

BeagleBone Black Industrial系统更新设置一贴通

前言

原创文章，转载引用务必注明链接。水平有限，欢迎指正。

本文使用markdown写成，为获得更好的阅读体验，推荐访问我的博客原文：

http://www.omoikane.cn/2016/09/15/bbb_debian/

系统环境：Windows 7 Home Pentium x64|Deepin Linux 15.2 x64；Debian On BBB Industrial

预装系统介绍

相信之前用过BeagleBone Black的都记得预装的Ångström Linux系统，[官方主页](#)一度停止更新，opkg软件仓库也不再维护，不过经MOP小弟提醒发现其[github项目主页](#)依然活跃，向这些默默努力的人致敬！但是BeagleBoard官方提供的镜像还停留在2013年6月20日的版本，现在4G版BBB都是使用Debian。我们来看一看官方预装的系统情况，使用一根MicroUSB线（这个接口已经相对不多见了）与电脑连接后，[下载安装驱动](#)后，打开SSH客户端（Kitty或Xshell等），主机地址填192.168.7.2或beaglebone.local，用户名root，密码为空（另一个默认账户是**debian:temp**pwd）。输入以下命令：

```
lsb_release -a
cat /proc/version
uname -a
df -Th
netstat -lnp

root@beaglebone:~# lsb_release -a
No LSB modules are available.
Distributor ID: Debian
Description:    Debian GNU/Linux 7.9 (wheezy)
Release:       7.9
Codename:      wheezy
root@beaglebone:~# cat /proc/version
Linux version 3.8.13-bone79 (root@a4-imx6q-wandboard-2gb) (gcc version 4.6.3 (Debian 4.6.3-14) ) #1
root@beaglebone:~# uname -a
Linux beaglebone 3.8.13-bone79 #1 SMP Tue Oct 13 20:44:55 UTC 2015 armv7l GNU/Linux
```

公告

昵称：思兼
园龄：2年6个月
粉丝：8
关注：1
[+加关注](#)

<	2017年10月						>
日	一	二	三	四	五	六	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	
8	9	10	11	12	13	14	
15	16	17	18	19	20	21	
22	23	24	25	26	27	28	
29	30	31	1	2	3	4	

搜索

找找看

谷歌搜索

常用链接

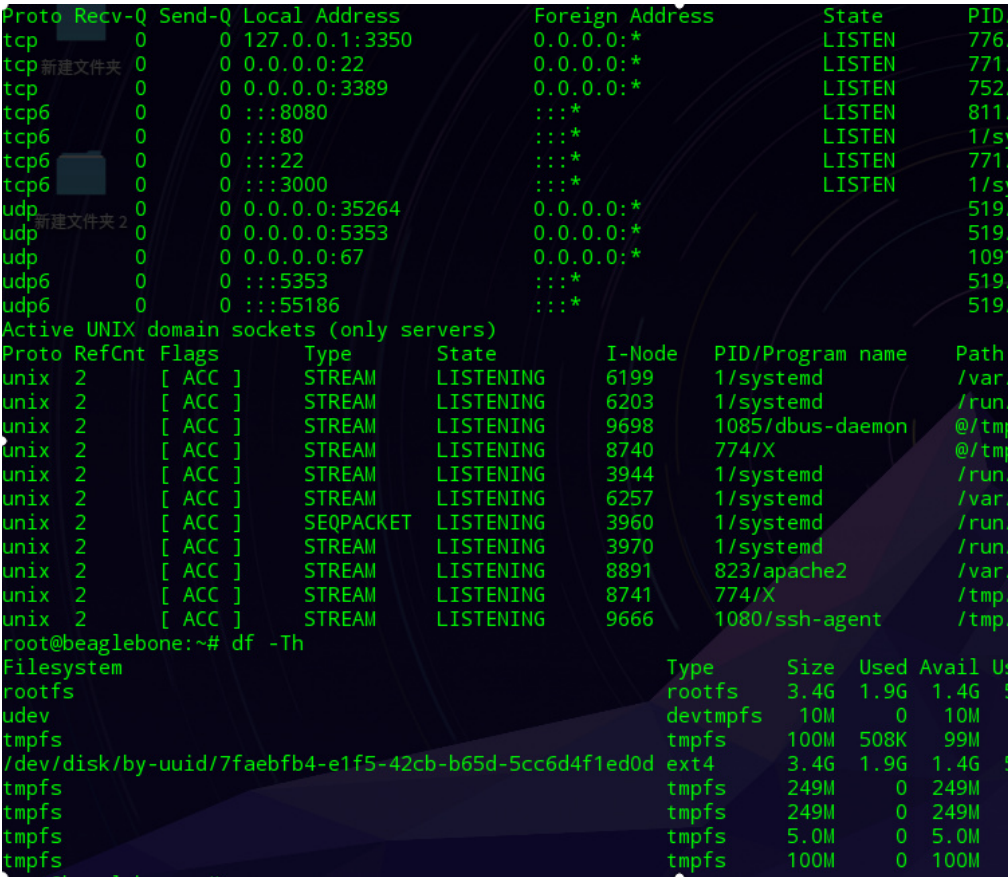
我的随笔
我的评论
我的参与
最新评论
我的标签

我的标签

- Intel Edison(5)
- BeagleBone Black(4)
- UP Board(3)
- Zero(3)
- 树莓派(3)
- Espressobin(2)
- LattePanda(2)
- Lemaker Guitar(2)
- MobaXterm(1)
- SSH(1)
- 更多

随笔分类

- 开发板(16)
- 树莓派2(3)



可以看到当前系统版本为Debian 7.9 Wheezy，在wandboard（飞思卡尔 imx6q四核SoC）上编译，内核版本3.8.13，预装apache2 Web服务器，rdp远程桌面服务器，udhcpd DHCP服务器。

我们打开[官方系统更新页面](#)，可以看到目前最新的是发布于2016年5月13日的Debian 8.5 Jessie MicroSD卡镜像（然而下载链接写的8.4），要求至少4G存储空间（eMMC或者SD），另外官方提供了烧录到板载eMMC存储的方法。

Recommended Debian Images

Jessie for BeagleBone via microSD card

▶ Debian 8.5 2016-05-13 4GB SD LXQT image for BeagleBone, BeagleBone Black, SeeedStudio BeagleBone Green, SeesatCloud BeagleBone Enhanced, element14 BeagleBone Black Industrial, Arrow BeagleBone Black Industrial and Merbmap - sha256sum: 28d67e8774971b9e521e60512c6e10baedanz3e9iasz3e9ed2076ae375aa777f

Jessie for BeagleBoard-X15 via microSD card

▶ Debian 8.5 2016-08-28 4GB SD LXQT image for BeagleBoard-X15 - more info - bmap - sha256sum: e7cae98823f7f9f9abdb76c76e78071586d89af760391770b4d31a56afa2811

Jessie for BeagleBoard-xM via microSD card

▶ Debian 8.3 2016-02-11 2GB SD console image for BeagleBoard-xM - more info

Wheezy for BeagleBone via microSD card

▶ Debian 7.11 2015-06-15 4GB SD LXDE image for BeagleBone, BeagleBone Black, SeeedStudio BeagleBone Green, SeesatCloud BeagleBone Black Industrial and Mentorel BeagleBone uSOMIQ - more info - bmap - sha256sum: 128d67b78e0adb8beefed73e88a3219b4cacdc8929663c2a997627cc6333591e

To turn these images into eMMC flasher images, edit the /boot/uEnv.txt file on the Linux partition on the microSD card to 'cmdline=init=/opt/scripts/tools/eMMC/init-eMMC-flasher-v3.sh'. Enabling this will cause booting the microSD card to no longer be provided here for this to avoid people accidentally overwriting their eMMC flash.

For testing, flasher and other Debian images, see http://elinux.org/Beagleboard:BeagleBoneBlack_Debian and [debiant](#)

FAQ：

- 如何从MicroSD卡启动

BBB默认从板载eMMC启动，按住USB接口旁边的USER/BOOT按钮（S2）然后开机即可从MicroSD启动

- 我不需要图形界面，默认系统占用空间过大

除了官方包含LXQT的Debian镜像，[访问这里](#)可以看到更多选择,其中console版就是仅包含核心功能的镜像，但是没有USB Gadget驱动（无法模拟串口、以太网卡和存储设备），没有预装Apache2，X11，xrdp等。

- 除了Debian还有那些系统可用

随笔档案

- 2017年9月 (6)
- 2017年6月 (1)
- 2017年5月 (2)
- 2017年4月 (3)
- 2017年3月 (1)
- 2017年2月 (1)
- 2017年1月 (20)
- 2016年12月 (5)
- 2015年11月 (1)
- 2015年6月 (1)
- 2015年5月 (1)
- 2015年4月 (9)

最新评论

1. Re: 不从SD卡启动树莓派2
正好需要弄这个，做一个web服务器，要把系统做到hdd里，借鉴下，谢谢分享
--Hardi
2. Re: 树莓派2安装使用小米Wifi (360 小度 腾讯wifi)
@姑苏记不清楚了，gcc、g++版本，还有确定rpi-source正确配置，另外就是有个文件参数改了没...
--思兼
3. Re: 树莓派2安装使用小米Wifi (360 小度 腾讯wifi)
FATAL: parse error in symbol dump filescripts/Makefile.modpost:90: recipe for target '___modpost' fai.....
--姑苏

阅读排行榜

1. 树莓派系统介绍：DIetPi(5191)
2. 忘掉VNC/RDP，拿起手中的MobaXterm轻松上手远程桌面(3555)
3. 树莓派2系统DietPi简单安装配置使用介绍(2636)
4. 树莓派2安装使用小米Wifi (360 小度 腾讯wifi) (2620)
5. 树莓派笔记之使用netselect选择最快Raspbian软件源(2383)

评论排行榜

1. 树莓派2安装使用小米Wifi (360 小度 腾讯wifi) (2)
2. 不从SD卡启动树莓派2(1)

推荐排行榜

1. 使用mosh取代ssh提高n2n网络连接稳定性(1)
2. 树莓派无线网卡选择(1)
3. 不从SD卡启动树莓派2(1)

- [ArchLinux](#)
- [Ostro Project](#)
- [Yocto Project](#)
- [Kali Linux](#)
- [Fedora Linux](#)
- 不一而足，后面预计讲一下BlackArch——安全审计，Ostro——专注物联网，Yocto——为BBB编译镜像等
- 串口调试功能如何使用

[官网wiki介绍见此](#)

- BBB上有没有类似树莓派的RPI.GPIO, wiringPi这种操控GPIO的库

有，[BoneScript Library](#)，另外也能使用upm。

- 通过 USB Gadget 驱动BBB的USB接口实现哪些功能
- 模拟以太网接口：192.168.7.2，**劫持局域网dns解析**后使用beaglebone.local也能访问
- 模拟Mass Storage（MSG）：在宿主机中显示BEAGLEBONE的磁盘
- 模拟串口：可以查看串口输出，但是因为是模拟的，所以开关机信息和uboot无法查看调试
- 有没有比较全的文档wiki

[看这里](#)——elinux.org

- 插上MicroSD卡后重启会从哪启动？

会从上次启动的设备启动

更新板载系统

所需硬件：

- MicroSD卡一张，4G以上大小
- BeagleBone Black Industrial
- MicroUSB数据线
- MicroSD 读卡器（能读MicroSD的设备如手机等也可以）

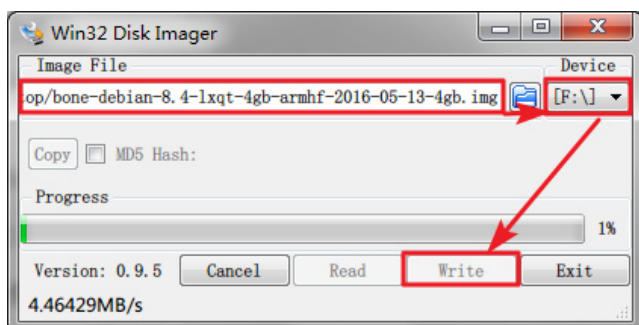
所需软件（以Windows为例）：

- 最新官方Debian系统镜像——[下载地址](#)
- 解压缩软件——7-zip、WinRAR等，用于解压下载来的img.xz压缩文件中的系统镜像
- Win32 Disk Imager —— [下载地址](#)，用于烧录系统文件至MicroSD卡

烧录系统（MicroSD+eMMC）：

插上MicroSD卡，打开Win32 Disk Imager，选择解压出来的

bone-debian-8.4-lxqt-4gb-armhf-2016-05-13-4gb.img 文件，**选择正确的写入设备(MicroSD卡)**，点击Write等待片刻即可。



烧录完成，将MicroSD卡插入BBB卡槽，按住USER/BOOT按钮后开机，即可从SD存储启动。

当然，我们的最终目的是**升级eMMC的板载系统**，具体方法[官网](#)也有讲到：

To turn these images into eMMC flasher images, edit the /boot/uEnv.txt file on the Linux partition on the microSD card and remove the '#' on the line with 'cmdline=init=/opt/scripts/tools/eMMC/init-eMMC-flasher-v3.sh'. Enabling this will cause booting the microSD card to flash the eMMC. Images are no longer provided here for this to avoid people accidentally overwriting their eMMC flash.

烧录好系统到MicroSD卡后，编辑/boot/uEnv.txt文件，取消注释

cmdline=init=/opt/scripts/tools/eMMC/init-eMMC-flasher-v3.sh，从而开启自动烧录到eMMC

的功能脚本。之所以不再提供单独的eMMC镜像是为了防止有些人太不清楚的时候误操作覆盖掉eMMC里面的系统文件（怎么感觉是官方自己这么干过得到的血的教训。。。）

注意！

1. MicroSD卡烧录完系统后，因为是Linux文件系统，Windows下不可见。另外/boot/uEnv.txt文件也无从修改（即使我知道你可以，但是仍然不建议在Windows下修改Linux文件系统文件），这时我们可以先启动BBB Industrial，登录之后插上MicroSD卡，进行修改。如果你不会的话，跟我一起学写shell：

```
lsblk |grep -q mmcblk1 && mkdir /media/sd && mount /dev/mmcblkp1 /media/sd && sed -i
's/#cmdline=init/cmdline=init/g' /media/sd/boot/uEnv.txt && umount /media/sd && poweroff
|| echo please insert ur sdcard properly!
```

别看命令这么长，其实shell编程逻辑比较简单，`|` 表示上一个命令的输出作为下一个命令的输入，

`&&` 表示上一命令执行成功则执行此命令，`||` 表示上一命令失败则执行此命令。

所以上述命令的意思是查看当前存储设备——查找是否有sd卡插入——有则创建/media/sd目录并挂载——修改uEnv.txt文件，取消注释——卸载sd卡，**关机**(`&& poweroff`，方便烧录，可选)——如果没有检测到会提醒你正确插入sd卡。sed命令的用法[看这里](#)。

1. 选择烧录到eMMC时，按住S2开机，随后以太网接口旁边四个LED全亮即可松开，随后进入流水灯模式代表正在烧录
2. 烧录到eMMC的过程可能会持续45分钟，当烧录完毕时，以太网接口旁边的四个灯会全亮（不闪烁），此时关闭电源，拔掉MicroSD卡，重启即会从eMMC新的镜像启动。
3. 想从MicroSD卡启动的话记得重新注释那一行（行首加个`#`）

好的，趁着等待的这会儿，我去吃饭了~

吃完饭继续更新

咦，这和剧本写得不一样啊导演！四个灯全灭而不是全亮，不管了，拔掉MicroSD卡重启看看，当然这时如果有串口调试模块就好了。等了好一会儿，宿主机显示出BEAGLEBONE磁盘，SSH和虚拟串口都可以使用了。我们再看看新系统：

```
root@beaglebone:~# lsb_release -a
No LSB modules are available.
Distributor ID: Debian
Description:   Debian GNU/Linux 8.4 (jessie)
Release:      8.4
Codename:     jessie
root@beaglebone:~# cat /proc/version
Linux version 4.4.9-ti-r25 (root@b3-jetson-tk1-2gb) (gcc version 4.9.2 (Debian 4.9.2-10) ) #1
root@beaglebone:~# uname -a
Linux beaglebone 4.4.9-ti-r25 #1 SMP Thu May 5 23:08:13 UTC 2016 armv7l GNU/Linux
root@beaglebone:~# df -Th
Filesystem      Type      Size      Used Avail Use% Mounted on
udev            devtmpfs  10M        0      10M   0% /dev
tmpfs           tmpfs     99M       8.4M   91M    9% /run
/dev/mmcblk0p1  ext4     3.5G      3.1G   218M   94% /
tmpfs           tmpfs     247M       4.0K   247M    1% /dev/shm
tmpfs           tmpfs     5.0M       4.0K   5.0M    1% /run/lock
tmpfs           tmpfs     247M        0     247M    0% /sys/fs/cgroup
tmpfs           tmpfs     50M        0      50M    0% /run/user/1000
tmpfs           tmpfs     50M        0      50M    0% /run/user/0
```



```
root@beaglebone:~# netstat -lnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/
tcp        0      0 127.0.0.1:53            0.0.0.0:*                LISTEN      761/
tcp        0      0 0.0.0.0:22              0.0.0.0:*                LISTEN      708/
tcp6       0      0 :::8080                 :::*                    LISTEN      1508/
tcp6       0      0 :::80                   :::*                    LISTEN      1/in
tcp6       0      0 :::1:53                 :::*                    LISTEN      761/
tcp6       0      0 :::22                   :::*                    LISTEN      708/
tcp6       0      0 :::3000                 :::*                    LISTEN      1/in
tcp6       0      0 :::1880                 :::*                    LISTEN      1/in
udp        0      0 0.0.0.0:33278           0.0.0.0:*                333/
udp        0      0 127.0.0.1:53            0.0.0.0:*                761/
udp        0      0 0.0.0.0:5353            0.0.0.0:*                333/
udp6       0      0 :::60963               :::*                    333/
udp6       0      0 :::1:53                 :::*                    761/
udp6       0      0 :::5353                 :::*                    333/
Active UNIX domain sockets (only servers)
Proto RefCnt Flags               Type               State              I-Node  PID/Program name  Path
unix   2      [ ACC ]                STREAM             LISTENING          32005   1245/systemd      /run/
unix   2      [ ACC ]                STREAM             LISTENING          15925   424/X              @/tmp
unix   2      [ ACC ]                STREAM             LISTENING          12855   1/init             /run/
unix   2      [ ACC ]                STREAM             LISTENING          16509   496/dbus-daemon    @/tmp
unix   2      [ ACC ]                SEQPACKET          LISTENING          12877   1/init             /run/
unix   2      [ ACC ]                STREAM             LISTENING          12880   1/init             /run/
unix   2      [ ACC ]                STREAM             LISTENING          17002   611/gvfsd-trash    @/dbu
unix   2      [ ACC ]                STREAM             LISTENING          15926   424/X              /tmp/
unix   2      [ ACC ]                STREAM             LISTENING          16493   488/ssh-agent      /tmp/
```

可以看到，更新后的系统是Debian Jessie 8.4（官网骗人啊），使用英伟达（Nvidia，造显卡那个）jetson tk1开发板编译，内核版本4.4.9，默认没有安装rdp远程桌面服务。**可用空间218M什么鬼。。。**

3000:Cloud9

1880:Node-RED

更多信息可以使用以下命令：

```
cat /proc/cpuinfo # 查看CPU信息
dmesg | grep scheduler # 支持的系统I/O调度
cat /sys/block/mmcblk0/queue/scheduler # 当前所使用的I/O调度算法
cat /proc/driver/rtc # 板载RTC信息
cat /sys/power/state # 支持的电源管理模式
```

系统初始设置

更改时区

```
timedatectl set-timezone Asia/Shanghai
```

更改时间

```
timedatectl set-time "yyyy-MM-dd hh:mm:ss"
```

注意年月日之间使用短横杠 连接。然而运行改命令会提示 ，由于设置了自动更新时间无法手动更改，后面也发现连上网重启之后会获得正确的时间。

关于硬件时钟和系统时钟可以看我之前写的文章[《吃我一记咸鱼突刺——使用板载RTC定时开机》](#)

使用无线网络

BBB可以使用有线网或无线网，我这里掏出之前买的Edimax无线网卡，使用的是经典RT3070芯片。

通过以下命令可以看到无线网卡已经被识别：

```
lsusb
iwconfig
lsmod
iw list # 该命令可以看到无线网卡支持的模式，比如monitor, AP, Station等，后续会讲
```

```

root@beaglebone:~# lsmod |grep rt2*
rt2800usb          16465  0
rt2800lib          63194  1 rt2800usb
rt2x00usb          10547  1 rt2800usb
rt2x00lib          40316  3 rt2x00usb,rt2800lib,rt2800usb
mac80211           486884  3 rt2x00lib,rt2x00usb,rt2800lib
cfg80211           418909  2 mac80211,rt2x00lib
crc_ccitt           1752  1 rt2800lib
root@beaglebone:~# lsusb
Bus 001 Device 002: ID 7392:7711 Edimax Technology Co., Ltd EW-7711UTn nLite Wire
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
root@beaglebone:~# iwconfig
tether             no wireless extensions.

wlan0              IEEE 802.11bgn Mode:Master Tx-Power=20 dBm
                    Retry short limit:7   RTS thr:off   Fragment thr:off
                    Power Management:off

```

一般修改 `/etc/network/interfaces` 文件（低版本内核修改此文件仅支持wep加密）、`wireless_tools`或者使用`wpa_supplicant`可以连接网络。然而`interfaces`文件里有这么一句：

```
# WiFi use: -> connmanctl
```

，正好 `netstat -lnp` 里面也有`connmanctl`的身影，看名字应该是个网络连接管理软件。

connmanctl是啥

`connmanctl --help` 看了一下使用介绍，发现功能挺多啊。

ConnMan是一个专为嵌入式设备和快速解析设计的命令行网络管理程序，支持诸多模块，而且内置dhcp（IP地址分配）和ntp（网络授时）功能。

使用方法可以看这些：

- [Connmanctl Cheat Sheet](#)——看这个就会用了
- [Arch Wiki Connman 页面](#)
- [Connman 文档](#)

简单来说，[technologies]理解为网卡interfaces，[services]理解为可连接的网络：

```

connmanctl
> enable wifi # 启用无线网卡
> scan wifi # 扫描无线网络
> services # 列举扫描到的无线网络
> services wifi_*_psk # [optional] 查看某个无线网络的具体信息
> agent on # [optional] 无密码的可以不需要这句
> connect wifi_*_psk # 不是SSID，二是后面对应的wifi_*，根据自己情况修改此时会要求你输入网络密码
> state # 成功连接状态为ready，已连上网是online
> help # 显示帮助文件
> exit # 退出

```

成功连接后，会在 `/var/lib/connman` 目录下生成配置文件，而且下次开机就会自动连接了。

```

root@beaglebone:~# connmanctl
connmanctl> services
*AO FAST_4FF4 wifi_8cbebe00914d_464153545f34464634_managed_psk
      BOOM    wifi_8cbebe00914d_424f4f4d_managed_psk
      TP-LINK_ECA4 wifi_8cbebe00914d_54502d4c494e4b5f45434134_managed_psk
      xiao      wifi_8cbebe00914d_7869616f_managed_psk
      FAST_5308  wifi_8cbebe00914d_464153545f35333038_managed_psk

```

【无线网络扫描结果】*——favorite；A——autoconnect；O——online。connect后面的参数是wifi开头的而不是你的无线网名称。managed代表AP模式（非Ad-Hoc），psk为加密方式。开放网络是none，此时不需要agent on。

注意：

- 随便掏出来的Edimax估计坏了，先后出现过各种奇怪问题，包括：
- connect时显示

```
Error /net/connman/service/[the ssid]: Method "Connect" with signature "" on interface "net.connman.Service" doesn't exist
```

——更新：需要关闭tether模式才行

- 连接成功后只能获取到254.169这种无效IP地址
- `ifconfig` 手动为wlan0设置IP地址仍然不能联网
- 于是我又掏出来一个小米随身wifi (使用mt7601u芯片, 包括360、小度等都使用该芯片), 还好新版Linux 内核已经包含该芯片驱动, 可以直接使用。具体可以参考我之前写的关于[USB无线网卡选择的文章](#), 这回正常了, 很顺利。
- `connmanctl` 还有一个tether命令, 需要无线网卡支持ap模式, 可惜mt7601u不支持, 仅支持station, monitor模式, rt3070倒是支持, `connmanctl tether wifi on` 之后可以搜索到一个名为BeagleBone-146C的无线网。更新: tether和station模式不能同时打开。
- 更多用法可以使用 `connmanctl help` 或者查看上面所说的文档

网线直连共享上网

这种主要针对没有路由器, 没有开箱即用的无线网卡, 手头只有一根网线的情况。

新入手一块板子, 没有路由器, 没有显示器, 也没用串口调试模块, 手头仅有一根网线, 眼前一抹黑, 怎么办?

继续参考我之前写的文章[《网线直连控制Guitar \(增强版\)》](#), 你可以学到[直连线和交叉线](#); **Windows上搭建DHCP服务器, tftp协议**相关的知识。

这样BBB Industrial也可以通过宿主机关上网更新了。当然如果你有路由器直接插上网线就可以, 多简单。

更新系统 (内核、软件)

更换软件源为tsinghua (清华大学), 也可以使用ustc (中国科学技术大学, 但是最近不稳定), 常用的阿里源不包括Debian armhf架构就不列出来了。

```
sed -i 's/httpredir.debian.org/mirrors4.tuna.tsinghua.edu.cn/g' /etc/apt/sources.list
```

是不是发现sed命令很好用!

```
# 更新系统
apt-get update && apt-get upgrade
```

推荐你去看看 `/etc/apt/sources.list` 文件, 值得一看。因为升级时速度过慢, 可以改upm, node.js的更新地址, 或者使用代理 (connman就支持设置代理)。此外内核的更新没有Raspberry Pi使用rpi-update那么方便了。

再次运行 `lsb_release -a` 命令可以看到已经升级为最新的Debian Jessie 8.5了。另外不推荐贸然执行 `apt-get dist-upgrade` 升级系统, 原因我也讲过, upgrade遇到依赖冲突时会保持原样, dist-upgrade会强行升级依赖而往往无法做到满足所有依赖, 例如软件A需要库1.1版本, 软件B需要库2.4版本, 后者会升级为库2.4并导致软件A无法使用。



分类: 开发板

标签: BeagleBone Black

好文要顶

关注我

收藏该文



思兼

关注 - 1

粉丝 - 8

±加关注

0

0

« 上一篇: [BeagleBone Black Industrial 工业版介绍](#)

» 下一篇: [忘掉VNC/RDP, 拿起手中的MobaXterm轻松上手远程桌面](#)

posted @ 2017-01-19 11:52 思兼 阅读(608) 评论(0) 编辑 收藏

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

【推荐】搭建微信小程序 就选腾讯云

【推荐】报表开发有捷径：快速设计轻松集成，数据可视化和交互



最新知识库文章：

- 实用VPC虚拟私有云设计原则
 - 如何阅读计算机科学类的书
 - Google 及其云智慧
 - 做到这一点，你也可以成为优秀的程序员
 - 写给立志做码农的大学生
- » [更多知识库文章...](#)

Copyright ©2017 思兼