

前言：

- 这是一份使用Ubuntu系统搭建hi3559CV100开发环境的操作指南类型的手册
- 我觉得最难的事情就是开发环境的配置了。因此记录下hi3559CV100的开发环境搭建过程。
- 同时我希望这份记录能像是一份面向使用者的操作指南，所以我会逐步记录该怎么操作，并简单阐述为什么选择这个操作。

逐步操作流程：

安装minicom:

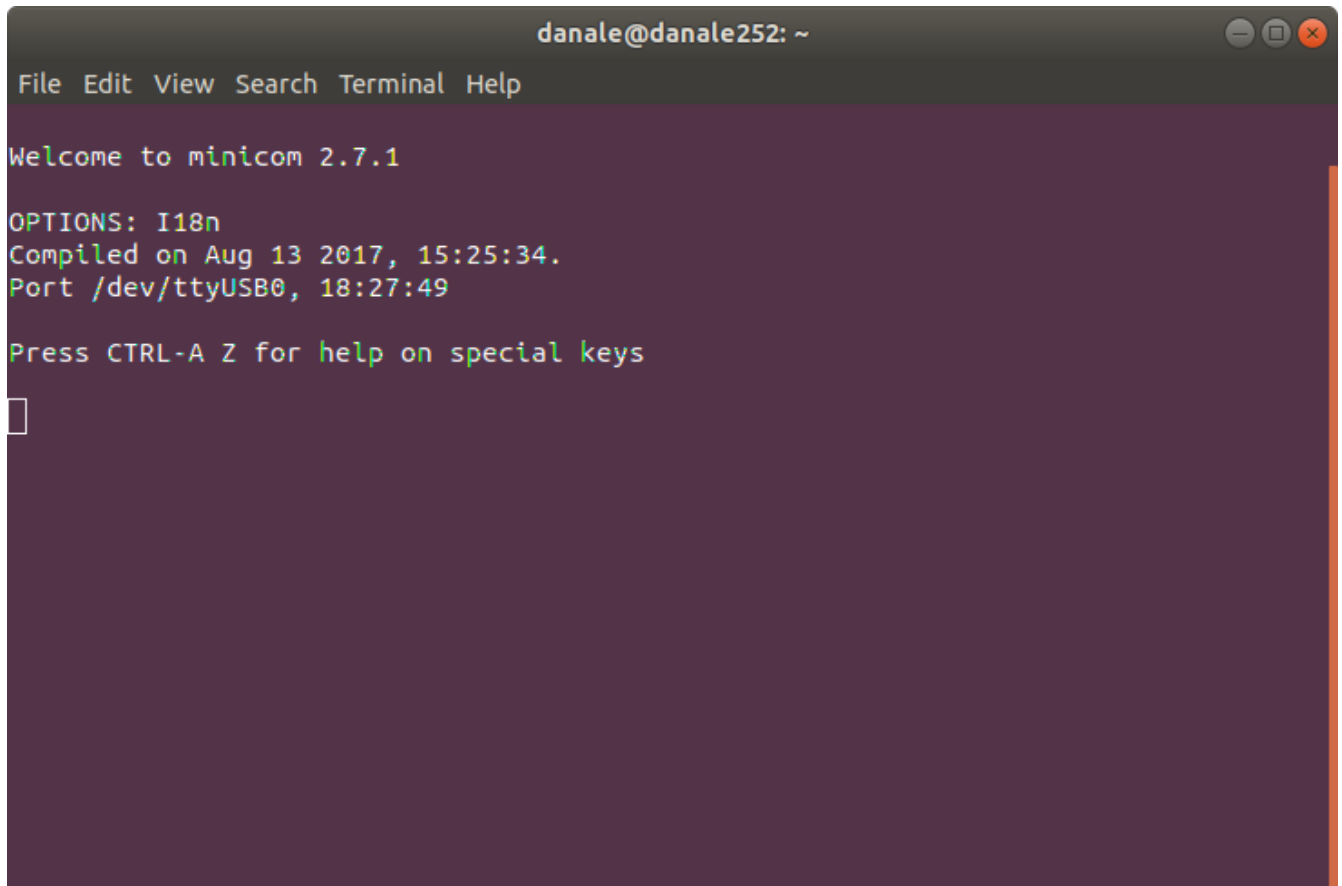
请把网线，串口，电源线分别接好

安装和设置minicom:

```
sudo apt-get install minicom
```

```
sudo minicom
```

完成上面两步之后，如果你是第一次使用（之后就不用重复设置了），要对minicom进行设置，过程如下：



```
danale@danale252: ~
File Edit View Search Terminal Help

Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB0, 18:27:49

Press CTRL-A Z for help on special keys
█
```

1. 按住ctrl+A，松开，再按Z进入如下界面

```
danale@danale252: ~
File Edit View Search Terminal Help

Welcome to minicom 2.7.1
Options: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB0, 18:27:49

Press CTRL-A Z for help on special keys

-----[configuration]-----+
| Filenames and paths         |
| File transfer protocols     |
| Serial port setup           |
| Modem and dialing           |
| Screen and keyboard         |
| Save setup as dfl           |
| Save setup as..             |
| Exit                         |
+-----+

Minicom Command Summary

Commands can be called by CTRL-A <key>

Main Functions                                Other Functions
Dialing directory..D  run script (Go)....G | Clear Screen.....C
Send files.....S    Receive files.....R | cOnfigure Minicom..O
comm Parameters....P  Add linefeed.....A | Suspend minicom....J
Capture on/off.....L Hangup.....H | eXit and reset....X
send break.....F    initialize Modem...M | Quit with no reset.Q
Terminal settings..T run Kermit.....K | Cursor key mode....I
lineWrap on/off....W local Echo on/off..E | Help screen.....Z
Paste file.....Y    Timestamp toggle...N | scroll Back.....B
Add Carriage Ret...U

Select function or press Enter for none.

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | tyUSB0
```

2. 按o，进入如下界面

```
danale@danale252: ~
File Edit View Search Terminal Help

Welcome to minicom 2.7.1
Options: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB0, 18:27:49

Press CTRL-A Z for help on special keys

-----[configuration]-----+
| Filenames and paths         |
| File transfer protocols     |
| Serial port setup           |
| Modem and dialing           |
| Screen and keyboard         |
| Save setup as dfl           |
| Save setup as..             |
| Exit                         |
+-----+

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | tyUSB0
```

3. 选中Serial port setup，进入设置界面，按照如下图中的内容填写

```
danale@danale252: ~
File Edit View Search Terminal Help

Welcome to minicom 2.7.1

OPTI+-----+
Comp| A -   Serial Device       : /dev/ttyUSB0
Port| B - Lockfile Location      : /var/lock
    | C -   Callin Program       :
Pres| D -   Callout Program       :
    | E -   Bps/Par/Bits         : 115200 8N1
    | F - Hardware Flow Control  : No
    | G - Software Flow Control  : No
    |
    | Change which setting? 
+-----+

    | Screen and keyboard
    | Save setup as dfl
    | Save setup as..
    | Exit
+-----+

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | tyUSB0
```

4. 按Enter键后再选中Save setup as dfl保存

```
danale@danale252: ~
File Edit View Search Terminal Help

Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB0, 18:27:49

Press CTRL-A Z for help on special keys

+-----[configuration]-----+
| Filenames and paths
| File transfer protocols
| Serial port setup
| Modem and dialing
| Screen and keyboard
| Save setup as dfl
| Save setup as..
| Exit
+-----+

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | tyUSB0
```

5. 选中Exit退出，然后再点击Enter键，这里是uboot的界面，说明不是完全的裸板

```
danale@danale252: ~  
File Edit View Search Terminal Help  
Welcome to minicom 2.7.1  
OPTIONS: I18n  
Compiled on Aug 13 2017, 15:25:34.  
Port /dev/ttyUSB0, 18:40:20  
Press CTRL-A Z for help on special keys  
hisilicon #
```

这样就可以在**Ubuntu**终端通过串口对板子进行命令操作了

安装和设置**tftp**：

1. 新开一个终端

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

```
vi /etc/default/tftpd-hpa
```

2. 将内容添加或修改如下：

```
TFTP_USERNAME="tftp"  
TFTP_DIRECTORY="/tftpboot" //去建立这个文件夹用来存放需要传递的文件，这里用到内核镜像文件和文件系统文件  
TFTP_ADDRESS=":69"  
TFTP_OPTIONS="--secure -c"
```

3. 重启tftp服务

```
service tftpd-hpa restart
```

4. 关闭防火墙：

```
iptables -F
```

有了**minicom**和**tftp**这两个工具我们就可以开始烧写内核镜像和文件系统了，

算法开发不需要使用到LiteOS，我们采用单系统**Linux**方案

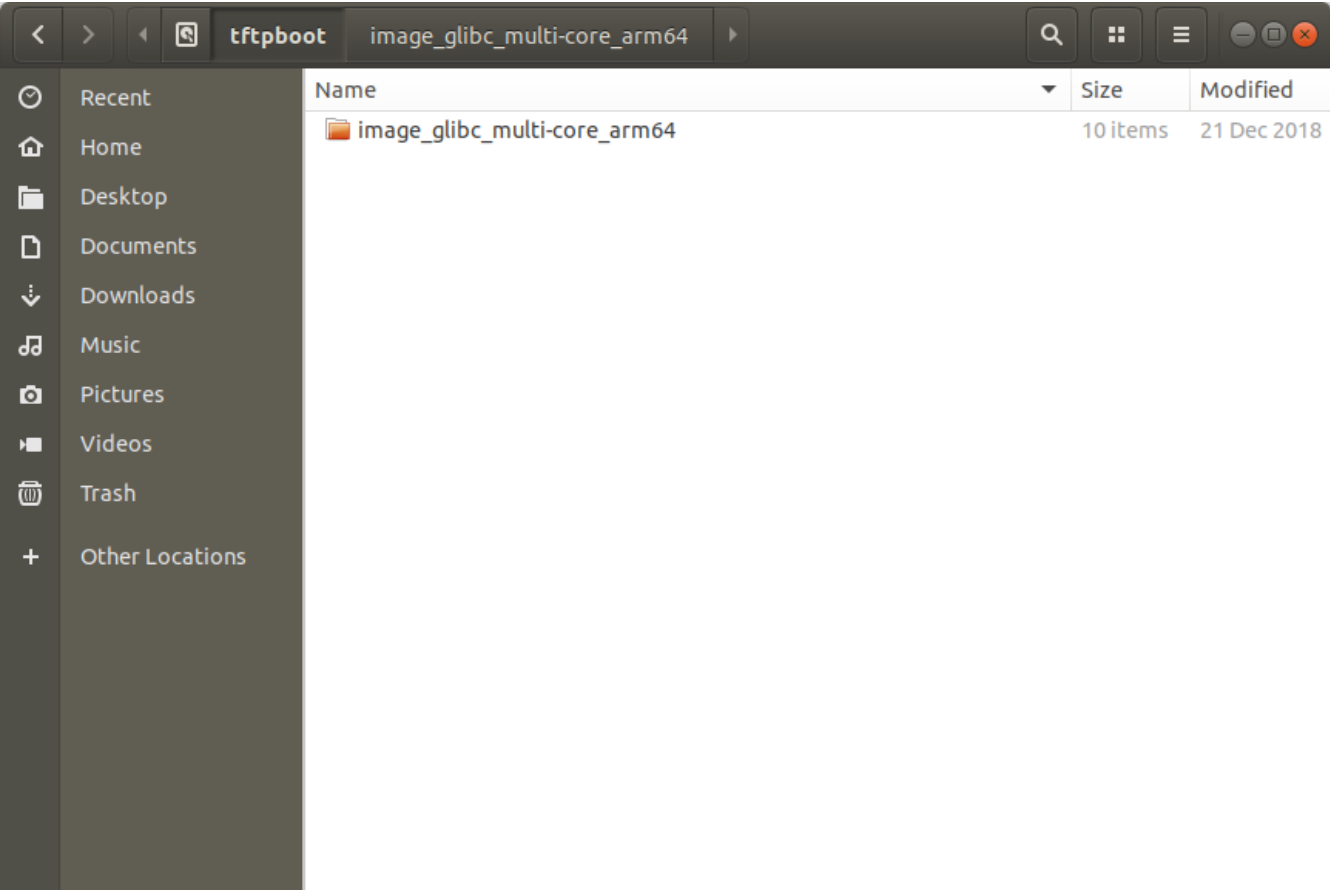
- 现在我们开始通过minicom界面使用tftp的功能来烧写内核镜像和文件系统文件

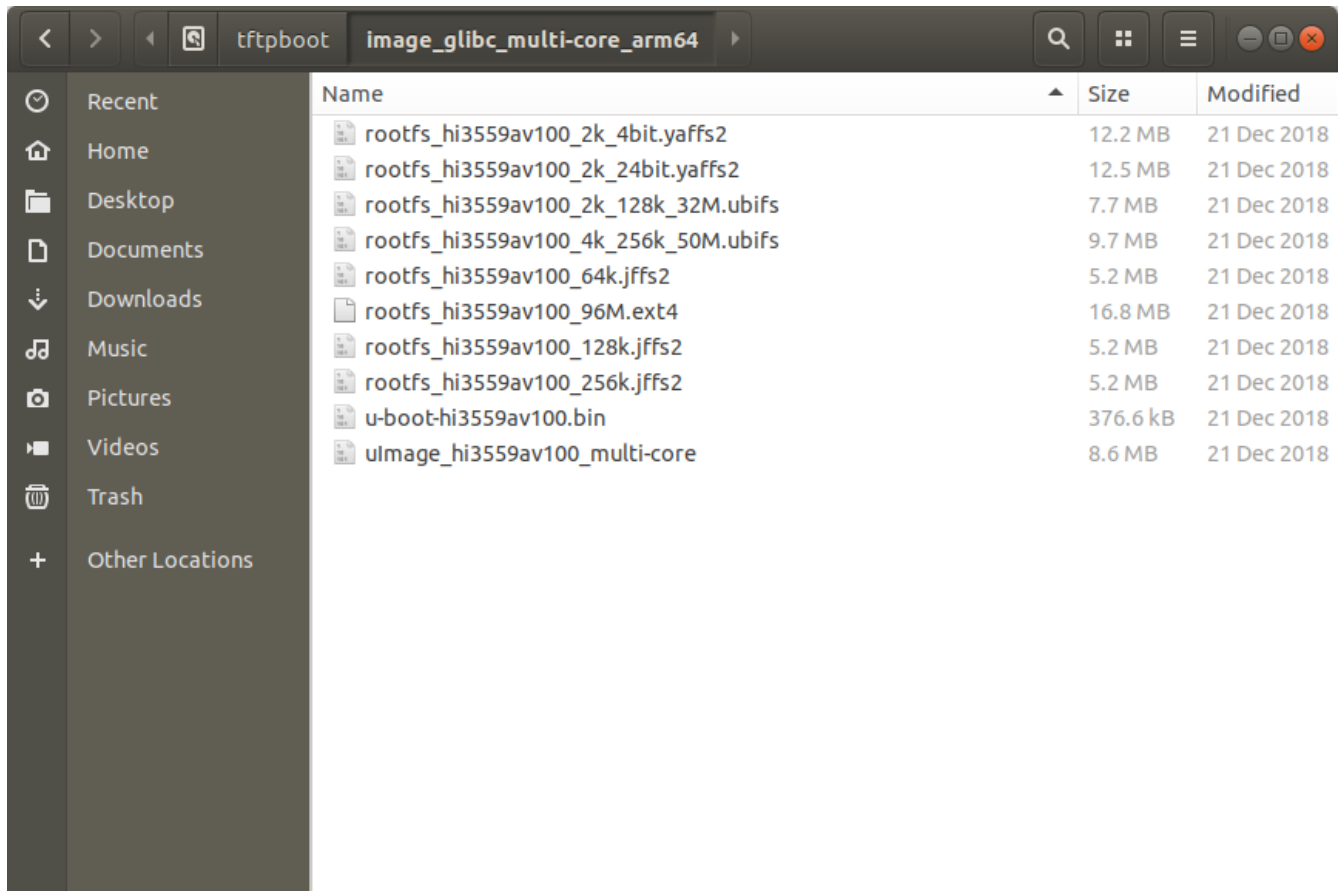
1. 地址空间说明

1M	9M	16M
uboot	kernel	rootfs

以下操作基于图示的地址空间分配，也可以根据实际情况进行调整

1. 先将包含了要烧写的内核镜像文件和文件系统文件image_glibc_multi-core_arm64拷贝到/tftpboot文件夹中





2. 查看flash模式：（这块板子的flash是NAND FLASH）

```
getinfo bootmode
```

3. 根据 2 选择使用启动介质烧写指南的方法，3559CV100用NAND FLASH，需要擦除的区域大小需要根据文件的实际大小改动一下。

- 烧写linux内核

```
mw.b 0x44000000 0xff 0x9000000
tftp 0x44000000 uImage_hi3559av100_multi-core
nand erase 0x100000 0x900000
nand write 0x44000000 0x100000 0x900000
```

- 烧写文件系统

```
mw.b 0x44000000 0xff 0x1000000
tftp 0x44000000 rootfs_hi3559av100_2k_24bit.yaffs2
nand erase 0xA00000 0x1000000
nand write.yaffs 0x44000000 0xA00000 0xcfeb00 (0xcfeb00 为 rootfs 文件实际大小)
```

- 设置启动参数

```
setenv bootargs 'mem=512M console=ttyAMA0,115200 root=/dev/mtdblock2 rw rootfstype=yaffs2
mtdparts=hinand:1M(boot),9M(kernel),16M(rootfs)'
```

```
setenv bootcmd 'nand read 0x44000000 0x100000 0x900000;bootm 0x44000000'
```

```
saveenv
```

- 重启系统

```
reset
```

内核和文件系统烧写完成。然后把板子先暂时放在一边。

（目前寄过去的开发版已经完成了之前的过程）

给自己的电脑安装**nfs-kernel-server**

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

先新建一个文件夹/root/rootfs作为服务器目录

nfs-kernel-server配置：

```
vi /etc/exports
```

添加以下第一行内容：

```
/root/rootfs 193.168.1.*(rw,sync,no_root_squash)
#/root/rootfs    nfs服务器目录名
#193.168.1.*     可以访问你的nfs的IP范围
#(rw,sync,no_root_squash)权限
```

终端输入下面命令使得修改生效

```
exportfs -a
```

重启NFS服务

```
/etc/init.d/rpcbind restart
/etc/init.d/nfs-kernel-server restart
```

关闭防火墙

```
iptables -F
```

nfs-kernel-server安装配置完成，可以从远端mount了

开始使用**minicom**进一步开发：

```
sudo minicom
```

- 给板子配网和挂nfs，以及加载外设驱动
- 这三件事可以同时开机启动的时候进行
- 进入开机启动文件

```
vi /etc/profile
```

添加:

```
ifconfig eth0 172.19.1.143 netmask 255.255.255.0
route add default gw 172.19.1.1

mount -t nfs -o nolock -o tcp -o rsize=32768,wsiz=32768 你的UbuntuIP地址:/root/rootfs/
/mnt

cd /mnt/Hi3559AV100_SDK_V2.0.2.0/mpp/out/linux/multi-core/ko
./load3559av100_multicore -i -sensor0 imx477 -sensor1 imx477
```

重启：

```
reboot
```

在板子上运行Demo

```
cd /mnt/Hi3559AV100_SDK_V2.0.2.0/mpp/sample/svp/multi-core/nnie
./sample_nnie_main 0
```

一些琐碎细节：

- 开发的环境是怎么工作的：Ubuntu终端使用minicom通过串口对板子上的linux系统进行访问和操作，开发过程中板子上的内存是不够使用的，所以文件系统使用mount nfs通过以太网访问Ubuntu上的nfs服务器即/root/rootfs。我们直接在Ubuntu对/root/rootfs中源代码进行开发和编译，然后通过minicom运行可执行文件。
- 对源码的开发在Ubuntu的Vscode（任一款你喜欢的编辑器都行）进行，使用终端在mpp/sample目录下输入make linux编译链接完成后，通过minicom在板端运行你的可执行文件
- 将sdk中的mpp文件在Vscode中打开，进入/mpp/sample/svp/multi-core/nnie进行开发