

---

# BB Black 扩展板 用户手册

---

## 版权声明

本手册所有权由深圳市微雪电子有限公司独家持有。未经本公司的书面许可，不得以任何方式或形式进行修改、分发或复制本文档的任何部分，否则一切后果由违者自负。

## 目录

BB Black 扩展板 用户手册.....	1
版权声明.....	1
1 准备工作.....	2
1.1 TF 卡系统镜像的烧写.....	2
1.2 电脑端部署通信环境.....	3
1.3 进入 Bash Shell 环境.....	4
1.4 API 源码.....	5
2 LCD CAPE.....	6
2.1 LCD CAPE 产品概述.....	6
2.2 LCD CAPE 使用说明.....	8
3 MISC CAPE.....	11
3.1 MISC CAPE 产品概述.....	11
3.2 LED 测试.....	12
3.3 蜂鸣器测试.....	12
3.4 DS18B20 测试.....	12
3.5 按键测试.....	13
3.6 RTC 测试.....	13
4 RS485 CAN CAPE.....	15
4.1 CAN 测试.....	15
4.2 RS485 测试.....	16
5 其他套件.....	18
5.1 USB Camera 测试.....	18
5.2 USB WIFI 测试.....	20

# 1 准备工作

## 1.1 TF 卡系统镜像的烧写

烧写我们提供的基于 TF 卡启动的 Angstrom 镜像。操作如下：

1) 解压系统镜像：

使用 7z920.exe 压缩工具解压相应的镜像文件 XXX.img.7z。

注意：“XXX”是泛指各种镜像文件名称，用户根据自己的扩展板模块选择相应的系统镜像，点击下面链接下载：

**LCD CPAE(4.3inch)测试镜像：**

<http://www.waveshare.net/wiki/LCD-CAPE-4.3inch-IMG>

**LCD CPAE(7inch)测试镜像：**

<http://www.waveshare.net/wiki/LCD-CAPE-7inch-IMG>

**MISC CAPE 和 RS485/CAN CAPE 测试镜像：**

<http://www.waveshare.net/wiki/MISC-CAPE-IMG>

说明：MISC CAPE 和 RS485/CAN CAPE 共用一个镜像文件。

2) 使用 HPUUSBDisk.exe 格式化 TF 卡。选择要格式化的 TF 卡，选择格式化为 FAT32，点击 **Start**。

注意：必须确保 TF 卡的容量不低于 4GB！

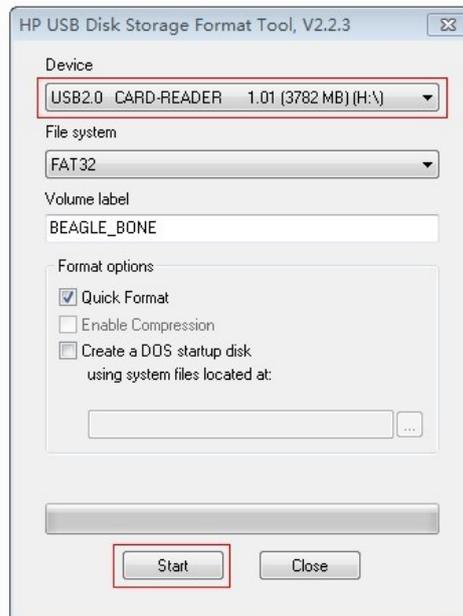


图 1. 使用 HPUUSBDisk.exe 格式化 TF 卡

3) 烧写系统镜像

打开 Win32DiskImager.exe，选择 1)解压出来的系统镜像，点击 **Write** 进行烧写。

烧写完毕，取下 TF 卡。

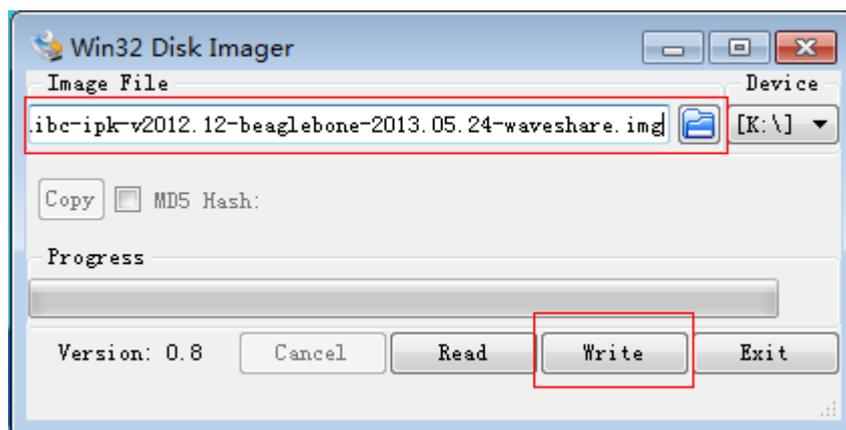


图 2. 使用 Win32DiskImager.exe 烧写系统镜像

## 1.2 电脑端部署通信环境

- 1) 通过 USB 转 TTL 线（内置 PL2303 芯片，兼容 Windows 8 系统），把 BB Black 的 DEBUG 接口连接到计算机。



接口定义：  
红色：VCC  
黑色：GND  
绿色：TXD(接外部设备 RXD)  
白色：RXD(接外部设备 TXD)

图 3. USB 转 TTL 线

- 2) 打开 **PL2303\_Prolific\_DriverInstaller\_v1.8.0.exe** 进行驱动的安装。
- 3) 成功安装驱动后，打开串口查看软件 PuTTY.exe，对下图红框位置进行设置。

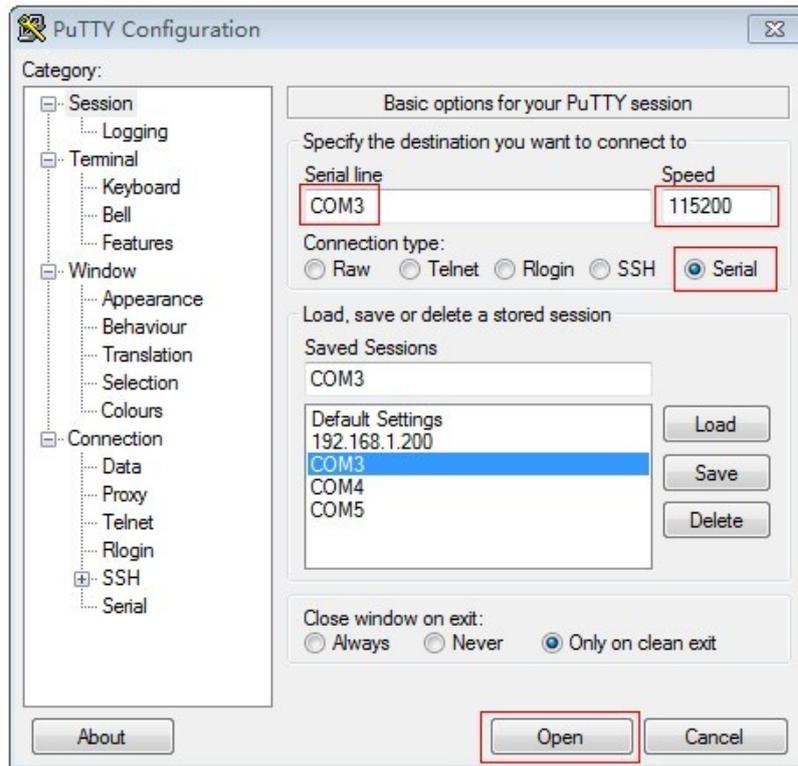


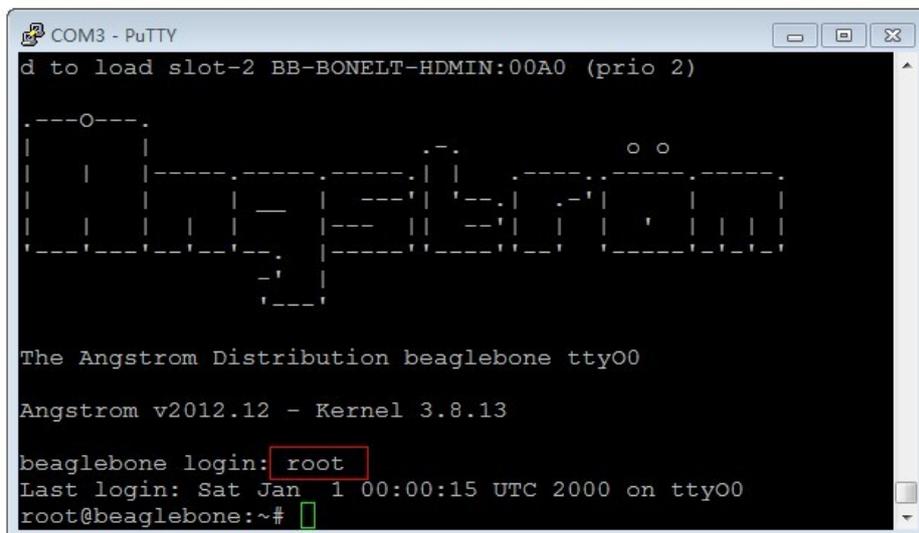
图 4. PuTTY 界面

说明：

- **Serial line:** 选择对应的串口。图中所示是 COM3，但用户应按照实际设置。（通过设备管理器可以查看）
  - **Speed:** 设置波特率为 115200。
  - **Connection type:** 设置为 Serial。
- 4) 点击 **Open**。

### 1.3 进入 Bash Shell 环境

- 1) 把烧写好镜像的 TF 卡插入 BB Black 的卡槽中，按住 BOOT 按钮，然后上电（上电后可以松开 BOOT 按钮）。
- 2) 当系统启动后，输入“root”，即可进入 Bash Shell 环境，可以输入 shell 命令。后面提到的操作命令都是在此终端进行。



```
COM3 - PuTTY
d to load slot-2 BB-BONELT-HDMIN:00A0 (prio 2)

The Angstrom Distribution beaglebone ttyO0

Angstrom v2012.12 - Kernel 3.8.13

beaglebone login: root
Last login: Sat Jan 1 00:00:15 UTC 2000 on ttyO0
root@beaglebone:~#
```

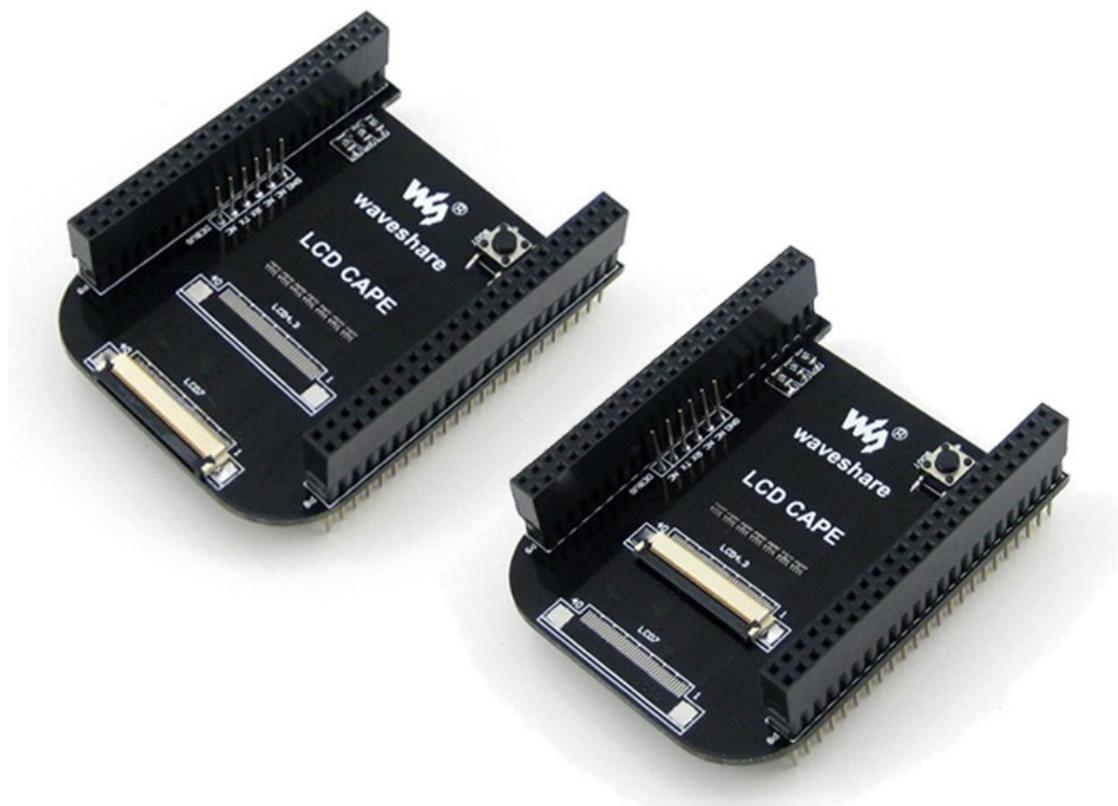
图 5. 进入 Bash Shell

## 1.4 API 源码

本手册用到的 API 源码在系统镜像/home/xuser/waveshare\_demo/API 目录下。

## 2 LCD CAPE

### 2.1 LCD CAPE 产品概述



本产品主要用于给 Beaglebone Black (BB Black) 扩展我司提供的 4.3 寸和 7 寸电阻屏。所有功能均是针对 BB Black 开发板所推出的，它本身不能单独使用。分为两个版本，分别对应 4.3 寸和 7 寸电阻屏。

## LCD CAPE (4.3inch)板载资源



1. **BB\_BLACK 接口**：方便接入 BB\_BLACK
2. **4.3inch 屏接口**：方便接入 4.3 寸电阻屏
3. **DEBUG 接口**：BB\_BLACK 的调试接口，方便用户外接不同的串口模块
4. **启动选择按键**：方便用户从 TF 卡启动

## LCD CAPE (7inch)板载资源



1. **BB\_BLACK 接口**：方便接入 BB\_BLACK
2. **4.3inch 屏接口**：方便接入 4.3 寸电阻屏
3. **DEBUG 接口**：BB\_BLACK 的调试接口，方便用户外接不同的串口模块
4. **启动选择按键**：方便用户从 TF 卡启动

## 2.2 LCD CAPE 使用说明

1) 连接屏幕。

**注意：**本产品用于连接 4.3 寸屏或 7 寸屏的时候，使用的接口是不同的。本产品部分批次同时提供了两种接口，如果用户把屏幕接到错误的接口，将会损坏主板和屏幕。请务必注意区分！

LCD CAPE (4.3inch)和 LCD CAPE (7inch)两者的区分点在于 FFC 排线的接口。



图 6. 左边 LCD CAPE (4.3inch)，右边 LCD CAPE (7inch)

- 4.3 寸屏连接

系统断电,把 4.3 寸屏接到 LCD4.3 接口, 如图 7.所示:

确保烧录的是 LCD CPAE(4.3inch)测试镜像(参见 1.1)。BB Black 的 P1 电源座接入电源适配器。



图 7. 连接 4.3 寸屏

- 7 寸屏显示测试

系统断电,把 7 寸屏接到 LCD7 接口, 如图 8 所示:

确保烧录的是 LCD CPAE(7inch)测试镜像(参见 1.1), BB Black 的 P1 电源座接入电源适配器。

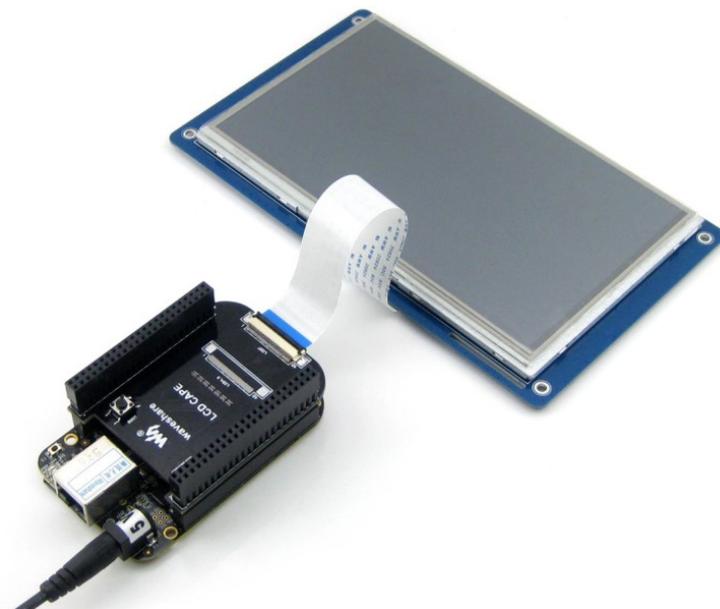


图 8. 连接 7 寸屏

## 2) 触摸屏校准，输入命令：

Angstrom 系统：

```
root@beaglebone:~# rm -rf /etc/pointercal*
```

```
root@beaglebone:~# ts_calibrate
```

运行该命令后，LCD 屏上提示 5 点校准，分别点击各点完成校准。

```
root@beaglebone:~# sync
```

直接开关电源或者终端执行 `reboot` 命令重启系统，用户触摸屏幕，有反馈则正常。

**注意：**如果存在触摸不准的情况，则再次进行校准或重启系统。

Debian 系统：

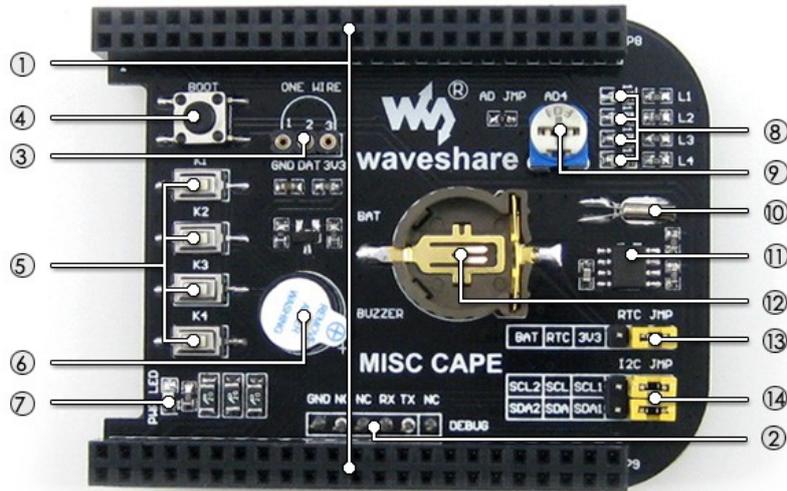
```
root@beaglebone:~# rm -rf /etc/pointercal*
```

```
root@beaglebone:~# sync
```

重新启动系统，校准程序会自动启动，校准后即可正常使用触摸功能。

### 3 MISC CAPE

#### 3.1 MISC CAPE 产品概述



##### [核心接口简介]

1. **BB\_BLACK 接口**  
方便接入 BB\_BLACK
2. **DEBUG 接口**  
BB\_BLACK 的调试接口，方便用户外接不同的串口模块
3. **ONE-WIRE 接口**  
方便接入各类 1-WIRE 接口器件（TO-92 封装），  
如温度传感器 DS18B20、电子注册码 DS2401 模块等

##### [器件介绍]

4. **启动选择按键**  
方便用户从 TF 卡启动
5. **用户按键**  
4 个用户按键
6. **蜂鸣器**

7. **电源 LED**
8. **用户 LED**  
4 个用户 LED
9. **电位器**  
AD 可调电位器
10. **32.768KHZ 晶振**  
PCF8563 使用晶振
11. **PCF8563**  
RTC 实时时钟芯片
12. **RTC 后备电池座**  
可接入 3.3V 电池

##### [跳线说明]

13. **RTC 电源选择跳线**
14. **RTC 的 I2C 选择跳线**  
可以选择接入 I2C1 或者 I2C2

### 3.2 LED 测试

在终端输入：

```
root@beaglebone:~# test_led
```

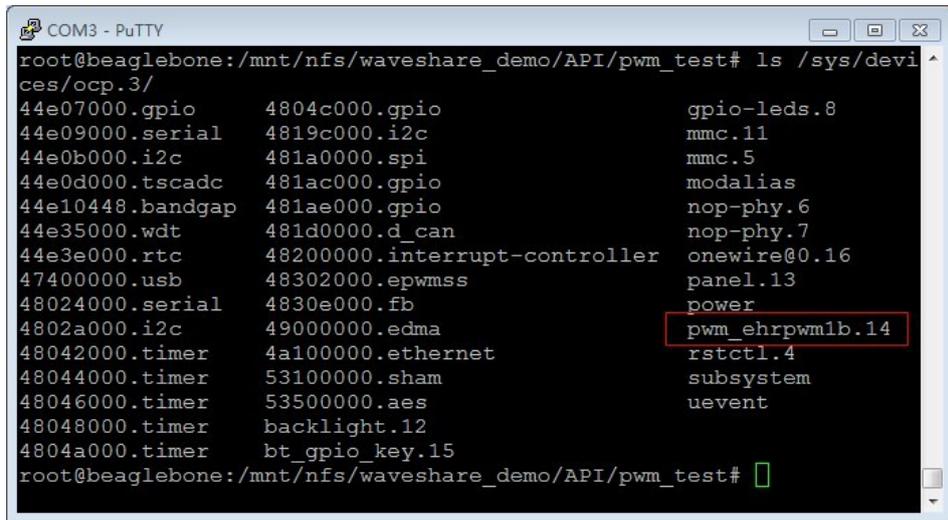
可以看到 4 个 LED 作循环的流动，按键盘 Ctrl+C 结束实验。

### 3.3 蜂鸣器测试

1) 在终端输入：

```
root@beaglebone:~# ls /sys/devices/ocp.3/
```

可以看到 pwm\_ehrpwm1b.14 文件，如下图所示：



```
COM3 - PuTTY
root@beaglebone:/mnt/nfs/waveshare_demo/API/pwm_test# ls /sys/devices/ocp.3/
44e07000.gpio      4804c000.gpio      gpio-leds.8
44e09000.serial   4819c000.i2c       mmc.11
44e0b000.i2c      481a0000.spi       mmc.5
44e0d000.tscadc   481ac000.gpio      modalias
44e10448.bandgap  481ae000.gpio      nop-phy.6
44e35000.wdt      481d0000.d_can     nop-phy.7
44e3e000.rtc      48200000.interrupt-controller onewire@0.16
47400000.usb      48302000.epwmss    panel.13
48024000.serial   4830e000.fb        power
4802a000.i2c      49000000.edma      pwm_ehrpwm1b.14
48042000.timer    4a100000.ethernet  rstctl.4
48044000.timer    53100000.sham      subsystem
48046000.timer    53500000.aes       uevent
48048000.timer    backlight.12
4804a000.timer    bt_gpio_key.15
```

图 9. 查看到 pwm\_ehrpwm1b.14 文件

2) 由于 pwm\_ehrpwm1b.14 的扩展名是.14,在执行蜂鸣器测试命令的时候，需要加入参数 14:

```
root@beaglebone:~# test_pwm 14
```

蜂鸣器会发出不同频率的响声。

### 3.4 DS18B20 测试

1) 插上 DS18B20 到 ONE-WIRE 接口，在终端上输入：

```
root@beaglebone:~#ls /sys/bus/w1/devices/
```

可以看到 28-00000 57c5948(每个 DS18B20 的后 7 位数不同,以你自己的为准,比如我这里的后 7 位是 57c5948,记下这串数字)

2) 在终端上输入：

```
root@beaglebone:~#test_ds18b20 57c5948
```

其中 57c5948 这串数字需要更改为之前记下那一串数字。

终端上会把当前环境的温度打印出来。

### 3.5 按键测试

- 1) 在终端上输入：

```
root@beaglebone:~# test_key event2
```

说明：

设备文件“event2”是根据用户所接入的中断设备有关，不一定是“event2”，可以进行查询，在终端上输入以下命令查看：

```
root@beaglebone:~# ls /dev/input
```

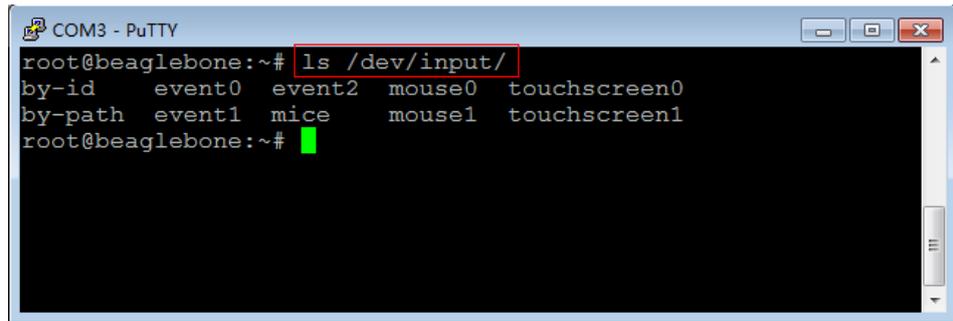


图 10. 查看中断设备

- 2) 分别按下各按键，终端上会显示所按下或松开的按键，按键盘 Ctrl+C 结束实验。

注意：

进行按键测试时，如果外接显示器，屏幕会有被触摸的反应（不清楚是什么样的反映），这是正常现象。

### 3.6 RTC 测试

确保使用纽扣电池（短接帽跳接到 BAT），在终端上输入相应命令：

- 1) 读取系统时间：

```
root@beaglebone:~# date
```

- 2) 设置系统时间：

```
root@beaglebone:~# date 020809302014.23
```

- 3) 设置 RTC 时钟模块的硬件时间：

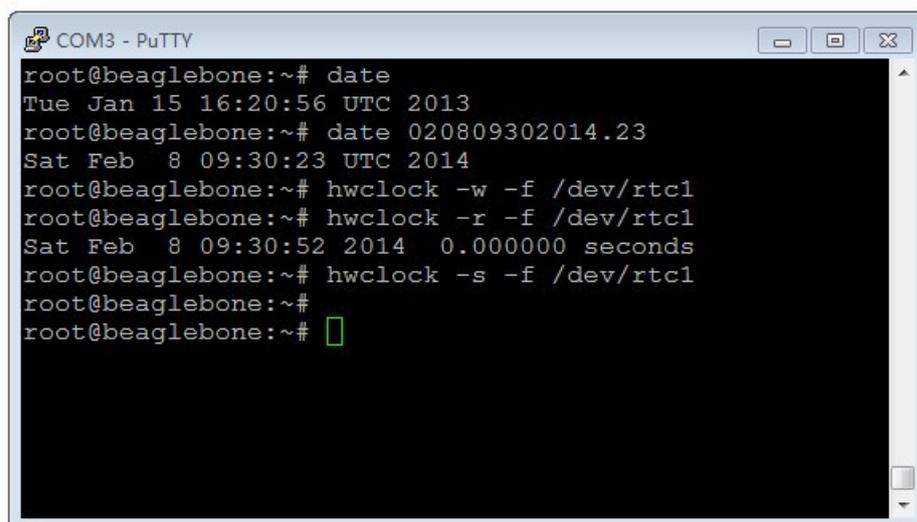
```
root@beaglebone:~# hwclock -w -f /dev/rtc1
```

- 4) 读取 RTC 时钟模块的硬件时间：

```
root@beaglebone:~# hwclock -r -f /dev/rtc1
```

- 5) RTC 时钟模块的硬件时间同步到系统时间：

```
root@beaglebone:~# hwclock -s -f /dev/rtc1
```

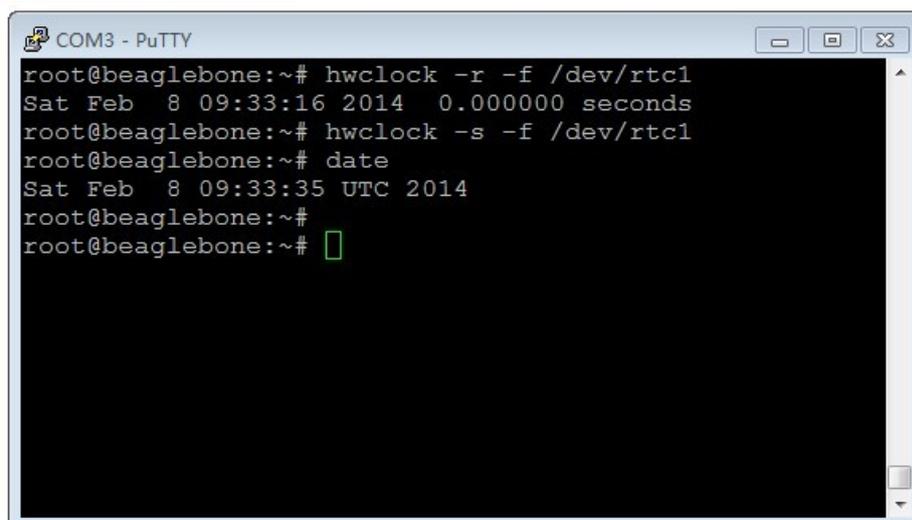
A terminal window titled 'COM3 - PuTTY' showing the process of synchronizing the RTC hardware time to the system time. The user enters 'date' and sees 'Tue Jan 15 16:20:56 UTC 2013'. Then they enter 'date 020809302014.23' and see 'Sat Feb 8 09:30:23 UTC 2014'. Next, they enter 'hwclock -w -f /dev/rtc1', then 'hwclock -r -f /dev/rtc1' which outputs 'Sat Feb 8 09:30:52 2014 0.000000 seconds'. Finally, they enter 'hwclock -s -f /dev/rtc1' and the prompt returns.

```
root@beaglebone:~# date
Tue Jan 15 16:20:56 UTC 2013
root@beaglebone:~# date 020809302014.23
Sat Feb 8 09:30:23 UTC 2014
root@beaglebone:~# hwclock -w -f /dev/rtc1
root@beaglebone:~# hwclock -r -f /dev/rtc1
Sat Feb 8 09:30:52 2014 0.000000 seconds
root@beaglebone:~# hwclock -s -f /dev/rtc1
root@beaglebone:~#
root@beaglebone:~#
```

图 11. RTC 时钟模块的硬件时间同步到系统时间

- 6) 断电重启，读取 RTC 的硬件时间，并同步到系统时间，输入命令：

```
root@beaglebone:~# hwclock -r -f /dev/rtc1
root@beaglebone:~# hwclock -s -f /dev/rtc1
root@beaglebone:~# date
```

A terminal window titled 'COM3 - PuTTY' showing the normal operation of the RTC. The user enters 'hwclock -r -f /dev/rtc1' and sees 'Sat Feb 8 09:33:16 2014 0.000000 seconds'. Then they enter 'hwclock -s -f /dev/rtc1', followed by 'date' which outputs 'Sat Feb 8 09:33:35 UTC 2014'. The prompt returns.

```
root@beaglebone:~# hwclock -r -f /dev/rtc1
Sat Feb 8 09:33:16 2014 0.000000 seconds
root@beaglebone:~# hwclock -s -f /dev/rtc1
root@beaglebone:~# date
Sat Feb 8 09:33:35 UTC 2014
root@beaglebone:~#
root@beaglebone:~#
```

图 12. RTC 正常运界面

此时，软、硬件时间已经同步，RTC 正常工作。

## 4 RS485 CAN CAPE

### 4.1 CAN 测试

BB Black 可以作为一个 CAN 设备使用，测试时连接两块 RS485/CAN CAPE 扩展板，跳线帽选择 UART1 (RXD1、TXD1)，并分别连接 CAN 模块的 H、L 到另一模块的 H、L 端。用户也可以使用自己的 CAN 设备进行测试。

打开两个终端，分别输入相应命令：

- 1) 设置波特率：

```
root@beaglebone:~# canconfig can0 bitrate 115200 ctrlmode triple-sampling on
```

- 2) 使能 CAN 设备：

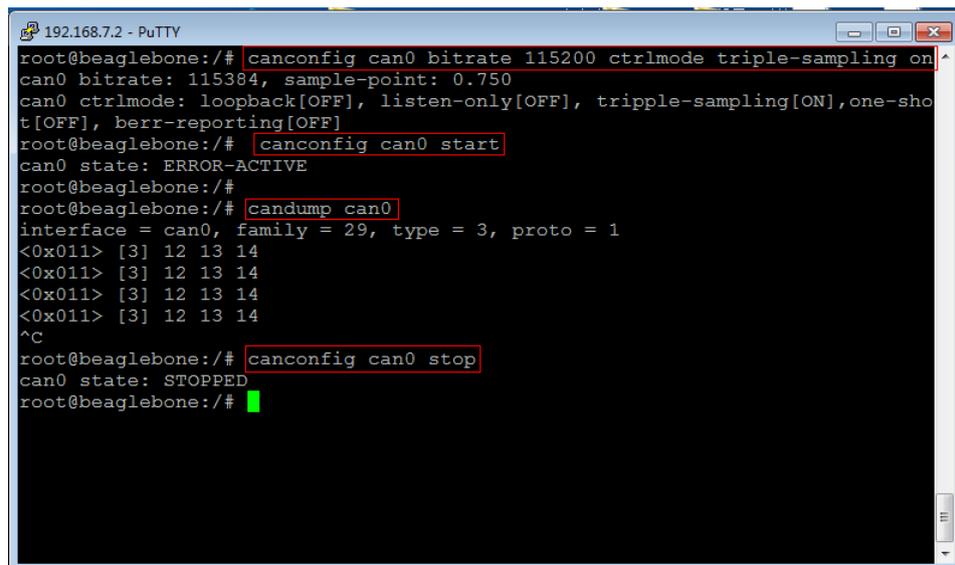
```
root@beaglebone:~# canconfig can0 start
```

两个 CAN 设备分别作为接收端和发送端，接收端应先处于接收状态，然后发送端再发送。

- 3) 接收端：

```
root@beaglebone:~# candump can0
```

接收端接收的结果如下：



```
192.168.7.2 - PuTTY
root@beaglebone:~# canconfig can0 bitrate 115200 ctrlmode triple-sampling on
can0 bitrate: 115384, sample-point: 0.750
can0 ctrlmode: loopback[OFF], listen-only[OFF], tripple-sampling[ON], one-shot[OFF], berr-reporting[OFF]
root@beaglebone:~# canconfig can0 start
can0 state: ERROR-ACTIVE
root@beaglebone:~# candump can0
interface = can0, family = 29, type = 3, proto = 1
<0x011> [3] 12 13 14
^C
root@beaglebone:~# canconfig can0 stop
can0 state: STOPPED
root@beaglebone:~#
```

图 13. CAN 接收端状态

- 4) 发送端：

```
root@beaglebone:~# cansend can0 -i 0x11 0x12 0x13 0x14
```

发送端发送的结果如下：

```

COM3 - PuTTY
root@beaglebone:~# canconfig can0 bitrate 115200 ctrlmode triple-sampling on
can0 bitrate: 115384, sample-point: 0.750
can0 ctrlmode: loopback[OFF], listen-only[OFF], tripple-sampling[ON], one-shot[OFF], berr-reporting[OFF]
root@beaglebone:~# canconfig can0 start
can0 state: ERROR-ACTIVE
root@beaglebone:~# cansend can0 -i 0x11 0x12 0x13 0x14
interface = can0, family = 29, type = 3, proto = 1
root@beaglebone:~#
root@beaglebone:~# cansend can0 -i 0x11 0x12 0x13 0x14
interface = can0, family = 29, type = 3, proto = 1
root@beaglebone:~# cansend can0 -i 0x11 0x12 0x13 0x14
interface = can0, family = 29, type = 3, proto = 1
root@beaglebone:~# cansend can0 -i 0x11 0x12 0x13 0x14
interface = can0, family = 29, type = 3, proto = 1
root@beaglebone:~# canconfig can0 stop
can0 state: STOPPED
root@beaglebone:~# [ 808.506587] tilcdc 4830e000.fb: timeout waiting for fr
amedone

root@beaglebone:~#
root@beaglebone:~# █

```

图 14. CAN 发送端状态

- 5) 关闭设备:

```
root@beaglebone:~# canconfig can0 stop
```

## 4.2 RS485 测试

BB Black 可以作为一个 RS485 设备使用，测试时连接两块 RS485/CAN CAPE 扩展板，跳线帽选择 UART2 (RXD2, TXD2)，并分别连接 RS485 模块的 A、B 端到另一模块的 A、B 端。用户也可以使用自己的 RS485 设备进行测试的。打开两个终端，两个 RS485 设备分别作为接收端和发送端。

- 1) 使接收端应先处于接收状态，输入命令：

```
root@beaglebone:~# test_485 -d /dev/ttyO2 -b 115200
```

出现 3 个选项，终端中输入数字选择即可，接收端选择“2”。在选择“3”停止接收前，接收端一直处于接收状态，如下所示：

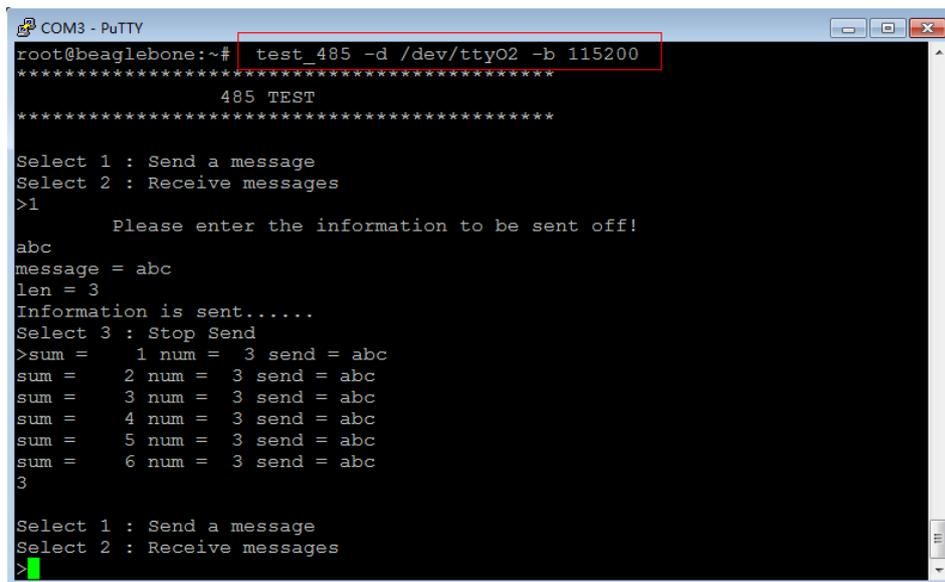
```

192.168.7.2 - PuTTY
root@beaglebone:~# ls
Desktop
root@beaglebone:~# test_485 -d /dev/ttyO2 -b 115200
*****
485 TEST
*****
Select 1 : Send a message
Select 2 : Receive messages
>2
Select 3 : Stop Receive
>
sum = 1 num = 3 recv = abc
sum = 2 num = 3 recv = abc
sum = 3 num = 3 recv = abc
sum = 4 num = 3 recv = abc
sum = 5 num = 3 recv = abc
sum = 6 num = 3 recv = abc
3
Select 1 : Send a message
Select 2 : Receive messages
>

```

图 15. RS485 接收端状态

- 2) 发送端选择“1”，输入要发送的信息，如“abc”，在选择“3”停止发送之前发送端一直处于循环发送状态，发送端可以一直发送数据，如下所示：



```
COM3 - PuTTY
root@beaglebone:~# test_485 -d /dev/ttyO2 -b 115200
*****
485 TEST
*****

Select 1 : Send a message
Select 2 : Receive messages
>1
    Please enter the information to be sent off!
abc
message = abc
len = 3
Information is sent.....
Select 3 : Stop Send
>sum = 1 num = 3 send = abc
sum = 2 num = 3 send = abc
sum = 3 num = 3 send = abc
sum = 4 num = 3 send = abc
sum = 5 num = 3 send = abc
sum = 6 num = 3 send = abc
3

Select 1 : Send a message
Select 2 : Receive messages
>
```

图 16. RS485 发送端状态

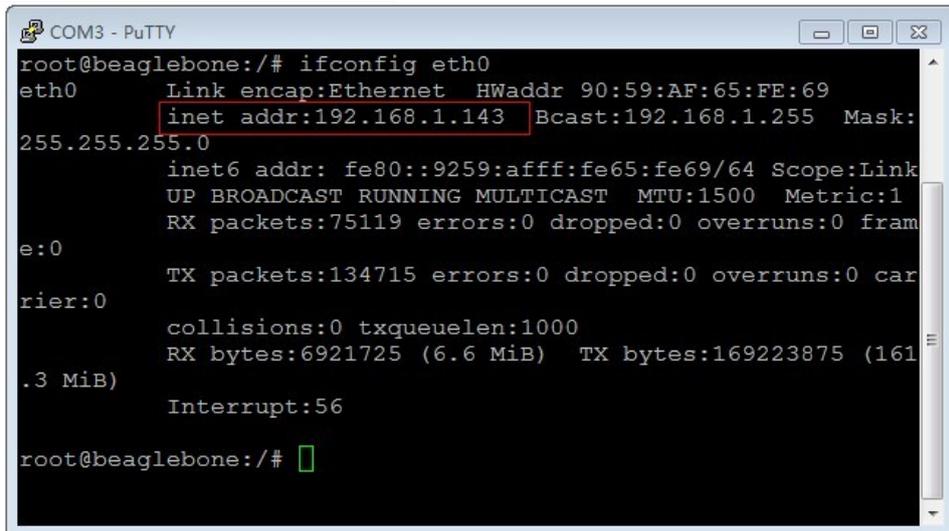
- 3) 按下 Ctrl+C 结束实验。

## 5 其他套件

### 5.1 USB Camera 测试

- 1) 将标配的 USB Camera 插入 BB Black 的 USB Host 接口，并插入网线。
- 2) 终端输入以下命令，查看分配到的 ip 地址：

```
root@beaglebone:~# ifconfig eth0
```



```
COM3 - PuTTY
root@beaglebone:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 90:59:AF:65:FE:69
          inet addr:192.168.1.143  Bcast:192.168.1.255  Mask:
          255.255.255.0
          inet6 addr: fe80::9259:afff:fe65:fe69/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:75119 errors:0 dropped:0 overruns:0 fram
          e:0
          TX packets:134715 errors:0 dropped:0 overruns:0 car
          rier:0
          collisions:0 txqueuelen:1000
          RX bytes:6921725 (6.6 MiB)  TX bytes:169223875 (161
          .3 MiB)
          Interrupt:56

root@beaglebone:~#
```

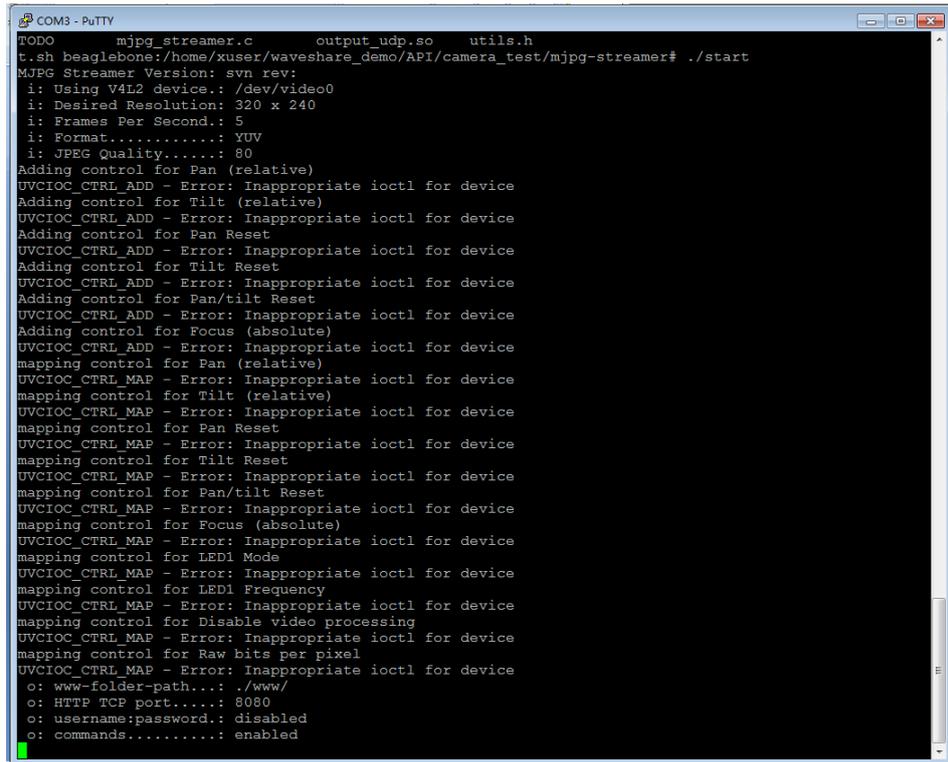
图 17. 查看分配到的地址

可以看到 ip 地址为 192.168.1.143，记下这个 ip。（这个 ip 不是固定的，用户应根据查看的 ip 进行之后的操作）。

- 3) 启动视频流服务器：

```
root@beaglebone:~# cd/home/xuser/waveshare_demo/API/camera_test/mjpg-streamer
```

```
root@beaglebone:~# ./start.sh
```



```
COM3 - PuTTY
TODO      mjpg_streamer.c      output_udp.so      utils.h
t.sh beaglebone:/home/xuser/waveshare_demo/API/camera_test/mjpg-streamer# ./start
MJPG Streamer Version: svn rev:
i: Using V4L2 device.: /dev/video0
i: Desired Resolution: 320 x 240
i: Frames Per Second.: 5
i: Format.....: YUV
i: JPEG Quality.....: 80
Adding control for Pan (relative)
UVCIOC_CTRL_ADD - Error: Inappropriate ioctl for device
Adding control for Tilt (relative)
UVCIOC_CTRL_ADD - Error: Inappropriate ioctl for device
Adding control for Pan Reset
UVCIOC_CTRL_ADD - Error: Inappropriate ioctl for device
Adding control for Tilt Reset
UVCIOC_CTRL_ADD - Error: Inappropriate ioctl for device
Adding control for Pan/tilt Reset
UVCIOC_CTRL_ADD - Error: Inappropriate ioctl for device
Adding control for Focus (absolute)
UVCIOC_CTRL_ADD - Error: Inappropriate ioctl for device
mapping control for Pan (relative)
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for Tilt (relative)
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for Pan Reset
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for Tilt Reset
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for Pan/tilt Reset
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for Focus (absolute)
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for LED1 Mode
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for LED1 Frequency
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for Disable video processing
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
mapping control for Raw bits per pixel
UVCIOC_CTRL_MAP - Error: Inappropriate ioctl for device
o: www-folder-path...: ./www/
o: HTTP TCP port.....: 8080
o: username:password.: disabled
o: commands.....: enabled
```

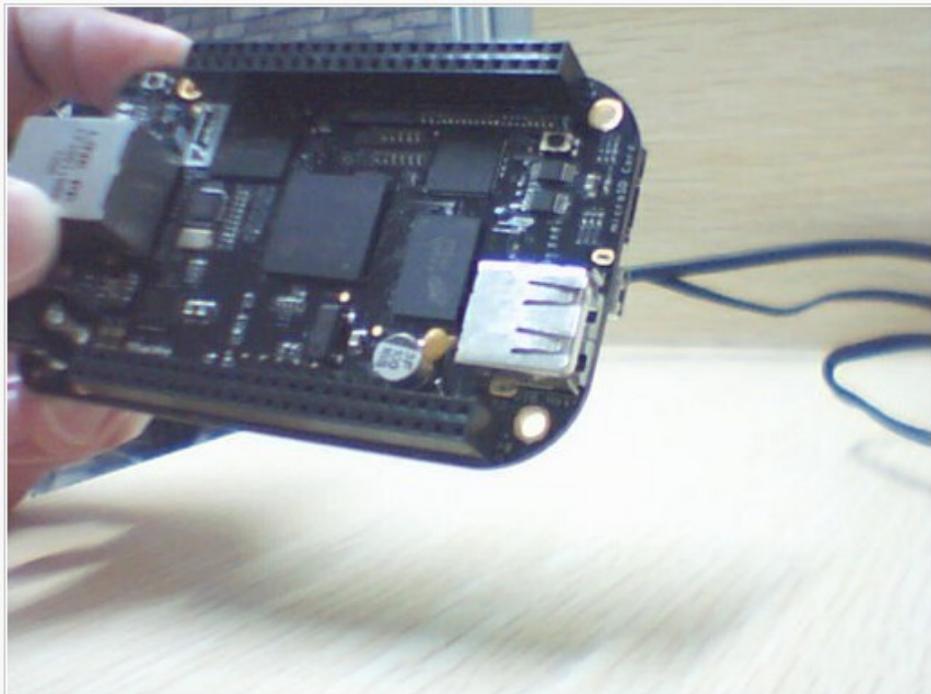
图 18. 启动视频流服务器

- 4) 在一台连接同一子网的电脑上打开浏览器，输入地址：

<http://192.168.1.143:8080/javascript.html>

（这个地址根据之前记下的 ip 地址确定）

即可看到上传的视频流。（默认端口号为 8080）。



© The MJPG-streamer team | Design by Andreas Viklund

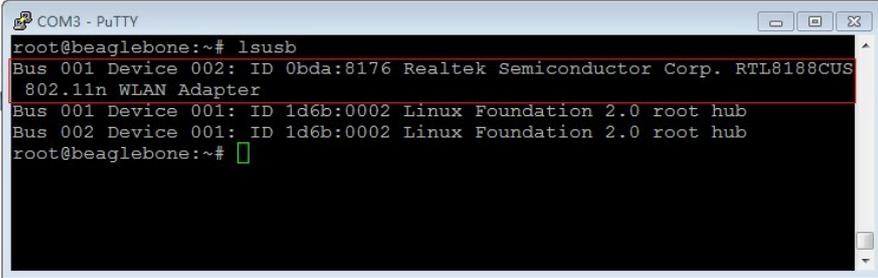
图 19. 查看上传的视频流

- 5) 按下 Ctrl+C 结束实验。

## 5.2 USB WIFI 测试

- 1) 配置无线网络:
- 系统断电，将标配的 USB WIFI 模块插入 BB Black 的 USB Host 接口，系统重新上电。
  - 查看 USB 状态:

```
root@beaglebone:~# lsusb
```

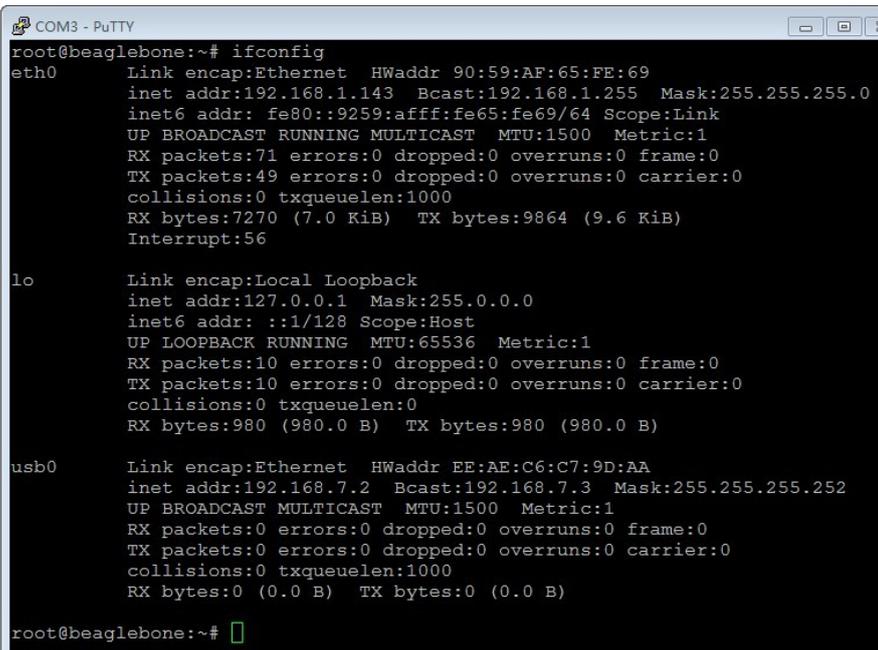


```
Bus 001 Device 002: ID 0bda:8176 Realtek Semiconductor Corp. RTL8188CUS
802.11n WLAN Adapter
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
root@beaglebone:~#
```

图 20. 查看 USB 状态

- 查看以太网网络状态:

```
root@beaglebone:~# ifconfig
```



```
eth0      Link encap:Ethernet  HWaddr 90:59:AF:65:FE:69
          inet addr:192.168.1.143  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::9259:aff:fe65:fe69/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:71 errors:0 dropped:0 overruns:0 frame:0
          TX packets:49 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7270 (7.0 KiB)  TX bytes:9864 (9.6 KiB)
          Interrupt:56

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:65536  Metric:1
          RX packets:10 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:980 (980.0 B)  TX bytes:980 (980.0 B)

usb0      Link encap:Ethernet  HWaddr EE:AE:C6:C7:9D:AA
          inet addr:192.168.7.2  Bcast:192.168.7.3  Mask:255.255.255.252
          UP BROADCAST 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)

root@beaglebone:~#
```

图 21. 查看以太网网络状态

- 关掉以太网卡，打开 WIFI 网卡

```
root@beaglebone:~# ifconfig eth0 down
```

```
root@beaglebone:~# ifconfig wlan0 up
```

- 查看 WI-FI 网络状态:

```
root@beaglebone:~# ifconfig
```

```

COM3 - PuTTY
root@beaglebone:~# ifconfig eth0 down
root@beaglebone:~# ifconfig wlan0 up
root@beaglebone:~# ifconfig
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:65536  Metric:1
            RX packets:10 errors:0 dropped:0 overruns:0 frame:0
            TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:980 (980.0 B)  TX bytes:980 (980.0 B)

usb0       Link encap:Ethernet  HWaddr EE:AE:C6:C7:9D:AA
            inet addr:192.168.7.2  Bcast:192.168.7.3  Mask:255.255.255.252
            UP BROADCAST 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)

wlan0      Link encap:Ethernet  HWaddr 44:33:4C:76:F3:BD
            UP BROADCAST 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)

root@beaglebone:~#

```

图 22. 查看 WI-FI 网络状态

由于没有连接 AP,可以看到 wlan0 接口的接受和发送数据包计数为 0。

- f) 配置 wlan0 的 IP:

```
root@beaglebone:~# ifconfig wlan0 192.168.2.107
```

- g) 配置网关:

```
root@beaglebone:~# route add default gw 192.168.2.1
```

- h) 配置 DNS:

```
root@beaglebone:~# vi /etc/resolv.conf
```

修改

```
nameserver 127.0.0.1
```

为:

```
nameserver 8.8.8.8
```

```

COM3 - PuTTY
# Generated by Connection Manager
nameserver 8.8.8.8

```

图 23. 配置 WI-FI 网络

保存后退出。

- i) 扫描无线路由:

```
root@beaglebone:~# iwlist wlan0 scan
```

```

COM3 - PuTTY
root@beaglebone:~# iwlist wlan0 scan
wlan0 Scan completed :
Cell 01 - Address: 74:EA:3A:1B:65:E2
Channel:1
Frequency:2.412 GHz (Channel 1)
Quality=70/70 Signal level=-35 dBm
Encryption key:on
ESSID:"MERCURY_814"
Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 6 Mb/s
12 Mb/s; 24 Mb/s; 36 Mb/s
Bit Rates:9 Mb/s; 18 Mb/s; 48 Mb/s; 54 Mb/s
Mode:Master
Extra:tsf=00000004751d0181
Extra: Last beacon: 650ms ago
IE: Unknown: 000B4D4552435552595F383134
IE: Unknown: 010882848B960C183048
IE: Unknown: 030101
IE: Unknown: 2A0100
IE: Unknown: 32041224606C
IE: IEEE 802.11i/WPA2 Version 1
Group Cipher : TKIP
Pairwise Ciphers (2) : TKIP CCMP
Authentication Suites (1) : PSK
Preauthentication Supported
IE: WPA Version 1
Group Cipher : TKIP
Pairwise Ciphers (2) : TKIP CCMP
Authentication Suites (1) : PSK
IE: Unknown: DD0900037F01010008FF7F
IE: Unknown: DD1A00037F030100000074EA3A1B65E276EA3A1B65
E264002C010808
Cell 02 - Address: B4:41:7A:55:05:A0
Channel:1
Frequency:2.412 GHz (Channel 1)
Quality=70/70 Signal level=-35 dBm
Encryption key:on

```

图 24. 扫描无线路由

## 2) 连接无线路由

- a) 配置密钥文件/etc/wpa\_supplicant.conf（这里以连接“WPA/PSK 加密无线路”由为例，假设连接的无线路由 ESSID 为 waveshare，密码 12345678）：

```
root@beaglebone:~# vi /etc/wpa_supplicant.conf
```

修改

```
network={
    key_mgmt=NONE
}
```

为：

```
network={
    ssid="waveshare"
    psk="12345678"
}
```

保存后退出。

- b) 手动连接：

```
root@beaglebone:~# wpa_supplicant -B -i wlan0 -c /etc/wpa_supplicant.conf
```

- c) 网络测试：

```
root@beaglebone:~# ping www.baidu.com
```