not be accepted; they will be returned to the customer at the customer's cost without prior notice.

An RMA number is only an authorization for returning a product; it is not an approval for repair or replacement. When requesting an RMA number, please access ADVANTECH's RMA web site: http://erma.ADVANTECH.com.tw with an authorized user ID and password.

You must fill out basic product and customer information and describe the problems encountered in detail in "Problem Description". Vague entries such as "does not work" and "failure" are not acceptable.

If you are uncertain about the cause of the problem, please contact ADVANTECH's Application Engineers (AE). They may be able to find a solution that does not require sending the product for repair.

The serial number of the whole set is required if only a key defective part is returned for repair. Otherwise, the case will be regarded as out-of-warranty.

5.3.2.2 Returning the Product for Repair

It's possible customers can save time and meet end-user requirements by returning defective products to an y authorized ADVANTECH repair facility without an extra cross-region charge . It is required to contact the local repair center before offering global repair service.

It is recommended to s end cards without accessories (manuals, cables, etc.). Remove any unnecessary components from the card, such as CPU, DRAM, and CF Card. If you send all these parts back (because you believe they may be part of the problem), please note clearly that they are included. Otherwise, ADVANTECH is not responsible for any items not listed. Make sure the " Problem Description " is enclosed.

European Customers that are located outside European Community are requested to use UPS as the forwarding company. We strongly recommend adding a packing list to all shipments. Please prepare a shipment invoice according to the following guidelines to decrease goods clearance time:

1. Give a low value to the product on the invoice, or additional charges will be levied by customs that will be borne by the sender.
2. Add information "Invoice for customs purposes only with no commercial value" on the shipment invoice.
3. Show RMA numbers, product serial numbers and warranty status on the shipment invoice.
4. Add information about Country of origin of goods

In addition, please attach an invoice with RMA number to the carton, then write the RMA number on the outside of the carton and attach the packing slip to save handling time. Please also address the parts directly to the Service Department and mark the package "Attn. RMA Service Department".

All products must be returned in properly packed ESD material or anti-static bags. ADVANTECH reserves the right to return unrepaired items at the customer's cost if inappropriately packed.

Besides that, "Door-to-Door" transportation such as speed post is recommended for delivery, otherwise, the sender should bear additional charges such as clearance fees if Air-Cargo is adopted.

Should DOA cases fail, ADVANTECH will take full responsibility for the product and transportation charges. If the items are not DOA, but fail within warranty, the sender will bear the freight charges. For out-of-warranty cases, customers must cover the cost and take care of both outward and inward transportation.

5.3.2.3 Service Charges

The product is excluded from warranty if :

- The product is repaired after expiry of the warranty period.
- The product is tested or calibrated after expiry of the warranty period, and a No Problem Found (NPF) result is obtained.
- The product, though repaired within the warranty period, has been misused, abused, or subjected to unauthorized disassembly/modification; placed in an unsuitable physical or operating environment; improperly maintained by the customer; or failure caused which ADVANTECH is not responsible whether by accident or other cause. Such conditions will be determined by ADVANTECH at its sole unfettered discretion.
- The product is damaged beyond repair due to a natural disaster such as a lighting strike, flood, earthquake, etc.
- Product updates and tests upon the request of customers who are without warranty.

If a product has been repaired by ADVANTECH, and within three months after such a repair the product requires another repair for the same problem, ADVANTECH will do this repair free of charge. However, such free repairs do not apply to products which have been misused, abused, or subjected to unauthorized disassembly/modification; placed in an unsuitable physical or operating environment; improperly maintained by the customer; or failure caused which ADVANTECH is not responsible whether by accident or other cause.

Please contact your nearest regional service center for detail service quotation.

Before we start out-of-warranty repairs, we will send you a pro forma invoice (P/I) with the repair charges. When you remit the funds, please reference the P/I number listed under "Our Ref". ADVANTECH reserves the right to deny repair services to customers that do not return the DOA unit or sign the P/I. Meanwhile, ADVANTECH will scrap defective products without prior notice if customers do not return the signed P/I within 3 months.

5.3.2.4 Repair Report

ADVANTECH returns each product with a "Repair Report" which shows the result of the repair. A "Repair Analysis Report" is also provided to customers upon request. If the defect is not caused by ADVANTECH design or manufacturing, customers will be charged US$60 or US$120 for in-warranty or out-of-warranty repair analysis reports respectively.

5.3.2.5 Custody of Products Submitted for Repair

ADVANTECH will retain custody of a product submitted for repair for one month while it is waiting for return of a signed P/I or payment (A/R). If the customer fails to respond within such period, ADVANTECH will close the case automatically. ADVANTECH will take reasonable measures to stay in proper contact with the customer during this one month period.

5.3.2.6 Shipping Back to Customer

The forwarding company for RMA returns from ADVANTECH to customers is selected by ADVANTECH. Per customer requirement, other express services can be adopted, such as UPS, FedEx and etc. The customer must bear the extra costs of such alternative shipment. If you require any special arrangements, please indicate this when shipping the product to us.

ADVANTECH

Enabling an Intelligent Planet

www.advantech.com

Please verify specifications before quoting. This guide is intended for reference purposes only.

All product specifications are subject to change without notice.

No part of this publication may be reproduced in any form or by any means, electronic, photocopying, recording or otherwise, without prior written permission of the publisher.

All brand and product names are trademarks or registered trademarks of their respective companies.

© Advantech Co., Ltd. 2017