

**TOPEET 迅为**

[www.topeetboard.com](http://www.topeetboard.com)

## 「新一代 AIOT 设备平台的优秀典范」

---

超长供货周期      7 X 24 小时稳定运行  
软硬件全开源      丰富接口和高扩展性

**iTOP-RK3568**

迅为电子团队编著



官方微信公众号



迅为商城

 010-8527-0708

 [www.topeetboard.com](http://www.topeetboard.com)

 Beijing TOPEET Electronics Co.Ltd

## 更新记录

更新版本	修改内容
V1.0	初版

## 目录

注意事项与售后维修 .....	2
技术支持与开发定制 .....	6
资料获取与后续更新 .....	8
更新记录 .....	2
目录 .....	3
第 1 章 TFTP&NFS 网络烧写 .....	9
1.1 安装 TFTP 服务端 .....	9
1.2 安装 NFS 服务 .....	10
1.3 配置开发板网络 .....	12
1.4 tftp 加载内核和设备树 .....	14
1.5 nfs 挂载文件系统 .....	15
1.5.1 获取文件系统镜像 .....	15
1.5.2 网络挂载 .....	17

## 注意事项与售后维修

### 一 注意事项

- ◆ 请注意和遵循标注在产品上的所有警示和指引信息；
- ◆ 请勿带电插拔核心板及外围模块；
- ◆ 使用产品之前，请仔细阅读本手册，并妥善保管，以备将来参考；
- ◆ 请使用配套电源适配器，以保证电压、电流的稳定；
- ◆ 请在凉爽、干燥、清洁的地方使用本产品；
- ◆ 请勿在冷热交替环境中使用本产品，避免结露损坏元器件；
- ◆ 请勿将任何液体泼溅在本产品上，如果不慎被任何液体泼溅或浸润，请立刻断电并充分晾干；
- ◆ 请勿使用有机溶剂或腐蚀性液体清洗本产品；
- ◆ 请勿在多尘、脏乱的环境中使用本产品，如果长期不使用，请包装好本产品；
- ◆ 请勿在震动过大的环境中使用，任何跌落、敲打或剧烈晃动都可能损坏线路及元器件；
- ◆ 请勿在通电情况下，插拔核心板及外围模块(特别是串口模块)；
- ◆ 请勿自行维修、拆解本产品，如产品出现故障应及时联系本公司进行维修；
- ◆ 请勿自行修改或使用未经授权的配件，由此造成的损坏将不予保修；

### 二 售后维修

凡是通过迅为直接购买或经迅为授权的正规代理商处购买的迅为产品，均可享受以下权益：

- 1、开发板本身 1 年免费保修服务（配件除外）；
- 2、保修期满后出现产品异常，迅为提供有偿维修服务，可与迅为取得联系，收费视具体情况而定。如遇损坏程度严重等其他不可控因素导致无法维修的，公司不再提供维修服务；
- 3、如您购买的产品需要维修或检测，请提前备份机器内的相关数据。迅为不对因数据丢失所造成的损失负责。

注：以下情况不属于免费维修范围，可提供有偿维修：

- 1、超出保修期的产品；
- 2、非保元件：CPU、内存芯片、Flash；
- 3、由于使用不当，出现诸如 PCB 烧毁、破裂等物理损伤的产品；

- 4、由于人为疏忽或错误使用、未按说明书规定使用而造成的产品损坏等；
- 5、拆装或更换组件、器件而造成无法复原的开发板；
- 6、在将故障件返回迅为技术服务部的过程中由于包装或运输操作不当造成损坏的产品。

**维修周期：**收到返修产品后，我们将即日安排工程师进行检测，我们将在短时间内维修或更换并寄回。一般的故障维修周期为 5 个工作日（自我司收到物品之日起，不计运输过程时间），由于特殊故障导致无法短期内维修的产品，我们会与用户另行沟通并确认维修周期。

**维修费用：**在免费保修期内的产品，由于产品质量问题引起的故障，不收任何维修费用；不属于免费保修范围内的故障或损坏，在检测确认问题后，我们将与客户沟通并确认维修费用，我们仅收取元器件材料费，不收取维修服务费；超过保修期限的产品，根据实际损坏的程度来确定收取的元器件材料费和维修服务费。

**运输费用：**产品正常保修时，用户寄回的运费由用户承担，维修后寄回给用户的费用由我司承担。非正常保修产品来回运费均有用户承担。

**联系方式：**

地 址：北京市海淀区永翔北路 9 号中国航发大厦三层北京迅为电子有限公司

联系人：迅为开发板售后服务部 北京迅为电子有限公司

邮 编：100094

电 话：010-58957586

## 技术支持与开发定制

### 1、技术支持范围

- (1) 了解产品的软、硬件资源提供情况咨询；
- (2) 产品的软、硬件手册使用过程中遇到的问题；
- (3) 下载和烧写更新系统过程中遇到的问题；
- (4) 产品用户的资料丢失、更新后重新获取；
- (5) 产品的故障判断及售后维修服务。

### 2、技术讨论范围

由于嵌入式系统知识范围广泛，涉猎种类繁多，我们无法保证对各种问题都能一一解答，以下内容无法供技术支持，只能提供建议。

- (1) 源码如何理解和修改，电路板的自行设计制作或修改；
- (2) 如何编译和移植操作系统；
- (3) 用户在自行修改以及开发中遇到的软硬件问题。

### 3、技术支持方式

论坛：<http://bbs.topeetboard.com/forum.php>

电话：0312-6796610

邮箱：[support@topeetboard.com](mailto:support@topeetboard.com)

### 4、技术支持服务时间

上午 9:00--12:00,下午 13:30--17:30(周一至周六)

QQ 群主动技术支持：

上午 10:00--11:00,下午 15:00--16:00(周一至周六)

### 5、定制开发服务

本公司提供嵌入式操作系统底层驱动、硬件板卡的有偿定制开发服务，以缩短您的产品开

发周期。请将需求：

发送邮件到： [support@topeetboard.com](mailto:support@topeetboard.com)

联系电话: 0312-6796610

淘宝店铺 1:

<https://arm-board.taobao.com/?spm=a1z10.1-c-s.0.0.7bf93dd3q2C808>

淘宝店铺 2:

[https://shop459378556.taobao.com/shop/view\\_shop.htm?spm=a230r.1.14.48.25b84a81oPy9vl&user\\_number\\_id=2207480684984](https://shop459378556.taobao.com/shop/view_shop.htm?spm=a230r.1.14.48.25b84a81oPy9vl&user_number_id=2207480684984)

## 资料获取与后续更新

### 一. 资料的获取

#### (1) 百度网盘下载

网盘的链接在购买开发板后可以在迅为电子技术支持 QQ 群下载。如果链接有更新，会在群里贴通告

#### (2) 视频配套教程

根据用户购买开发板的产品类别，迅为电子会提供相应的光盘。

### 二 后续更新

后续文档、视频等资料的更新，为了确保您的资料是最新状态，请密切关注我们的动态，我们将会通过微信公众号和 QQ 群推送。关注“迅为电子”微信公众号，不定期分享教程、资料 and 行业干货及产品一线资料。



迅为电子  
让学习更容易，让开发更简单



## 第 1 章 TFTP&NFS 网络烧写

使用网络烧写的方式来进行调试是非常方便的，本章将介绍使用 tftp 在服务器下载 Linux 内核镜像和设备树，并使用 NFS 网络挂载根文件系统的方法。

文档中使用的开发环境：Ubuntu18.04 LTS

平台：迅为 iTOP-3568 开发板

此外，开发板和 Ubuntu 的 IP 要在同一网段下，本文档 Ubuntu 的 IP 地址为 192.168.1.9，开发板的 IP 地址设置为 192.168.1.16，关于网络配置不再赘述。

手册以 buildroot 系统为例，如使用 ubuntu 系统，Debian 系统，yocto 系统等系统，也可以参考本手册。

### 1.1 安装 TFTP 服务端

首先在 ubuntu 中执行以下命令安装 TFTP 服务：

```
apt-get install tftp-hpa tftpd-hpa
```

```
root@ubuntu:/home/topeet# apt-get install tftp-hpa tftpd-hpa
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-5.4.0-96-generic linux-hwe-5.4.0-headers-5.4.0-94 linux-hwe-5.4.0-headers-5.4.0-96 linux-image-5.4.0-96-generic linux-modules-5.4.0-96-generic linux-modules-extra-5.4.0-96-generic
Use 'apt autoremove' to remove them.
Suggested packages:
  pxellinux
The following NEW packages will be installed:
  tftp-hpa tftpd-hpa
0 upgraded, 2 newly installed, 0 to remove and 15 not upgraded.
Need to get 57.5 kB of archives.
After this operation, 172 kB of additional disk space will be used.
Get:1 http://mirrors.aliyun.com/ubuntu bionic/main amd64 tftp-hpa amd64 5.2+20150808-1ubuntu3 [18.3 kB]
Get:2 http://mirrors.aliyun.com/ubuntu bionic/main amd64 tftpd-hpa amd64 5.2+20150808-1ubuntu3 [39.1 kB]
Fetched 57.5 kB in 0s (144 kB/s)
Preconfiguring packages ...
Selecting previously unselected package tftp-hpa.
(Reading database ... 243327 files and directories currently installed.)
Preparing to unpack .../tftp-hpa_5.2+20150808-1ubuntu3_amd64.deb ...
Unpacking tftp-hpa (5.2+20150808-1ubuntu3) ...
Selecting previously unselected package tftpd-hpa.
Preparing to unpack .../tftpd-hpa_5.2+20150808-1ubuntu3_amd64.deb ...
Unpacking tftpd-hpa (5.2+20150808-1ubuntu3) ...
Setting up tftp-hpa (5.2+20150808-1ubuntu3) ...
Setting up tftpd-hpa (5.2+20150808-1ubuntu3) ...
Processing triggers for systemd (237-3ubuntu0.53) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for urcadhead (0.100.0-21) ...
root@ubuntu:/home/topeet#
```

安装完成以后创建 TFTP 服务器工作目录,并对 TFTP 的服务配置文件进行修改,具体步骤如下：

输入以下命令在家目录创建 tftpboot 文件夹，如下图所示：

```
mkdir -p /home/topeet/tftpboot
```

```
root@ubuntu:/home/topeet# mkdir -p /home/topeet/tftpboot
root@ubuntu:/home/topeet# ls
Desktop linux Linux rootfs.img tftpboot ubuntu20desktop work
root@ubuntu:/home/topeet#
```

然后输入以下命令修改 tftpboot 文件夹权限，如下图所示：

```
chmod 777 /home/topeet/tftpboot
```

```
root@ubuntu:/home/topeet# chmod 777 /home/topeet/tftpboot
root@ubuntu:/home/topeet# ls
Desktop  linux  Linux  rootfs.img  tftpboot  ubuntu250desktop  work
root@ubuntu:/home/topeet#
```

最后修改配置文件，打开配置文件以后如下图所示：

```
vim /etc/default/tftpd-hpa
```

```
File Edit View Search Terminal Help
1 # /etc/default/tftpd-hpa
2
3 TFTP_USERNAME="tftp"
4 TFTP_DIRECTORY="/var/lib/tftpboot"
5 TFTP_ADDRESS=":69"
6 TFTP_OPTIONS="--secure"
```

修改为如下内容：

```
TFTP_USERNAME="tftp"
TFTP_DIRECTORY="/home/topeet/tftpboot/"
TFTP_ADDRESS=":69"
TFTP_OPTIONS="-l-c-s"
```

修改完成如下图所示：

```
1 TFTP_USERNAME="tftp"
2 TFTP_DIRECTORY="/home/topeet/tftpboot/"
3 TFTP_ADDRESS=":69"
4 TFTP_OPTIONS="-l-c-s"
```

修改完成保存退出，在终端输入以下命令重启 tftpd-hpa 服务。

```
service tftpd-hpa restart
```

如下图所示：

```
root@ubuntu:/home/topeet/tftpboot# service tftpd-hpa restart
root@ubuntu:/home/topeet/tftpboot#
```

## 1.2 安装 NFS 服务

首先执行以下命令安装 NFS 服务：

```
apt-get install nfs-kernel-server
```

```

root@ubuntu:/home/topeet# apt-get install nfs-kernel-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
nfs-kernel-server is already the newest version (1:1.3.4-2.1ubuntu5.5).
The following packages were automatically installed and are no longer required:
 fonts-liberation2 fonts-opensymbol gir1.2-gst-plugins-base-1.0
 gir1.2-gstreamer-1.0 gir1.2-gudev-1.0 gir1.2-udisks-2.0
 grilo-plugins-0.3-base gstreamer1.0-gtk3 libboost-date-time1.65.1
 libboost-filesystem1.65.1 libboost-iostreams1.65.1 libboost-locale1.65.1
 libcdr-0.1-1 libclucene-contribs1v5 libclucene-core1v5 libcmis-0.5-5v5
 libcolamd2 libdazzle-1.0-0 libe-book-0.1-1 libdataserverui-1.2-2 libeot0
 libepubgen-0.1-1 libetonyek-0.1-1 libexiv2-14 libfreerdp-client2-2
 libfreerdp2-2 libgc1c2 libgee-0.8-2 libgexiv2-2 libgom-1.0-0 libgpgmepp6
 libgpod-common libgpod4 liblangtag-common liblangtag1 liblirc-client0
 liblua5.3-0 libmediaart-2.0-0 libmsspub-0.1-1 libodfgen-0.1-1 libqqwing2v5
 libraw16 librevenge-0.0-0 libsgutils2-2 libssh-4 libsuitesparseconfig5
 libvncclient1 libwinpr2-2 libxapian3 libxmlsec1-nss lp-solve
 media-player-info python3-mako python3-markupsafe syslinux-legacy
 usb-creator-common
Use 'apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ubuntu:/home/topeet#

```

然后使用以下命令创建 ubuntu20desktop 文件并赋予其权限如下图所示：

```
mkdir ubuntu20desktop
```

```
chmod 777 ubuntu20desktop/
```

```

root@ubuntu:/home/topeet# mkdir ubuntu20desktop
root@ubuntu:/home/topeet# chmod 777 ubuntu20desktop/
root@ubuntu:/home/topeet# ls
Desktop  .ltpboot  ubuntu20desktop
root@ubuntu:/home/topeet#

```

使用以下命令编辑/etc/exports 文件，并添加 NFS 服务的工作目录的路径：

```
vim /etc/exports
```

添加 NFS 目录：

以/home/topeet/ubuntu20desktop 为例，将其添加到/etc/exports 文件中，如下所示：

```
/home/topeet/ubuntu20desktop *(rw,nohide,insecure,no_subtree_check,async,no_root_squash)
```

```

1 # /etc/exports: the access control list for filesystems which may be exported
2 #           to NFS clients.  See exports(5).
3 #
4 # Example for NFSv2 and NFSv3:
5 # /srv/homes      hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
6 #
7 # Example for NFSv4:
8 # /srv/nfs4       gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
9 # /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
10 /home/topeet/ubuntu20desktop *(rw,nohide,insecure,no_subtree_check,async,no_root_squash)

```

配置参数说明：

rw：读写访问

sync：所有数据在请求时写入共享

async：NFS 在写入数据前可以相应请求



secure : NFS 通过 1024 以下的安全 TCP/IP 端口发送

insecure : NFS 通过 1024 以上的端口发送

wdelay : 如果多个用户要写入 NFS 目录, 则归组写入 (默认)

no\_wdelay : 如果多个用户要写入 NFS 目录, 则立即写入, 当使用 async 时, 无需此设置。

no\_hide 共享 NFS 目录的子目录

subtree\_check 如果共享/usr/bin 之类的子目录时, 强制 NFS 检查父目录的权限

no\_subtree\_check 和上面相对, 不检查父目录权限

all\_squash 共享文件的 UID 和 GID 映射匿名用户 anonymous, 适合公用目录。

no\_all\_squash 保留共享文件的 UID 和 GID

root\_squash root 用户的所有请求映射成如 anonymous 用户一样的权限 no\_root\_squas root 用户具有根目录的完全管理访问权限

使用以下命令重启 NFS 服务:

```
service nfs-kernel-server restart
```

最后使用命令: “showmount -e ”查看 nfs 目录是否成功挂载 (这里也可能要重启 ubuntu 才会又挂载目录), 如下图所示:

```
root@ubuntu:/home/topeet# showmount -e
Export list for ubuntu:
/home/topeet/ubuntu20desktop *
```

至此, Ubuntu nfs 服务器端的配置完成。

### 1.3 配置开发板网络

连接好网线, 上电启动开发板, 在加载阶段点击“ctrl+c”进入 uboot 命令行, 如下图所示:

```
VOP update mode to: 800x1280p0, type: MIPI1 for VP1
VOP VP1 enable Smart0[654x270->654x270@73x505] fmt[2] addr[0x7df04000]
final DSI-Link bandwidth: 396 Mbps x 4
CLK: (sync kernel. arm: enter 816000 KHz, init 816000 KHz, kernel 0N/A)
apll 1416000 KHz
dpll 780000 KHz
gp1l 1188000 KHz
cp1l 1000000 KHz
np1l 1200000 KHz
vp1l 600000 KHz
hp1l 24000 KHz
pp1l 200000 KHz
armclk 1416000 KHz
aclk_bus 150000 KHz
pclk_bus 100000 KHz
aclk_top_high 500000 KHz
aclk_top_low 400000 KHz
hclk_top 150000 KHz
pclk_top 100000 KHz
aclk_perimid 300000 KHz
hclk_perimid 150000 KHz
pclk_pmu 100000 KHz
Net: eth1: ethernet@fe010000, eth0: ethernet@fe2a0000
Hit key to stop autoboot('CTRL+C'): 0
=>
=>
=>
```

然后来到 ubuntu 虚拟机终端，输入命令“ifconfig”来查看虚拟机 ubuntu 的 ip，如下图所示：

```
root@ubuntu:/home/topeet# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.9 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 240e:341:5205:b000:b4d1:411c:2fd7:92d4 prefixlen 64 scopeid 0x0<
global>
    inet6 240e:341:5205:b000:4f55:3e3f:538f:4d71 prefixlen 64 scopeid 0x0<
global>
    inet6 fe80::9ccd:9e7:99c1:33ee prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:5d:90:7a txqueuelen 1000 (Ethernet)
    RX packets 2005 bytes 624457 (624.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 512 bytes 64495 (64.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 200 bytes 16309 (16.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 200 bytes 16309 (16.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@ubuntu:/home/topeet#
```

可以看到 ubuntu 虚拟机的 ip 地址为 192.168.1.9，然后输入以下命令设置开发板的网络，设置完成如下图所示：

```
setenv ipaddr 192.168.1.16
setenv serverip 192.168.1.9
setenv gateway 192.168.1.1
```

```
=> setenv ipaddr 192.168.1.16
=> setenv serverip 192.168.1.9
=> setenv gateway 192.168.1.1
=> █
```

然后输入以下命令测试开发板与 ubuntu 虚拟机之间的网络是否畅通，测试成功如下图所示：

```
ping 192.168.1.9
```

```
=> ping 192.168.1.9
Using ethernet@fe2a0000 device
host 192.168.1.9 is alive
=> █
```

如果环境变量不知道哪里设置错了，或者想要恢复成 emmc 启动，可以使用以下命令重置环境变量

```
env default -a
saveenv
```

## 1.4 tftp 加载内核和设备树

内核镜像 boot.img 和设备树镜像 rk3568-evb1-ddr4-v10-linux.dtb 需要通过 Linux 源码编译获得，具体的源码编译步骤可以查看“05\_【北京迅为】itop-3568 开发板源码编译手册 v1.0.pdf”文件。

内核源码编译完成之后将内核镜像 boot.img 和设备树文件 rk3568-evb1-ddr4-v10-linux.dtb 拷贝到第一小节创建的“/home/topeet/tftpboot”目录下，拷贝完成如下图所示：

```
root@ubuntu:/home/topeet/tftpboot# ls
boot.img rk3568-evb1-ddr4-v10-linux.dtb
root@ubuntu:/home/topeet/tftpboot#
```

然后根据上一小节内容配置自身 ip 和 server ip，设置完成之后，使用以下命令将内核和设备树通过网络加载进内存之中，如下图所示：

```
setenv bootcmd "tftpboot 0x08300000 rk3568-evb1-ddr4-v10-linux.dtb;tftpboot 0x0027f800
boot.img ; bootm 0x0027f800 - 0x08300000"
```

```
=> setenv bootcmd " tftpboot 0x08300000 rk3568-evb1-ddr4-v10-linux.dtb;tftpboot 0x0027f800 boot.img ;bootm 0x0027f800 - 0x08300000"
=> █
```

通过以上命令将 boot.img 镜像下载到首地址为 0x0027f800 的内存空间；将 rk3568-evb1-ddr4-v10-linux.dtb 设备树镜像将下载到首地址为 0x08300000 的内存空间；随后输入以下命令，等待网络下载内核和设备树，如下图所示：

```
env run bootcmd
```





```
root@ubuntu:/home/topeet/Linux/rk356x_linux/rockdev# ls
boot.img          oem.img          rootfs.ext4      update.img
MiniLoaderAll.bin parameter.txt    rootfs.img       userdata.img
misc.img          recovery.img     uboot.img
root@ubuntu:/home/topeet/Linux/rk356x_linux/rockdev#
```

在第二小节中创建的 nfs 目录为“/home/topeet/ubuntu20desktop”，将 rootfs.ext4 文件拷贝到“/home/topeet”目录下，拷贝完成如下图所示：

```
root@ubuntu:/home/topeet# ls
Desktop  linux  Linux  rootfs.ext4  tftpboot  ubuntu20desktop  work
root@ubuntu:/home/topeet#
```

然后使用以下命令将文件系统挂载到 ubuntu20desktop 文件夹，如下图所示：

```
mount rootfs.ext4  ubuntu20desktop/
```

```
root@ubuntu:/home/topeet# mount rootfs.ext4 ubuntu20desktop/
root@ubuntu:/home/topeet#
```

至此 ubuntu 虚拟机中的配置就完成了。

## 2. update.img 解包

为了方便，提供好的镜像为多个独立镜像整合在一起的 update.img 镜像文件，可以使用 RKDevTool 工具来进行 update.img 的解包，获得 rootfs.img 文件系统镜像，可以通过“06\_【北京迅为】itop-3568 开发板快速烧写手册 v1.0.pdf”手册中的 3.4 小节“解包 update.img”来进行解包解包完成之后如下图所示：

« RKDevTool_Release_v2.84 > Output > Android > Image					在 Image 中搜索
名称	修改日期	类型	大小		
boot.img	2022/6/6 18:31	好压 IMG 压缩文件	19,116 KB		
config.cfg	2022/6/6 15:02	txtfile	6 KB		
MiniLoaderAll.bin	2022/6/6 18:31	BIN 文件	299 KB		
misc.img	2022/6/6 13:48	好压 IMG 压缩文件	48 KB		
oem.img	2022/6/6 13:48	好压 IMG 压缩文件	17,408 KB		
parameter.txt	2022/6/6 18:31	文本文档	1 KB		
recovery.img	2022/6/6 13:48	好压 IMG 压缩文件	29,715 KB		
rootfs.img	2022/6/6 18:31	好压 IMG 压缩文件	5,242,880...		
trust.img	2022/6/6 18:31	好压 IMG 压缩文件	4,096 KB		
uboot.img	2022/6/6 18:31	好压 IMG 压缩文件	4,096 KB		
userdata.img	2022/6/6 13:48	好压 IMG 压缩文件	5,120 KB		

在第二小节中创建的 nfs 目录为“/home/topeet/ubuntu20desktop”，将 rootfs.ext4 文件拷贝到“/home/topeet”目录下，拷贝完成如下图所示：

```
root@ubuntu:/home/topeet# ls
Desktop  linux  Linux  rootfs.img  tftpboot  ubuntu20desktop  work
root@ubuntu:/home/topeet#
```



然后使用以下命令讲文件系统挂载到 ubuntu20desktop 文件夹，如下图所示：

```
mount rootfs.img ubuntu20desktop/
```

```
root@ubuntu:/home/topeet# mount rootfs.img ubuntu20desktop/  
root@ubuntu:/home/topeet#
```

至此 ubuntu 虚拟机中的配置就完成了。

### 1.5.2 网络挂载

开发板最新的 uboot 镜像已经适配了网络功能，iTOP-3568 开发板两个网口都支持 uboot 网络，作者选择使用 eth1。

Ubuntu 虚拟机配置完成之后，回到开发板 uboot 终端，输入以下命令设置要挂载的文件系统，**注意：标黄的部分是 nfs 挂载目录，要根据实际情况进行修改。**

```
setenv bootargs "root=/dev/nfs root=/dev/nfs nfsroot=${serverip}:/home/topeet/ubuntu20desktop,v3 rw  
ip=${ipaddr}:${serverip}:${gateway}:255.255.255.0::eth1:off"  
saveenv
```

```
=> setenv bootargs "root=/dev/nfs root=/dev/nfs nfsroot=${serverip}:/home/topeet/ubuntu20desktop,v3 rw  
ip=${ipaddr}:${serverip}:${gateway}:255.255.255.0::eth1:off"  
=> saveenv  
Saving Environment to ENV_BLK...  
Writing to mmc(0)... done  
=>
```

最后输入以下命令进行文件系统的网络挂载，挂载成功如下图所示：

```
env run bootcmd
```

```
[ 5.646659] #0: rockchip_rk809-codec  
[ 5.668900] VFS: Mounted root (nfs filesystem) on device 0:16.  
[ 5.669670] devtmpfs: mounted  
[ 5.673439] Freeing unused kernel memory: 1600K  
[ 5.686310] Run /sbin/init as init process  
[ 6.182227] systemd[1]: Failed to insert module 'autofs4': No such file or directory  
[ 6.364222] systemd[1]: systemd 229 running in system mode. (+PAM +AUDIT +SELINUX +IMA +APPARMOR +SMACK +SYSVINIT +UTMP +LIBCRYPT  
Welcome to Ubuntu 16.04.7 LTS!
```

系统正常启动完成之后，如下图所示：

```
[ 44.906215] rc.local[873]: , prefix: PRODUCT=  
[ 44.910637] rc.local[873]: line: PRODUCT=bda/b720/200  
[ 44.915634] rc.local[873]: , prefix: PRODUCT=  
[ 44.920099] rc.local[873]: pid:vid : 0bda:b720  
[ 44.924531] rc.local[873]: found device pid:vid : 0bda:b720  
[ 44.930135] rc.local[873]: wifi detectd return ret:0  
[ 44.934985] rc.local[873]: USB WIFI identify success  
[ 44.939852] rc.local[873]: check_wifi_chip_type_string: RTL8723BU  
[ 44.945939] rc.local[873]: Cannot create "/data/wifi_chip": No such file or directorywifibt_load_driver matched ko file  
[ 44.959883] rc.local[873]: wifibt_load_driver insmod /system/lib/modules/8723bu.ko  
[ 44.967413] rc.local[873]: wifibt_load_driver insmod /system/lib/modules/8723bu.ko failed  
  
Ubuntu 16.04.7 LTS localhost.localdomain ttyFIQ0  
localhost login: █
```

至此 nfs 挂载文件系统章节就结束了。