

OpenTSN 操作手册

(V1.0)



*Open***TSN**

1、文档的目的

本文档描述如何对 OpenTSN 的演示环境进行搭建。

所需的设备：8 个 openbox_s4、装有 linux 系统的计算机 1 台、装有 linux 系统并安装了 Qt 环境的计算机 2 台、普通交换机 1 个、有线摄像头 1 个、普通计算机 1 台。

搭建的实验环境如下图 1 所示：

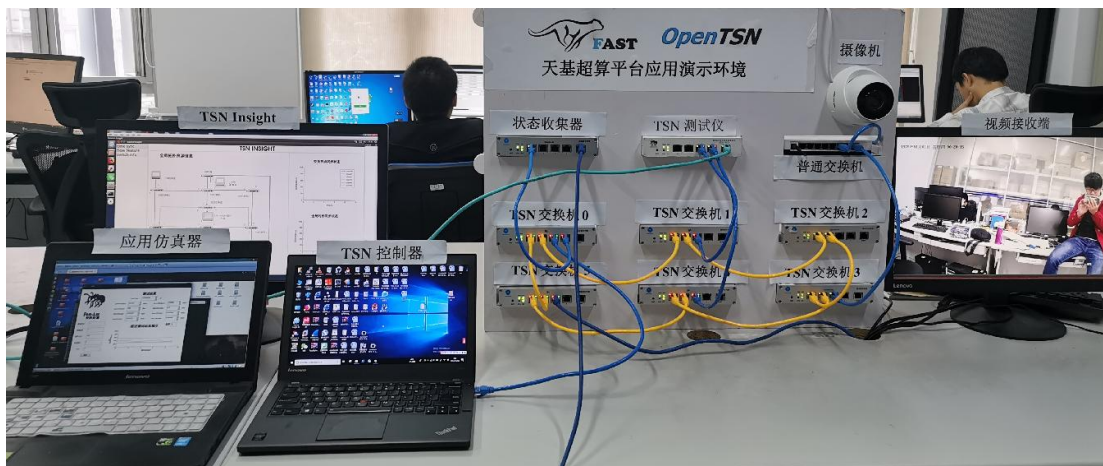


图1 实验环境

搭建的环境能实现的效果：（1）能够在普通计算机上看到无明显延迟无丢帧的摄像头传输的视频画面；（2）能够在 linux 的 Qt 环境上看到当前 TSN 网络的时间同步精度；（3）能够在 linux 的 Qt 环境上看到时间敏感流在 TSN 网络中传输的实际延迟与理论延迟的关系。

2、设备的配置

2.1 TSN 交换机

TSN 网络中使用 openbox_s4 组网，下面对 openbox_s4 进行简单的介绍。

2.1.1 openbox_s4 的各接口介绍

openbox_s4 的正面有 4 个数据网络、1 个管理网络、1 个复位按钮以及 4 个 led 灯，如下图 2 所示：

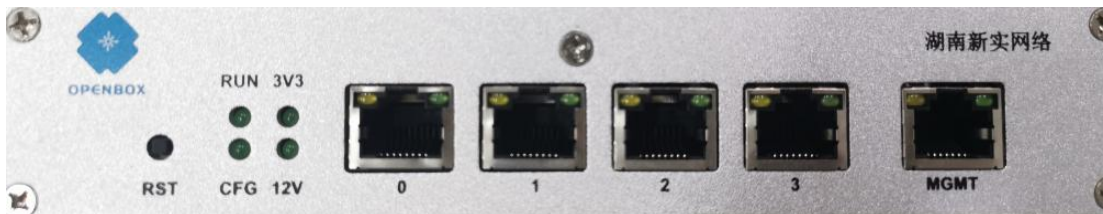


图2 openbox_s4 正面图

openbox_s4 的背面有 JTAG 接口、USB 接口、COM 串口、开关和电源接口，如下图 3 所示：



图3 openbox_s4 背面图

2.1.2 openbox_s4 配置成 OpenTSN 交换机

以 6 个 TSN 交换节点组网的拓扑为例进行下面的步骤介绍。

➤ 步骤一：下载文件

下载 bin/tsn_switch/目录下的全部内容，如下图 4 所示：



图4 下载码云的内容

➤ 步骤二：将文件拷贝到设备中（拷贝的方法可参考附录一）

把 BOOT.bin 拷贝到 openbox-S4 的 TF 卡中 mnt 目录下，如下图 5 所示：



图5 将 BOOT.BIN 拷贝到 mnt 目录下

把 S99hnxs.sh、deviceID_config.sh 和 deviceID 拷贝到 mnt/openbox/config 中，如下图 6 所示：

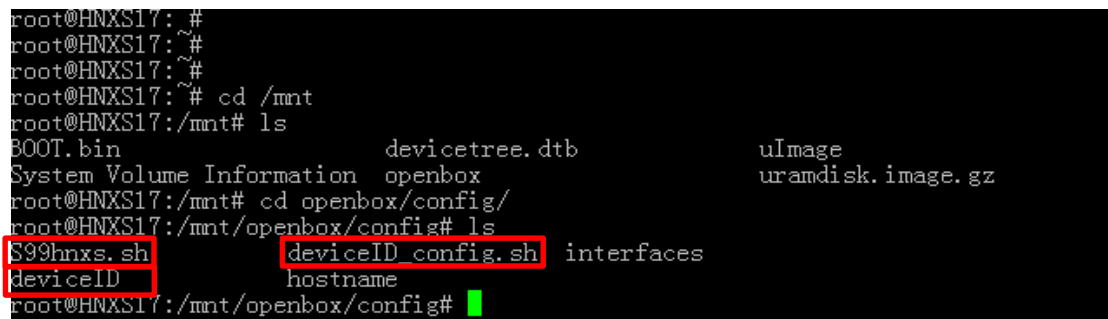


图6 将上述三个文件拷贝到 mnt/openbox/config 中

➤ 步骤三：将交换机进行编号

重启设备，通过串口登录到 openbox_s4 运行界面上，具体操作可查看附录二。

进入 putty 界面，如下图 7，并进入到 /mnt/openbox/config 目录下，操作命令如下图 8：

前拓扑中有总共 6 个交换节点，因此需要将 6 个 openbox_s4 分别编号为 0 到 5。

操作为：按一下字母“a”键，即可对数字进行修改，将 6 个 openbox_s4 分别修改为 0-5；修改完成后按一下“Esc”键，再同时按下“Shift”+“;”键，最后输入“wq”即可。

至此，各 openbox_s4 即配置成了 TSN 交换节点。

2.2 流量分析器（状态收集器）

流量分析器也是使用 openbox_s4 做的，下面介绍如何将 openbox_s4 设置成流量分析器。

2.2.1 openbox_s4 配置成流量分析器

➤ 步骤一：下载文件

下载 tool/流量分析器/目录下的所有文件，如下图 12 所示：



图12 下载码云的内容

BOOT.bin: 状态分析器中的硬件环境

controller_info: 连接到控制器中的文件配置，其中 controller_ip 代表与该 openbox 连接的 Linux 设备的 IP 地址，controller_port 代表端口号，使用默认端口号 8080。

flow_info: 表示流的特征文件。

topology_info: 表示拓扑文件。

tsn_NA: 可执行文件，运行流量分析器时执行该文件./tsn_NA

➤ 步骤二：将文件拷贝到设备中

把所有文件拷贝到 openbox-S4 中的 TF 卡的 mnt 目录下，如下图 13 所示：

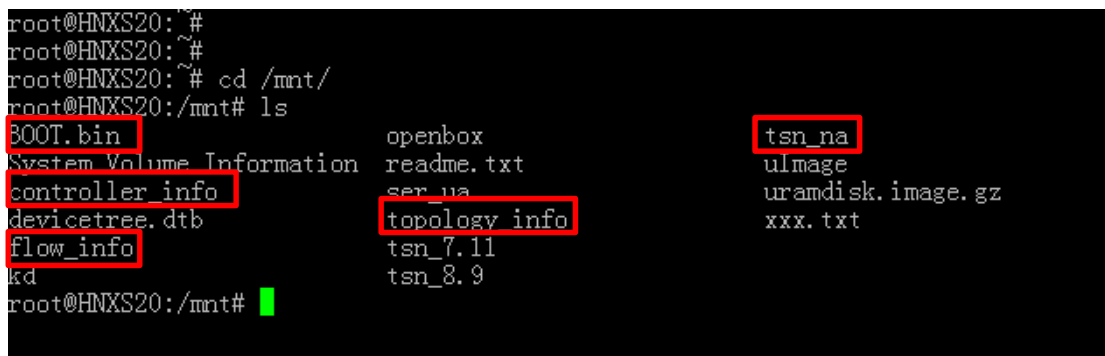


图13 将上述所有文件拷贝到 mnt 目录下

➤ 步骤三：重启即可。

2.3 ANT 测试仪（TSN 测试仪）

ANT 测试仪也是使用 openbox_s4 做的，下面介绍如何将 openbox_s4 设置成 ANT 测试仪。

2.3.1 openbox_s4 配置成 ANT 测试仪

➤ 步骤一：下载文件

下载 tool/ANT 测试仪/目录下的 BOOT.bin、ant 和 upload.sh 文件，如下图 14 所示：



图14 下载码云的内容

➤ 步骤二：将文件拷贝到设备中

把下载的三个文件拷贝到 openbox-S4 中的 TF 卡的 mnt 目录下，如下图 15 所示：

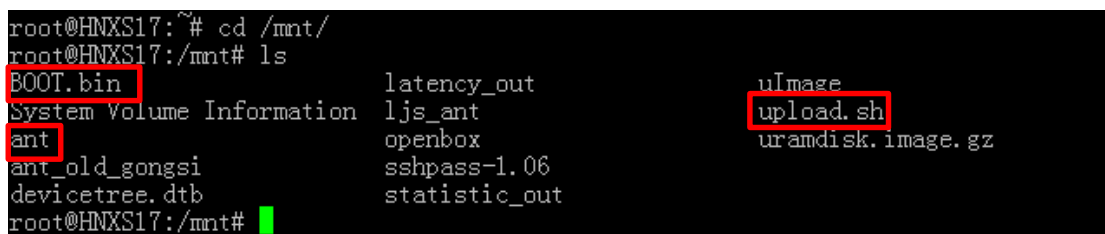


图15 将上述所有文件拷贝到 mnt 目录下

➤ 步骤三：重启即可。

2.4 TSN_CNC (TSN 控制器)

TSN_CNC 使用的是装有 linux 系统的 PC。

2.4.1 TSN_CNC 的配置

➤ 步骤一：下载文件

下载 bin/tsn_CNC/目录下的所有文件，如下图 16 所示：



图16 下载码云的内容

➤ 步骤二：将文件拷贝到 linux 系统中即可。

2.5 TSN_insight

测试仪控制终端使用的是装有 linux 系统的 PC，同时在 linux 系统上还需安装 Qt 环境

2.5.1 TSN_insight 的配置

➤ 步骤一：下载文件

下载 bin/tsn_insight/目录下的 tsn_insight 和 TuoPu 文件，如下图 17 所示，并拷贝到安装 Qt 环境的 linux 设备中：



图17 下载码云的内容

- 步骤二：将文件拷贝到 linux 系统中即可。

2.6 测试仪控制终端（应用仿真器）

测试仪控制终端使用的是装有 linux 系统的 PC，同时在 linux 系统上还需安装 Qt 环境。

2.6.1 测试仪控制终端的配置

- 步骤一：下载文件

下载 tool/ANT 测试仪/GUI/目录下的 ANT 和 ant.jpg 文件，如下图 18 所示，并拷贝到安装 Qt 环境的 linux 设备中：



图18 下载码云的内容

- 步骤二：将文件拷贝到 linux 系统中。
- 步骤三：在 linux 系统上安装 ssh-server,ssh-pass 服务。
- 步骤四：在测试仪控制终端和 ANT 测试仪建立 scp 通信。
- 步骤五：执行./ANT 即可。

2.7 网络摄像头

摄像头是有线的摄像头即可。同时拿一台普通交换机作为视频接收端，两者之间必须在接入网络之前进行测试，看是否能够正常使用。

3、设备的连接

3.1 TSN 网络的连接

根据 2.1 节，将各 openbox_s4 都编号为 0、1、2、3、4、5，即对应了 TSN 交换机 0、TSN 交换机 1、TSN 交换机 2、TSN 交换机 3、TSN 交换机 4、TSN 交换机 5。将 6 个交换机按如下图 19 的方式连接：



图19 TSN 网络连接方式

具体连接方式为：交换机 0 的 1 号口连交换机 1 的 0 号口、交换机 1 的 1 号口连交换机 2 的 0 号口、交换机 2 的 1 号口连交换机 3 的 0 号口、交换机 3 的 1 号口连交换机 4 的 0 号口、交换机 4 的 1 号口连交换机 5 的 0 号口、交换机 5 的 1 号口连交换机 0 的 0 号口。

3.2 流量分析器与 TSN 网络的连接

流量分析器与 TSN 网络的连接方式如下图 20 所示：



图20 流量分析器与 TSN 网络的连接图

具体连接方式为 TSN 交换机 0 的 3 号口连接到流量分析器（状态收集器）的 0 号口即可。

3.3 TSN 测试仪与 TSN 网络的连接

ANT 测试仪与 TSN 网络的连接方式如下图 21 所示：



图21 ANT 测试仪与 TSN 网络的连接图

具体连接方式为 TSN 交换机 4 的 2 号口连接到 ANT 测试仪（TSN 测试仪）的 3 号口、TSN 交换机 1 的 2 号口连接到 ANT 测试仪（TSN 测试仪）的 2 号口即可。

3.4 TSN_CNC 与 TSN 网络的连接

TSN_CNC（TSN 控制器）与 TSN 网络的连接方式如下图 22 所示：



图22 TSN_CNC 与 TSN 网络的连接图

具体连接方式为 TSN 交换机 0 的 2 号口连接到 TSN_CNC（TSN 控制器）上即可。

3.5 TSN_insight 与流量分析器的连接

TSN_insight 与流量分析器（状态收集器）的连接方式如下图 23 所示：



图23 TSN_insight 与流量分析器的连接图

具体连接方式为 TSN_insight 连接到状态收集器（流量分析器）的管理网口上即可。

3.6 测试仪终端与 TSN 测试仪的连接

测试仪终端（应用仿真器）与 TSN 测试仪的连接方式如下图 24 所示：



图24 测试仪终端与 TSN 测试仪的连接图

具体连接方式为应用仿真器（测试仪控制终端）连接到 TSN 测试仪的管理网口上即可。

3.7 摄像头、视频接收端与 TSN 网络的连接

摄像头、视频接收端与 TSN 网络的连接方式如下图 25 所示：



图25 摄像头、视频接收端与 TSN 网络的连接图

具体连接方式为摄像头连接到一个普通交换机再将普通交换机连接到 TSN 交换机 3 的 2 号口上、视频接收端连接到 TSN 交换机 5 的 2 号口上。

使用普通交换机是为了将摄像头与 TSN 网络做网络的适配，原因是摄像头是百兆的，而 TSN 网络是千兆的，因此做了一个适配，如果摄像头是千兆的，那么就将摄像头直接连在 TSN 交换机 3 的 2 号口上。

4、TSN 网络外连设备应用的使用

4.1 TSN_CNC 的使用

按照 3.4 节的方法将 TSN_CNC 与 TSN 网络进行连接，在 TSN_CNC 的 linux 系统上执行以下命令即可完 TSN_CNC 对 TSN 网络的配置。

- 执行 ./tsn_CNC 时，出现如下图 26 提示信息：

```
joejiang@ubuntu: /mnt/hgfs/tsn_cnc-B/lib/libsouthio
joejiang@ubuntu:/mnt/hgfs/tsn_cnc-B/lib/libsouthio$ ./tsn_CNC
./tsn_cnc all
./tsn_CNC sw_id
or
./tsn CNC sw id direction tb B tb rate time slot
default: direction = 0,tb_B = 2048 tb_rate = 100,time slot = 14
joejiang@ubuntu:/mnt/hgfs/tsn_cnc-B/lib/libsouthio$
```

图26 执行 “./tsn_CNC”

- 执行 ./tsn_CNC all，配置拓扑中所有交换机，效果图如下图 27：

```

root@hnxs-VirtualBox: /mnt/bd/tsn_cnc-2019.9.16/lib/libsouthio

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
TSN-INFO:@sw_id:0
TSN-INFO:!sw->sw_id:0
TSN-INFO:ptp.dst_macffffffff: 0:6:6:0:0:4
TSN-INFO:@sw_id:1
TSN-INFO:!sw->sw_id:0
TSN-INFO:!sw->sw_id:1
TSN-INFO:@sw_id:2
TSN-INFO:!sw->sw_id:0
TSN-INFO:!sw->sw_id:1
TSN-INFO:!sw->sw_id:2
send success!
TSN-INFO:@sw_id:5
TSN-INFO:!sw->sw_id:0
TSN-INFO:!sw->sw_id:1
TSN-INFO:!sw->sw_id:2
TSN-INFO:!sw->sw_id:3
TSN-INFO:!sw->sw_id:4
TSN-INFO:!sw->sw_id:5
TSN-INFO:@sw_id:0
TSN-INFO:!sw->sw_id:0
TSN-INFO:ptp.dst_macffffffff: 0:6:6:0:0:5
TSN-INFO:@sw_id:1
TSN-INFO:!sw->sw_id:0
TSN-INFO:!sw->sw_id:1
TSN-INFO:@sw_id:2
TSN-INFO:!sw->sw_id:0
TSN-INFO:!sw->sw_id:1
TSN-INFO:!sw->sw_id:2
send success!
root@hnxs-VirtualBox: /mnt/bd/tsn_cnc-2019.9.16/lib/libsouthio#

```

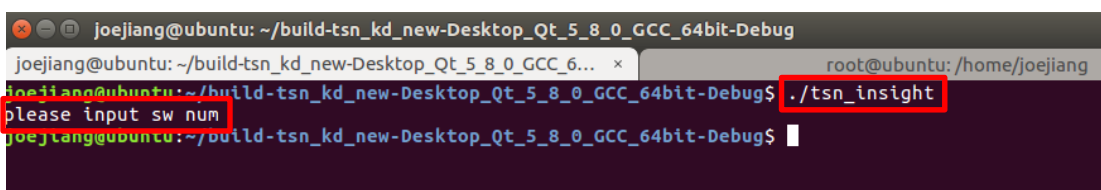
图27 执行“./tsn_CNC all”运行结果

- 执行./tsn_CNC sw_id, sw_id 表示交换机的编号,对单个交换机默认配置。例如./tsn_CNC 0 表示对交换机 0 进行默认配置。
- 执行./tsn_CNC sw_id direction tb_B tb_rate time_slot, 对单个交换机进行个性化配置, 其中 sw_id 代表交换机编号, direction 代表方向, tb_B 代表令牌桶的桶深, tb_rate 代表令牌桶速率, time_slot 代表时间槽大小, 其中时间槽的大小计算方式是 2 的 time_slot 次方纳秒。

4.2 TSN_insight 的使用

按照 3.5 的方式将 TSN_insight 与流量分析器进行连接, 在 TSN_insight 的 linux 系统下通过管理网口登录到 openbox_s4 运行界面, 登录方式可参考附录三。

执行“./tsn_insight”会出现如下提示: “please input sw num”, 其中 num 代表交换机的数量。



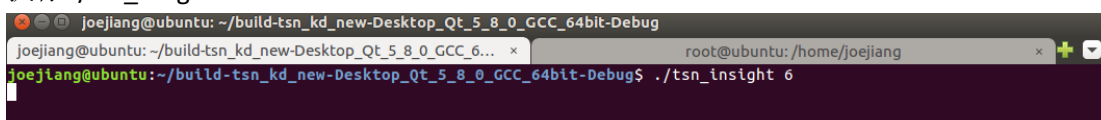
```

joejiang@ubuntu: ~/build-tsn_kd_new-Desktop_Qt_5_8_0_GCC_64bit-Debug
joejiang@ubuntu: ~/build-tsn_kd_new-Desktop_Qt_5_8_0_GCC_64bit-Debug$ ./tsn_insight
please input sw num
joejiang@ubuntu: ~/build-tsn_kd_new-Desktop_Qt_5_8_0_GCC_64bit-Debug$

```

图28 执行“./tsn_insight”运行结果

执行“./tsn_insight 6”



```

joejiang@ubuntu: ~/build-tsn_kd_new-Desktop_Qt_5_8_0_GCC_64bit-Debug
joejiang@ubuntu: ~/build-tsn_kd_new-Desktop_Qt_5_8_0_GCC_64bit-Debug$ ./tsn_insight 6

```

图29 执行“./tsn_insight 6”运行终端

出现以下界面, 表示 tsn_insight 启动成功

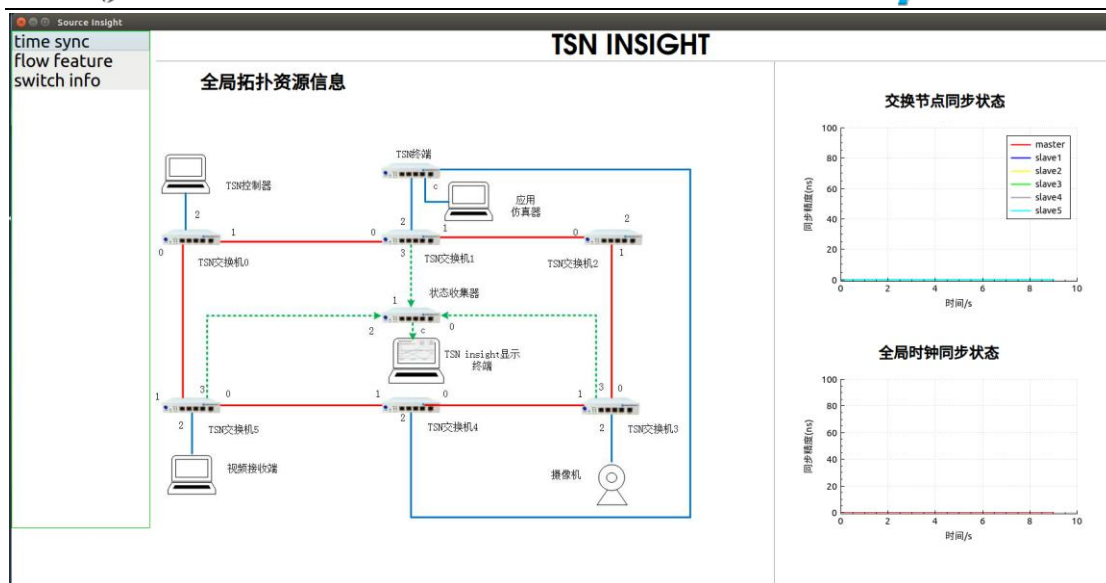


图30 执行“./tsn_insight 6”运行结果

在 TSN_insight 的 linux 系统下通过管理网口登录到 openbox_s4 界面，用于控制流量分析器（状态收集器/tsn_na），登录方式可参考附录三。

在/mnt/目录下有如下内容

```
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~# cd /mnt/
root@HNXS20:/mnt# ls
BOOT.bin                               openbox
System Volume Information               readme.txt
controller_info                       ser_uu
devicetree.dtb                       topology_info
flow_info                             tsn_7.11
kd                                    tsn_8.9
root@HNXS20:/mnt#
```

图31 执行“./tsn_na”

运行步骤

- 先在 tsn_insight 终端运行./tsn_insight 6
- 再在 tsn_na 终端运行./tsn_na

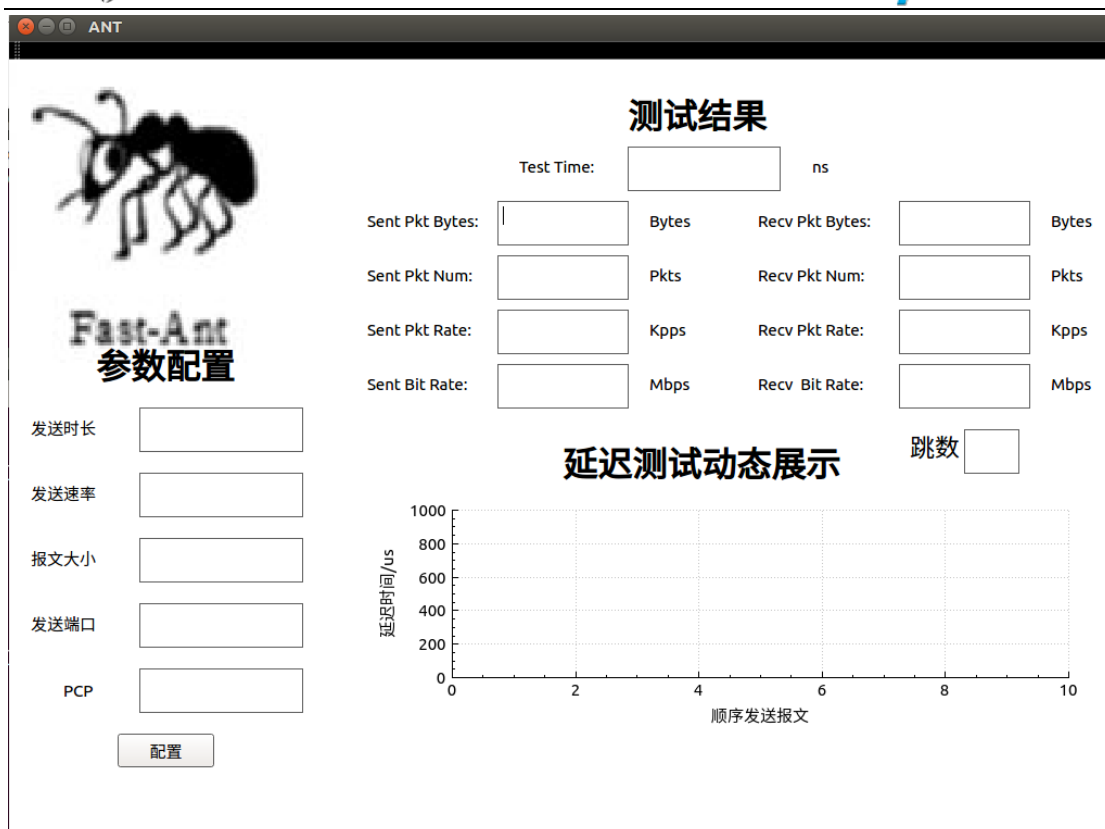
4.3 测试仪控制终端（应用仿真器）的使用

按照 3.6 节的方式将测试仪控制终端（应用仿真器）与 TSN 终端（测试仪）连接。在测试仪控制终端的 Linux 设备中把文件放在同一目录下，所需文件如下 ANT 和 ant.jpg。



图32 码云上的文件

- 步骤一：在测试仪控制终端的 Linux 系统下找到该目录并执行./ANT 可以看到运行界面为下图，则表示可以运行。



测试结果

Test Time: ns

Sent Pkt Bytes:	<input type="text"/>	Bytes	Recv Pkt Bytes:	<input type="text"/>	Bytes
Sent Pkt Num:	<input type="text"/>	Pkts	Recv Pkt Num:	<input type="text"/>	Pkts
Sent Pkt Rate:	<input type="text"/>	Kpps	Recv Pkt Rate:	<input type="text"/>	Kpps
Sent Bit Rate:	<input type="text"/>	Mbps	Recv Bit Rate:	<input type="text"/>	Mbps

Fast-Ant 参数配置

发送时长:

发送速率:

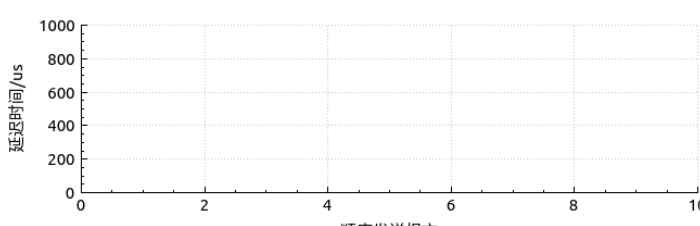
报文大小:

发送端口:

PCP:

延迟测试动态展示 跳数

延迟时间/us



顺序发送报文

图33 测试仪控制终端的界面

- 步骤二：在测试仪控制终端（应用仿真器）的 linux 系统下通过管理网口登录到 openbox_s4 界面，用于 TSN 测试仪，登录方式可参考附录三。
- 登录后进入到/mnt/目录下，目录的内容如下图所示

```

root@HNXS17:~# cd /mnt/
root@HNXS17:/mnt# ls
BOOT.bin                               latency_out                             uImage
System Volume Information               ljs_ant                                upload.sh
ant                                     openbox                                 uramdisk.image.gz
ant_old_gongsi                           sshpass-1.06
devicetree.dtb                           statistic_out
root@HNXS17:/mnt#

```

图34 TSN 测试仪运行界面

- 步骤三：需要修改 upload.sh 文件，目的是把 openbox_s4 中的 latency_out 和 statistic_out 文件传输到 linux 系统中。

```

root@ubuntu: ~
root@ubuntu: ~
joejiang@ubuntu: ~/build-ANT-Desktop_Qt_5_8_0_GCC_64bit-Debug
#! /bin/bash
#upload.sh
#password = 970904
#user = joejiang
#ip = 192.168.1.110

cd /mnt/sshpass-1.06
./sshpass -p 970904 scp /mnt/latency_out joejiang@192.168.1.30:/home/joejiang/build-ANT-Desktop_Qt_5_8_0_GCC_64bit-Debug
./sshpass -p 970904 scp /mnt/statistic_out joejiang@192.168.1.30:/home/joejiang/build-ANT-Desktop_Qt_5_8_0_GCC_64bit-Debug

```

图35 修改 opload.sh 文件

需要修改的内容：

- 其中 970904 是 Linux 机器的密码，
- [Joejiang@192.168.1.30:/home/joejiang/...](#)，分别修改 linux 中的用户名，IP，目录，其中

目录要与 ANT 和 ant.jpg 在同一目录下。

- 步骤四：在测试仪控制终端进行发包
 - 在 Linux 终端中运行 ./ANT
 - 配置响应的参数

发送时长	<input type="text" value="1"/>
发送速率	<input type="text" value="1"/>
报文大小	<input type="text" value="128"/>
发送端口	<input type="text" value="3"/>
PCP	<input type="text" value="7"/>
<input type="button" value="配置"/>	

图36 测试仪控制终端配置参数

- 点击配置按钮。
- 界面出现如下，输出经过的跳数

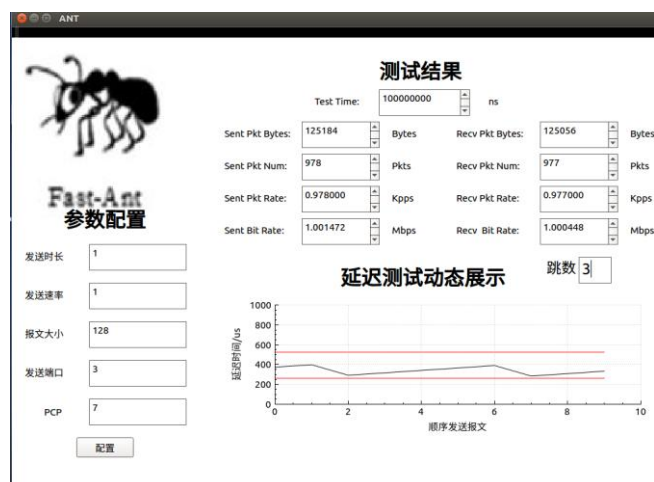


图37 测试仪控制终端显示界面

4.4 摄像头的使用

按照 3.7 节的方式将摄像头、视频接收端与 TSN 网络进行连接，由于不同的摄像头、视频接收端都有不同的 MAC 地址，因此需要修改 TSN_CNC 的配置文件，具体配置的方法如下：

- 步骤一：在 TSN_CNC 的 Linux 系统下打开 topology_info 文件，如下图：

```

}
    sw_id:3
    sync_type:slave
    host_id:3
    sw_mac:00:06:06:00:00:03
    host_mac:66:55:44:33:22:11
    next_port:1
    prev_port:0
}

    sw_id:4
    sync_type:slave
    host_id:4
    sw_mac:00:06:06:00:00:04
    host_mac:00:00:00:00:00:00
    next_port:1
    prev_port:0
}

    sw_id:5
    sync_type:slave
    host_id:5
    sw_mac:00:06:06:00:00:05
    host_mac:f8:4d:fc:77:21:d9
    next_port:1
    prev_port:0
}

```

图38 TSN_CNC 中的 topology_inso 文件

- 步骤二：修改拓扑文件中 sw_id 为 3 的 host_mac 地址，改为摄像头的 mac 地址；修改拓扑文件中 sw_id 为 5 的 host_mac 地址，改为视频接收端的 mac 地址。
- 步骤三：在 TSN_CNC 重新对 TSN 网络进行配置即可。

附录一：文件拷贝进 openbox_s4 的方法

- 步骤一：将 linux 系统设备将 ip 地址设置与 openbox_s4 处于同一网段，具体操作可查看附录四。
- 步骤二：在 linux 系统中使用 scp 进行拷贝文件到 openbox_s4 中。

```
scp BOOT.bin root@192.168.1.18:/mnt/
```

图39 拷贝命令

其中 BOOT.bin 为需要拷贝的文件，root 表示 openbox 中的用户名，192.168.1.18 表示 openbox 中的管理网口的 IP，管理网口的查询方式可参考附录四，/mnt/为拷贝到 openbox 中的/mnt/目录下。

- 步骤三：输入密码“123123”

附录二：使用串口登录 openbox_s4 运行界面的方法

1) 将串口线插入 PC 端以及 openbox_s4 设备的串口接口，然后打开计算机的设备管理器，如下图：

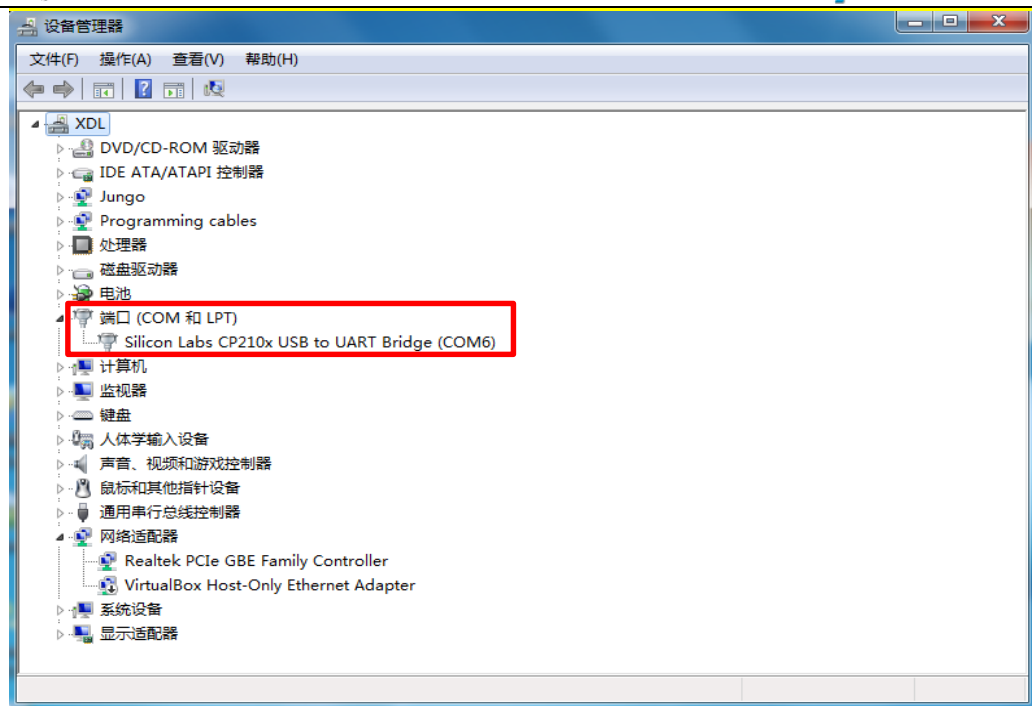


图40 查看设备管理器

2) 点开 putty 软件，如下图：

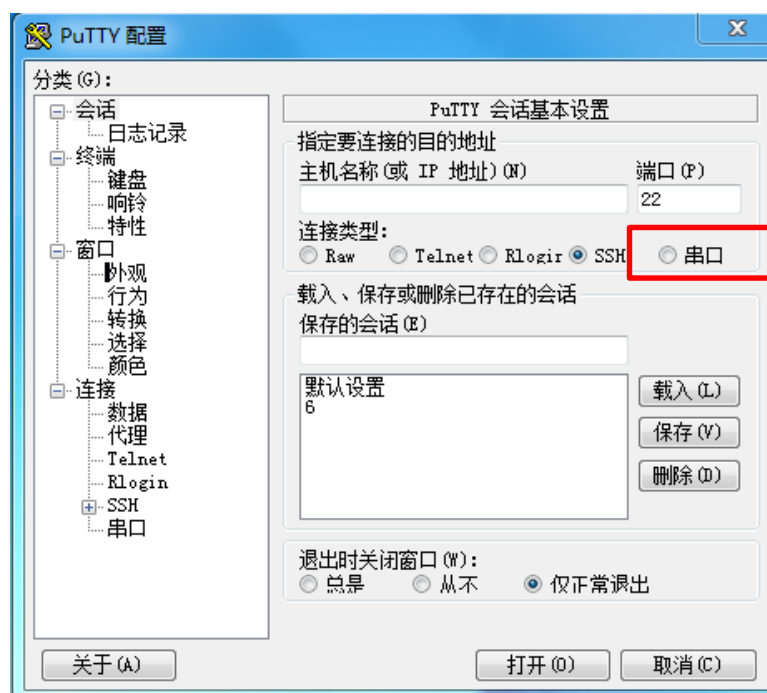


图41 Putty 上选择使用串口登录

3) 点击串口（如上图红框）选项，然后出现下图，并将串行口改为与图一中设备管理器中的端口名称一致（例：图一中为 com6，则下图应改为 com6）；将速度改为 115200：

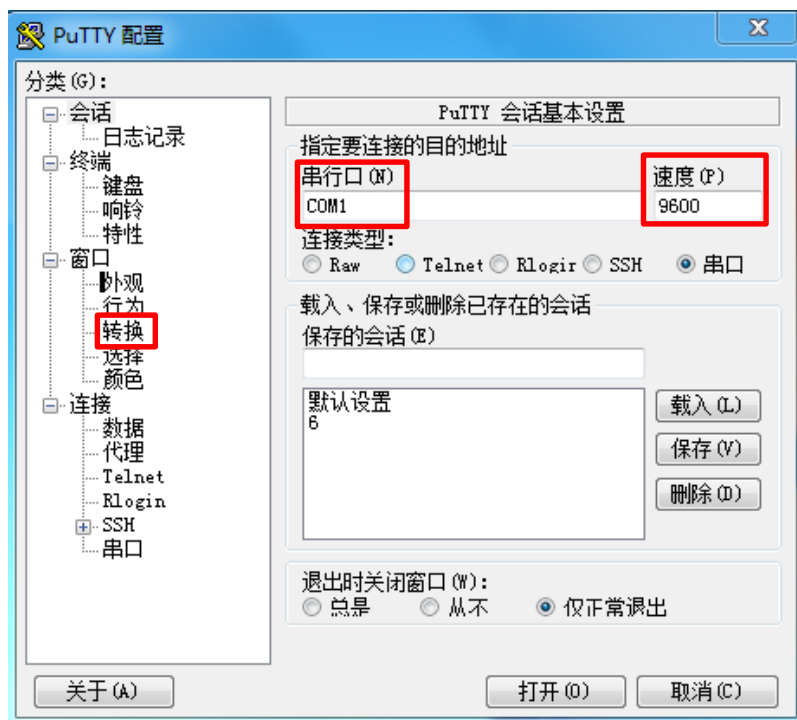


图42 配置串口信息

4) 然后点击上图左框内的“转换”，出现下图：然后点击“假定接收数据的字符集”选择“UTF-8”，点击打开即可。

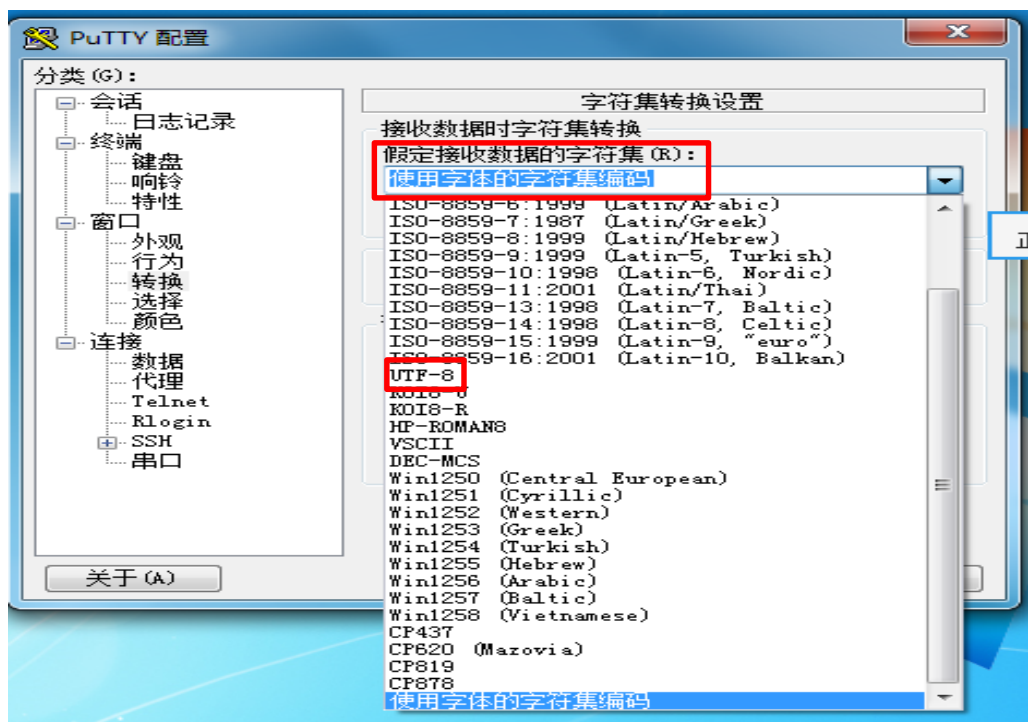


图43 选择字符集

附录三：使用管理网口登录 openbox_s4 运行界面的方法

1 windows 系统下登录 openbox_s4

- 1) 将 PC 端的 ipv4 地址修改与 openbox_s4 的管理网口 IP 地址在同一网段下（如何查看 openbox_s4 的 IP 地址参考附录四），例如：openbox_s4 的管理网口 IP 地址为 192.168.1.18，那么将 PC 端的 ipv4 地址修改为同一网段的 192.168.1.98，如下图所示：

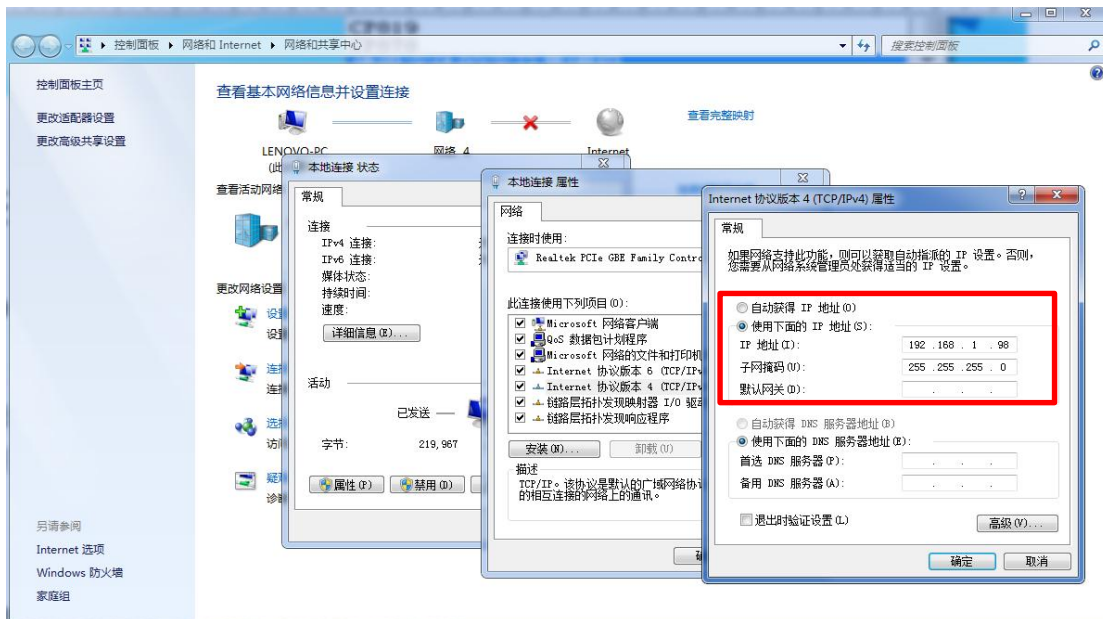


图44 修改本地设备 ip 地址

- 2) 打开 putty 软件，并在下图中的“主机名称（或 IP 地址）”区域输出 openbox_s4 的 ip 地址，例如上面提到的 192.168.1.18；然后点击打开即可登录到 openbox_s4 的运行界面上。

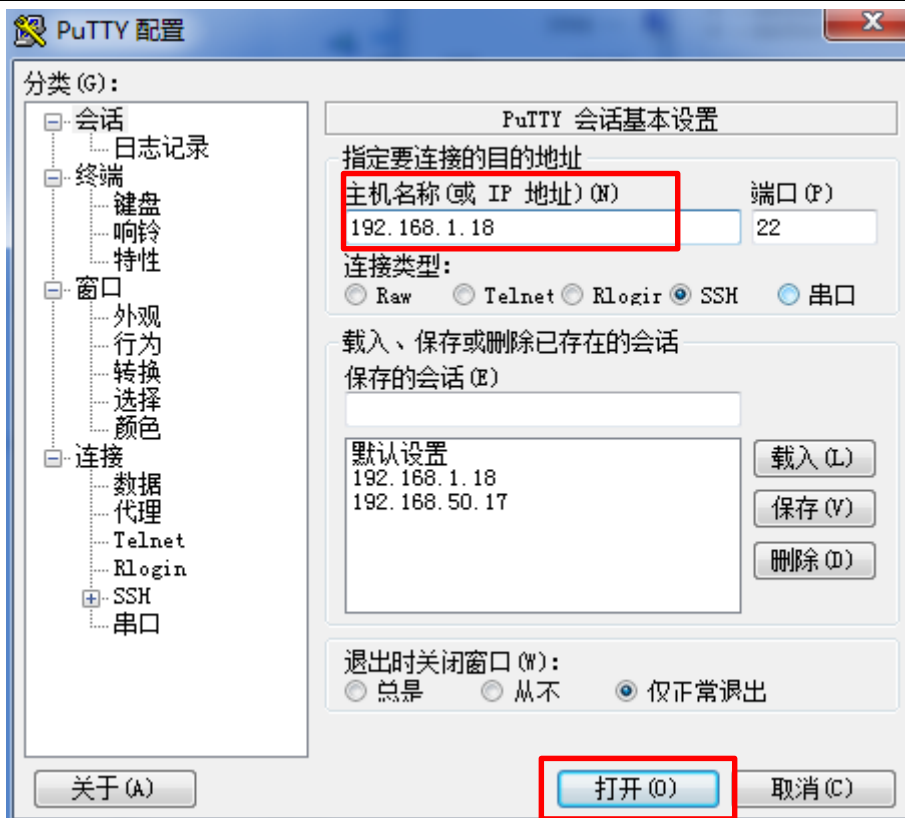


图45 登录 openbox_s4 的 ip 地址

2 linux 系统下登录 openbox_s4

- 步骤一：执行一下命令，必须是在 root 目录下执行

```
root@ubuntu:~# ssh root@192.168.1.18
```

图46 执行登录命令

- 步骤二：其中 192.168.1.18 为管理网口的 IP，出现输入密码的界面

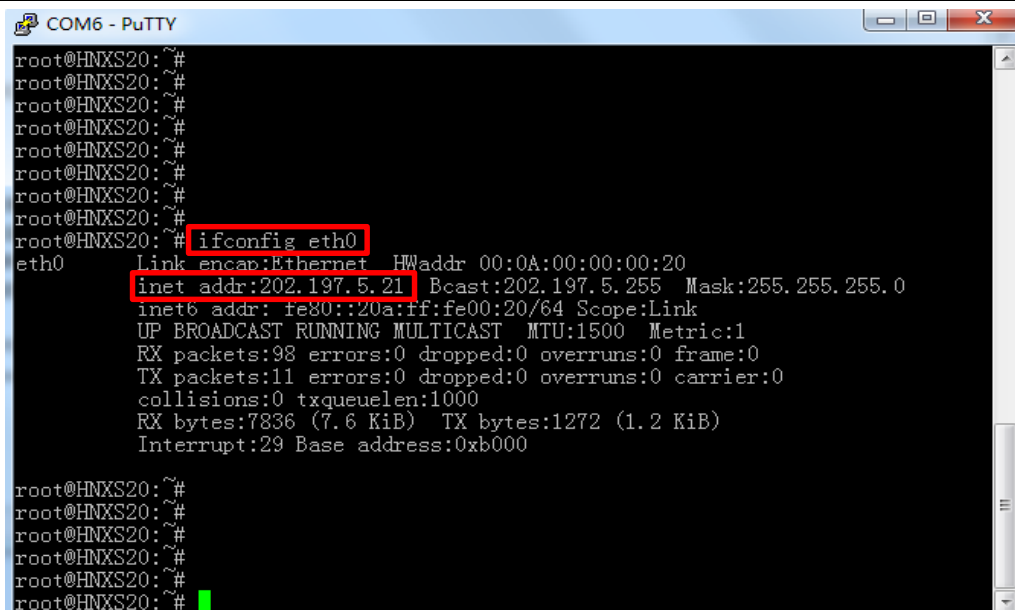
```
root@ubuntu:~# ssh root@192.168.1.18
root@192.168.1.18's password:
```

图47 输入密码

- 步骤三：输入 123123 即可。

附录四：查看 openbox_s4 的 ip 地址

- 步骤一：使用串口登录到 openbox_s4 的运行界面上，具体查看附录一。



```

COM6 - PuTTY
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0A:00:00:00:20
          inet addr:202.197.5.21  Bcast:202.197.5.255  Mask:255.255.255.0
          inet6 addr: fe80::20a:ff:fe00:20/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:98 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7836 (7.6 KiB)  TX bytes:1272 (1.2 KiB)
          Interrupt:29 Base address:0xb000

root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#
root@HNXS20:~#

```

图48 查看 openbox_s4 管理网口的 ip 地址

- 步骤二：输入命令“ifconfig eth0”即可查看管理网口的 IP 地址，例如上图中的 IP 地址为 202.197.5.21。

附录五：TSNLight 源码使用方式

上传 TSNLight 源码在“openTSN/src/软件代码/TSNLight/”文件夹下，用户需要拷贝到安装有交叉编译环境中的 linux 设备中进行编译。其中拓扑分为环形拓扑和星型拓扑，应用分为 TSN_CNC 和 TSN_NA，在编译时带入不同的参数分别对不同的拓扑和不同的应用进行编译。

- 对环形拓扑、TSN_CNC 应用进行编译
执行 `make TOPOLOGY_TYPE=0 APP_TYPE=CNC`
 - 对环形拓扑、TSN_NA 应用进行编译
执行 `make TOPOLOGY_TYPE=0 APP_TYPE=NA`
 - 对星型拓扑、TSN_CNC 应用进行编译
执行 `make TOPOLOGY_TYPE=1 APP_TYPE=CNC`
 - 对星型拓扑、TSN_NA 应用进行编译
执行 `make TOPOLOGY_TYPE=1 APP_TYPE=NA`
- 编译后生成 `tsn_controller` 可执行文件。

附录六：配置说明

对 tsn 网络配置形式包含命令行的形式和配置文件的形式，其中命令行的形式是在执行 `tsn_cnc` 应用时，在后面携带的参数，参数包含时间槽的大小、令牌桶的桶深和速率（详细配置信息参考 4.1 节 TSN_CNC 的使用）；配置文件包含 `topology_info` 和 `flow_info` 文件。

其中 `topology_info` 文件保存网络的拓扑信息，包含交换机的编号和 mac 地址等，用户可以根据需要需求，组织自己的网络，然后把网络信息填充到该文件中，然后由控制器生成配置信息配置到网络中；`flow_info` 保存的是当前的流特征信息，包含 BE、RC 和 TSN 流。

`topology_info` 文件内容如下所示

```

root@ubuntu: /mnt/hgfs/tsn_cnc-B-flowtable/lib/libsouthio
{
  sw_id:0
  sync_type:master
  host_id:0
  sw_mac:00:06:06:00:00:00
  host_mac:01:02:03:04:05:06
  port1:00:06:06:00:00:01
  port1:00:06:06:00:00:02
  port1:00:06:06:00:00:03
  port1:00:06:06:00:00:04
  port1:00:06:06:00:00:05
  port2:01:02:03:04:05:06
}

{
  sw_id:1
  sync_type:slave
  host_id:1
  sw_mac:00:06:06:00:00:01
  host_mac:f8:4d:fc:77:21:d9
  port0:01:02:03:04:05:06
  port0:00:06:06:00:00:00
  port1:00:06:06:00:00:02
  port2:00:06:06:00:00:03
  port3:00:06:06:00:00:04
  port3:00:06:06:00:00:05
  port1:11:22:33:44:55:66
  port1:12:22:33:44:55:66
  port1:f8:4d:fc:77:21:d9
  port3:10:c3:7b:23:0d:9e
}

{
  sw_id:2
  sync_type:slave
  host_id:2
  sw_mac:00:06:06:00:00:02
  host_mac:12:22:33:44:55:66
  port0:01:02:03:04:05:06
}
1,1 顶端

```

图49 topology_info 文件内容

其中 sw_id 表示交换机的 ID，从 0 开始，依次进行编号；sync_type 代表设备的类型，用于主从设备的配置，包含 master 和 slave；host_id 代表与该交换机连接的主机的 ID 号，一般也是从 0 开始表示；sw_mac 代表交换机的 mac 地址；host_mac 代表交换机连接主机的 mac 地址；port*代表需要从*号端口转发的目的 mac 地址为后面参数的报文，例如上图红色框图中，如果该交换机接收到目的 mac 地址为 00:06:06:00:00:01 的报文，则从 1 号端口转发，如果该端口没有需要转发的报文，则不需要填写，如果该端口需要转发多条报文，则需要全部列举出来。

flow_info 文件内容如下所示（目前流特征用于界面展示，不用于配置）：

```

root@ubuntu: /mnt/hgfs/tsn_controller-9.16-xiugai/tsn_co
{
  type:tsn
  flow_id:0
  src_mac:12:22:33:44:55:66
  dst_mac:00:00:00:00:00:00
  priority:7
  interval:1000
  pkt_num:1
  pkt_size:1000
  latency: 100000
}

{
  type:rc
  flow_id:1
  src_mac:8c:16:45:49:27:b0
  dst_mac:12:22:33:44:55:66
  priority:4
  bandwidth:10
}
1,1 全部

```

图50 topology_info 文件内容

其中 type 字段标识流量的类型，包含 tsn、rc 和 be 三种流量类型；flow_id 用于唯一标识流的 ID，并且对三种流量进行统一标识，一般从 0 开始；src_mac 字段表示报文的源 mac 地址；dst_mac 字段表示报文的目的 mac 地址；priority 字段表示优先级，其中 tsn 流的优先级为 6 和 7，rc 优先级为 3、4 和 5，be 流的优先级为 0、1 和 2；interval 字段表示流量发送周期；pkt_num 表示每个周期内发送的报文个数；pkt_size 表示每个报文的尺寸；latency 字段表示 tsn 流从发送到接收延迟的最大时间；bandwidth 字段表示 rc 流预约的带宽。