



QY-1808EK Linux测试手册

版本号 v1.0

2013年07月

杭州启扬智能有限公司版权所有
QIYANG TECHNOLOGY Co., Ltd
Copyright Reserved

前言

欢迎使用杭州启扬智能科技有限公司产品 QY-1808EK，本产品 Linux 部分包含 3 份手册：QY-1808EK linux 用户手册、QY-1808EK 硬件说明以及 QY-1808EK Linux 测试手册。硬件相关部分可以参考 QY-1808EK 硬件说明，主板测试可以参考 QY-1808EK Linux 测试手册。

使用之前请仔细阅读 QY-1808EK Linux 用户手册以及 QY-1808EK 硬件说明！

公司简介

杭州启扬智能科技有限公司位于美丽的西子湖畔,是一家集研发、生产、销售为一体的高新技术产业。公司致力于成为嵌入式解决方案的专业提供商,为嵌入式应用领域客户提供硬件开发工具和嵌入式系统完整解决方案。产品范围主要包括: Cirrus Logic EP93xx 系列 ARM9 主板、ATMEL AT91SAM926x 系列主板, FreeScale iMX 系列主板, TI Davinci 系列音视频通用开发平台等等。可运行 Linux2.4/2.6、WinCE5.0/6.0 操作系统, 并可根据客户需求开发各种功能组合的嵌入式硬件系统。应用领域涉及: 工业控制、数据采集、信息通讯、医疗设备、视频监控、车载娱乐等等。

客户的需求是公司发展的动力, 公司将不断完善自身, 与客户互助互惠, 共同发展。

电话: 0571-87858811, 87858822

传真: 0571-87858822

技术支持 E-MAIL: support@qiyangtech.cn

网址: <http://www.qiyangtech.com>

地址: 杭州市西湖科技园西园一路 8 号 3A 幢 5 层

邮编: 310012

修改历史记录

版本号	时间	修订内容
v1.0	2013-07	创建

目录

一、准备工作.....	5
1.1 主机要求.....	5
1.2 硬件配置.....	5
1.3 主板要求.....	5
1.4 软件准备.....	5
二、主板测试.....	7
2.1 蜂鸣器测试.....	7
2.2 RTC 测试.....	7
2.3 LCD 触摸屏测试.....	8
2.4 串口测试.....	9
2.5 音频测试.....	11
2.6 USB 测试.....	12
2.7 SD 卡测试.....	12
2.8 扩展板 I/O, ADC 测试.....	13

一、准备工作

1.1 主机要求

- 1、装有 Linux 系统（redhat7 、 8、 9 或其它 Linux 发行版）和 Windows（最好为 2000 以上系统）；
- 2、到少 500MB 自由空间；
- 3、一个 RS232 串口；
- 4、终端通讯软件（迷你终端或 Windows 下的超级终端）；
- 5、以太网线；
- 6、一个 RS232 转 RS485 接口；

1.2 硬件配置

- 1、一根 RS2323 串口线，两端母头的交叉线，三线就行了（RXD，TXD 和 GND）。一端连 PC 机 COM1 一端连主板的调试串口；
- 2、接上主板电源；

1.3 主板要求

以下的测试用例都在主板的 Linux 环境下运行，请确认主板已经正常启动 Linux 系统。

1.4 软件准备

- 1、打开迷你终端（光盘中附带），设置 PC 上相应的串口，选择[文件] →[系统设置] →[通讯设置] →左下角[断开] →[串口设置]，波特率为[115200]，停止位为[1]位，数据位为[8]位，无奇偶校验和流控。如下图所示：



2、点击工具条上的[通讯] → [建立连接]按钮建立连接:



3、给主板上电，系统自动启动 Linux 系统。

4、进入 test 目录

```
#cd /usr/test
```

```
#ls
```

可以看到 test 目录下有预先编译好的测试程序。运行各个测试程序可以看到相关的测试结果。

二、主板测试

2.1 蜂鸣器测试

- 进入/usr/test 目录，运行 buzzer_test 程序

```
# cd /usr/test/
```

打开蜂鸣器

```
# ./buzzer_test 1
```

关闭蜂鸣器

```
# ./buzzer_test 0
```

- 设备文件 /dev/buzzer
- 测试代码： buzzer-test/buzzer_test.c
- 驱动代码： linux-2.6.37-am1808/drivers/misc
- 内核对应选项：
Device Drivers --->
Misc devices --->
BUZZER FOR QIYANG BOARD

2.2 RTC 测试

Linux2.6 内核已经包含对 RTC 的支持，用户可以通过以下方式测试：

1、通过 date 命令设置当前时间

```
$ date 072210502010 //设置当前时间为 2010-07-22 10:50:00
```

2、通过 hwclock 写入硬件 RTC

```
$ hwclock -w
```

3、断电后重新启动系统，用 date 命令可以查看当前时间

```
$ date
```

有任何技术问题或需要帮助，请联系：supports@qiyangtech.com

购买产品，请联系销售：sales@qiyangtech.com

更多信息请访问：<http://www.qiytech.com>

©2012 Qiyangtech 版权所有

4、如果要通过程序读 RTC 时间，可以参考 rtc_test.c

先到测试程序目录下

```
# cd /usr/test
```

```
# ./rtc_test
```

程序运行后，会连续读 10 秒钟的 RTC 时间并在串口打印。

- 测试结果：

可以得到如下结果

```
# ./rtc_test

                RTC Driver Test Example.
Current RTC date/time is 11-7-2013, 10:23:32.
Current RTC date/time is 11-7-2013, 10:23:33.
Current RTC date/time is 11-7-2013, 10:23:34.
Current RTC date/time is 11-7-2013, 10:23:35.
Current RTC date/time is 11-7-2013, 10:23:36.
Current RTC date/time is 11-7-2013, 10:23:37.
Current RTC date/time is 11-7-2013, 10:23:38.
Current RTC date/time is 11-7-2013, 10:23:39.
Current RTC date/time is 11-7-2013, 10:23:40.
Current RTC date/time is 11-7-2013, 10:23:41.

                *** Test complete ***
```

·测试代码： rtc_test/rtc_test.c

·驱动代码： linux-2.6.37/driver/rtc/rtc-ds1307.c

- 内核对应选项：

Device Drivers --->

Real Time Clock --->

.....

- 可能遇到的问题：

- 运行 hwclock -w 命令的时候提示出错：

hwclock: Could not access RTC: No such file or directory

- 或者是运行 rtc_test 的时候提示：

/dev/rtc: No such file or directory

※请检查/dev 目录下是否有 rtc 和 rtc0，rtc 为 rtc0 的软链接如果没有 rtc 则用以下命令创建：

```
# cd /dev
```

```
# ln -s rtc0 rtc
```

如果没有 rtc0 则用以下命令创建：

```
# mknod rtc0 c 254 0
```

※如果/dev 目录下有 rtc 和 rtc0 并正确则检查内核 rtc 配置。

2.3 LCD 触摸屏测试

连接好触摸屏，选择好 LCD 电源跳线，详细说明请参考 QY-1808EK 硬件说明书.doc
给板子上电后如果正常可以在 LCD 屏左上角看到 linux 企鹅的 logo，进入根文件系统

后可以使用 `ts_calibrate` 对触摸屏进行校准。

如若 LCD 屏不能显示，请检查 LCD 电源跳线，以及内核 LCD 屏大小选择

内核 LCD 屏分辨率设置选项

Device Drivers-->

Graphics support->

Support for frame buffer devices->

DA8xx/OMAP-L1xx/AM1xxx Framebuffer support->

选择相应的大小

2.4 串口测试

2.4.1 PRU 串口测试

在测试之前确认在内核选项中是否关闭

Device Drivers ---->

Sound Card Support -->

.....

到测试程序目录下

```
# /usr/test/
```

先加载 `suart_emu.ko` 模块，命令如下：

```
# insmod suart_emu.ko
```

加载成功后会出现如下信息：

```
# insmod suart_emu.ko
ti_omap1_pru_suart ti_omap1_pru_suart.1: fw size 3700, downloading..
ti_omap1_pru_suart.1: ttySU0 at MMIO 0x1d00000 (irq = 3) is a suart_tty
ti_omap1_pru_suart.1: ttySU1 at MMIO 0x1d00000 (irq = 4) is a suart_tty
ti_omap1_pru_suart.1: ttySU2 at MMIO 0x1d00000 (irq = 5) is a suart_tty
ti_omap1_pru_suart.1: ttySU3 at MMIO 0x1d00000 (irq = 6) is a suart_tty
ti_omap1_pru_suart.1: ttySU4 at MMIO 0x1d00000 (irq = 7) is a suart_tty
ti_omap1_pru_suart.1: ttySU5 at MMIO 0x1d00000 (irq = 8) is a suart_tty
ti_omap1_pru_suart.1: ttySU6 at MMIO 0x1d00000 (irq = 9) is a suart_tty
ti_omap1_pru_suart.1: ttySU7 at MMIO 0x1d00000 (irq = 10) is a suart_tty
ti_omap1_pru_suart ti_omap1_pru_suart.1: ti_omap1_pru_suart device registered(pru_clk=228000000, asp_clk=150000000)
```

加载可能出现的问题：

(1) 提示：`ti_omap1_pru_suart ti_omap1_pru_suart.1: can't load firmware;`

解决方法：查看文件系统目录下 `/lib/firmware/PRU_SUART_Emulation.bin` 文件是否存在，若没有，则到测试源码 `/serial_test` 目录下将 `PRU_SUART_Emulation.bin` 复制到开发板的 `/lib/firmware/` 目录下即可。

打开测试程序：

```
# cd /
```

```
# cd /usr/test
```

```
# ./serial_test 0~7
```

红色部分表示输入 0~7 之间的任意一个数，分别表示 `/dev` 下设备文件 `ttySU0 ~ ttySU7`。

例：当输入的命令是 `./serial_test 0` 时，就是对 `ttySU0` 进行 `pru` 测试，此时将开发板上 J5 插座的 3, 4 引脚用跳帽短接，会出现以下信息：

有任何技术问题或需要帮助，请联系：supports@qiyangtech.com

第 9 页 共 15 页

购买产品，请联系销售：sales@qiyangtech.com

更多信息请访问：<http://www.qiytech.com>

©2012 Qiyangtech 版权所有

如带有 LCD 屏 可以使用 `pru-8uart-test` 程序对 `pruss` 进行测试
如没有 LCD 屏 使用上述测试普通串口的方式即可。

2.4.2 RS232 串口测试

在测试之前需要先把底板的 R34,R35 焊掉, 并且把 R21,R18 换成 0 欧姆的电阻。
串口测试程序位于 `/usr/test` 目录下, 测试如下:

```
# cd /usr/test
# ./serial_test 8
```

终端打印如下信息:

```
Serial test start.
```

此时将串口线插入到开发板的 COM1 接口, 会持续打印出 “this is a Serial_Suart test!”。
rs232 的串口设备文件在 `/dev` 下 `ttyS1`

·测试代码: `serial_test/serial_test.c`

·驱动代码: `linux-2.6.37/driver/serial/8250.c`

• 可能遇到问题的解决方法:

- 1) 确认设备节点 `/dev/ ttyS1`
- 2) 确认串口是否接好, 并且设置好(波特率为 115200 数据位为 8 停止位为 1 无校验位)

- 3) 确认内核选项是否选上

```
Device Drivers --->
```

```
Char devices --->
```

```
Serial drivers->
```

```
8250.....
```

- 4) 其它

2.4.3 RS485 串口测试

在测试之前需要先把底板的 R18,R21 焊掉, 并且把 R34,R35 换成 0 欧姆的电阻; 准备一个 STM485 转换器, 转换器一端与电脑串口线相连接, 另一端的 A 口与底板 `com1(J10)` 的第三引脚相连接, B 口与底板 `com1(J10)` 的第二引脚相连接。

串口测试程序位于 `/usr/test` 目录下, 测试如下:

```
# cd /usr/test
# ./rs485_serial_test
```

无错运行后, 在命令终端能不断打印出 0。

• 测试代码: `rs485_serial_test/rs485_serial_test.c`

• 驱动代码: `linux-2.6.37/driver/serial/8250.c`

可能遇到问题的解决方法:

- 1) 确认设备节点 `/dev/ ttyS1`
- 2) 确认串口是否接好, 并且设置好(波特率为 115200 数据位为 8 停止位为 1 无校验位)

3) 确认内核选项是否选上

```
Device Drivers --->
  Char devices --->
    Serial drivers->
      8250.....
```

和

```
Device Drivers --->
  Misc devices --->
    Am1808 RS485 CTRL
```

4) 其它

2.5 音频测试

2.5.1 音频播放

先将音频文件复制到开发板中，文件系统中也有一个音频文件

```
# cd /
```

```
# aplay test.wav
```

此时将耳机插到板子上的 Headphone(J16)可以听到流畅的音乐，并打印出以下信息：

```
# aplay test.wav
Playing WAVE 'test.wav' : Unsigned 8 bit, Rate 22050 Hz, Mono
```

2.5.2 音量调节

查看当前音频系统所提供的音频接口

```
# amixer controls
```

打印信息如图所示：

```
# amixer controls
numid=1,iface=MIXER,name='PCM Playback Volume'
numid=22,iface=MIXER,name='Line DAC Playback Volume'
numid=20,iface=MIXER,name='Line Line2 Bypass Volume'
numid=21,iface=MIXER,name='Line PGA Bypass Volume'
numid=32,iface=MIXER,name='Line Playback Switch'
numid=25,iface=MIXER,name='Mono DAC Playback Volume'
numid=23,iface=MIXER,name='Mono Line2 Bypass Volume'
numid=66,iface=MIXER,name='Mono Mixer DACL1 Switch'
numid=69,iface=MIXER,name='Mono Mixer DACR1 Switch'
numid=64,iface=MIXER,name='Mono Mixer Line2L Bypass Switch'
numid=67,iface=MIXER,name='Mono Mixer Line2R Bypass Switch'
numid=65,iface=MIXER,name='Mono Mixer PGAL Bypass Switch'
numid=68,iface=MIXER,name='Mono Mixer PGAR Bypass Switch'
numid=24,iface=MIXER,name='Mono PGA Bypass Volume'
numid=33,iface=MIXER,name='Mono Playback Switch'
numid=39,iface=MIXER,name='ADC HPF Cut-off'
numid=36,iface=MIXER,name='AGC Switch'
numid=28,iface=MIXER,name='HP DAC Playback Volume'
```

我们这里设置 Headphone 接口的音量，具体设置方法是：

1、打开接口

```
# amixer cset numid=34 on
numid=3,iface=MIXER,name='Headphone Playback Switch'
; type=BOOLEAN,access=rw---,values=2
: values=on,on
```

2、设置音量大小

```
# amixer cset numid=28 0
numid=4,iface=MIXER,name='Headphone Playback Volume'
; type=INTEGER,access=rw---,values=2,min=0,max=31,step=0
: values=0,0
```

此时，音量变小了

```
# amixer cset numid=4 100
numid=4,iface=MIXER,name='Headphone Playback Volume'
; type=INTEGER,access=rw---,values=2,min=0,max=31,step=0
: values=100,100
```

而此时，音量变大了。

注：以上红色标注的值为音量的设置值，范围在 0~118。

2.6 USB 测试

首先在开发板的 USB -S(J14)插入 U 盘，此时串口终端会出现如下信息：

```
# usb 2-1: new full speed USB device using ohci and address 2
usb 2-1: not running at top speed; connect to a high speed hub
scsi0 : usb-storage 2-1:1.0
scsi 0:0:0:0: Direct-Access      Sony      Storage Media    0100 PQ: 0 ANSI: 4
sd 0:0:0:0: [sda] 30523392 512-byte logical blocks: (15.6 GB/14.5 GiB)
sd 0:0:0:0: [sda] Write Protect is off
sd 0:0:0:0: Attached scsi generic sg0 type 0
sd 0:0:0:0: [sda] Assuming drive cache: write through
sd 0:0:0:0: [sda] Assuming drive cache: write through
sda: sda1
sd 0:0:0:0: [sda] Assuming drive cache: write through
sd 0:0:0:0: [sda] Attached SCSI removable disk
sda1 add
```

这时，系统会自动将优盘挂载到/media/sda1 目录下。

2.7 SD 卡测试

首先在开发板的(J15)插入 SD 卡，此时串口终端会出现如下信息：

```
# mmc0: new high speed SDHC card at address aaaa
mmcblk0: mmc0:aaaa SU04G 3.69 GiB
  mmcblk0: p1
mmcblk0p1 add
```

这时，系统会自动将 SD 卡挂载到/media/mmcblk0p1 目录下。

2.8 扩展板 I/O, ADC 测试

2.8.1 I/O 测试

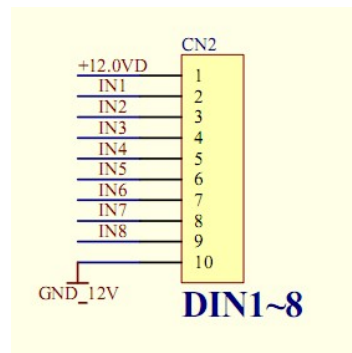
将扩展板与开发板的 AD/DA-DIG (J3) 相连接。该 I/O 测试分为开关信号输入和开关量输出两部分，接下去分别对两部分进行测试，测试程序位于/usr/test 下面:

```
# cd /usr/test
# ./key_signal_test
出现信息:
```

```
# ./key_signal_test
key_signal_test begin...
Input the num(1-8) from keyoard please!
If the value of input except 1-8,the test will exit!
Input the pin(1~8):
```

开关信号输入部分测试:

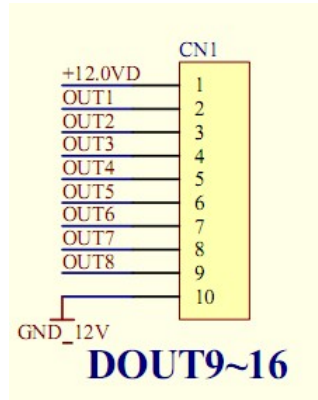
该部分测试，不需要在终端输入任何信息，直接给扩展板的 CN2 接口（如下图所示）



的 2~9 中任意引脚加 12v 电压，与电阻 R65 相连接的是 IN1。例：给 IN4 一个高电平，屏幕会打印出 “IN4 is inputing high lever”。

开关量输出部分测试:

该部分测试，你可以输入 1~8 之间的数字，表示控制扩展板上 CN1（如下图所示）



上 OUTx (x 表示输入的数字) 输出电平, 与电阻 R67 相连接的引脚是 OUT1。

例: 在终端输入 1, 出现如下信息:

```
Input the pin(1~8):1
Input value(0 or 1):
```

此时在终端输入 0 或 1, 0 表示让 CN1 的 OUT1 输出低电平, 1 表示让 CN1 的 OUT1 输出高电平。

最后用万能表测量下对应的引脚的电压是否正确。

- 设备文件 /dev/key_signal
- 测试代码: key_signa_test/key_signa_test.c
- 驱动代码: linux-2.6.37-am1808/drivers/misc/key_signa_driver.c
- 内核对应选项:

Device Drivers --->

Misc devices --->

AM1808 KEY SIGNAL DRIVER

2.8.2 ADC 测试

杭州启扬智能科技有限公司

电话: **0571-87858811 / 87858822**

传真: **0571-89935912**

支持: **0571-89935913**

E-MAIL: **supports@qiyangtech.com**

网址: <http://www.qiytech.com>

地址: 杭州市西湖区西湖科技园西园 1 路 8 号 3A 幢 5F 楼

邮编: **310012**