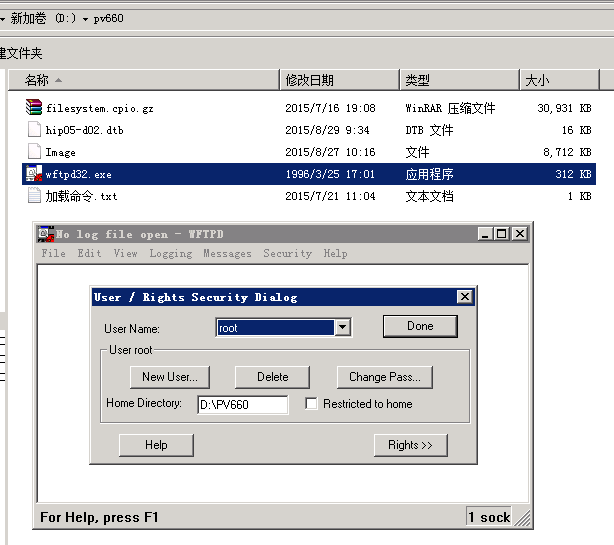
ODP使用手册

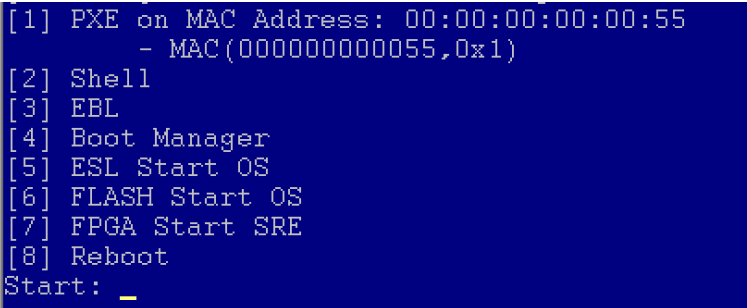
# 环境搭建，这里以D02单板为例

第一步：连接串口，串口波特率：115200 数据位：8 停止位：1 校验位：0

第二步：保证D02单板的网口和加载电脑在同一网段，且网络是能互通。

第三步：把内核文件、DTB文件、文件系统放入PC机中，并打开wftpd32.exe，设置好用户名和密码，指定文件加载路径。



第四步：当串口出现下面界面时

依据提示输出下面的脚本：

3

ifconfig -s eth0 192.168.1.20 255.255.255.0 192.168.1.1

provision 192.168.1.151 -u root -p huawei -f Image -a 0x80000

provision 192.168.1.151 -u root -p huawei -f hip05-d02.dtb -a 0x6000000

provision 192.168.1.151 -u root -p huawei -f filesystem.cpio.gz -a 0x8000000

exit

5

ifconfig eth1 192.168.1.20

其中黑色IP为前面所提示FTP服务器的IP，标红部分别为内核文件、DTB文件、文件系统。

单板加载成功后，配置好IP后，会有下面提示：



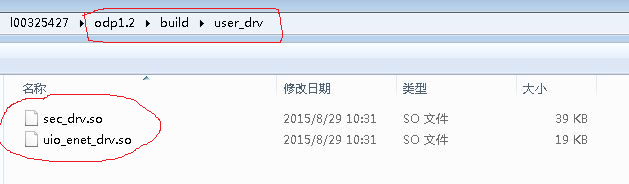
这时可以确认单板已加载成功，并且网络已互通，这时可以使用SSH连接单板了。

# 加载ODP网卡驱动

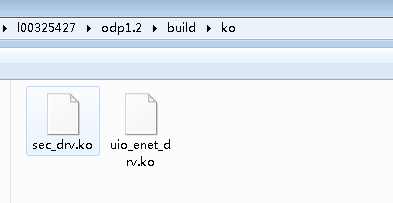
**第一步：**在单板的usr目录下，创建lib64/odp目录



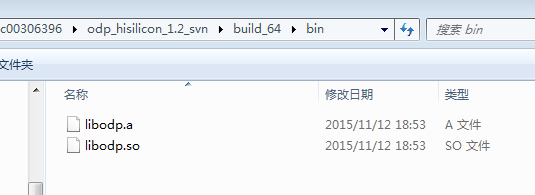
**第二步：** 把编译机的odp1.2/build/user\_drv 目录下的文件上传到单板的上面目录下



**第三步：**把odp1.2/build/ko 目录下的文件上传到单板，目录随便



**第四步：**把编译机的odp1.2/build/bin 目录下的libodp.so文件上传到单板的/usr/lib目录下

****

**第五步：**执行 insmod uio\_enet\_drv.ko 插入UIO网口驱动

小技巧：执行odp1.2/build/copy.sh，该脚本会把相应的文件拷贝到对应的目录中，具体目录可以查看脚本。

# 编译ODP应用程序

包含ODP应用程序编译，当前是在编译的时候需要显示的指定libodp.so，这样才能把相应的符号表编译到应用程序中，在应用程序中才能正确的调用到ODP当中的API接口。下面就具例应用程序的makefile编写。

LIBS := -L$(ROOT)/build/bin/ -lODP -lpthread -ldl -lrt -lm

显示指定libODP.so的路径和SO名称。

$(APP\_OUT) : $(OBJ\_FILE)

$(ECHO) "LD " $@;\

$(CC) $(LIBS) $(CCFLAGS) -o $(ROOT)/build/app/$@ $^

生成应用程序规则，编译后存放路径为：odp1.2\build\app 目录下，生成的名字规则为应用程序的 目录名\_app 。

具体应用程序举例可以参照 odp1.2\example\ 下面的应用程序。

小技巧：直接在ODP的工程 odp1.2\example\ 下新建自已的应用程序目录，并把\odp1.2\example\classifier 下的 Makefile 、 build.sh 两个文件拷入自已新的应用程序目录中，然后在 odp1.2\build\app\_build.sh 中，加入相应的 $ROOT\_DIR/example/新建目录名/build.sh 即可。然后在build目录中调用 app\_build.sh 即会编译相应的应用程序了。

# 运行ODP应用程序

在运行ODP应用程序之前，需要在单板上执行如下几条脚本，创建大页内存。

mkdir /mnt/huge

mount none /mnt/huge -t hugetlbfs

echo 200 > /proc/sys/vm/nr\_hugepages

在这些脚本执行成为功后，即可以运行ODP相关的应用程序了。

Demo程序执行命令：

64位ODP层2转发：

./l2fwd\_app -i pktio\_15 -c 1 -m 0

64位ODP加解密：

./sec\_app -i pktio\_15 -c 1 -m 0 -s 0

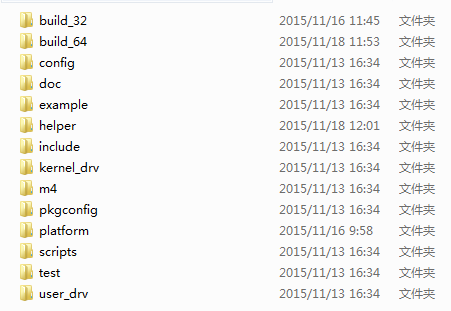
32位ODP层2转发：

./l2fwd\_app32 -i pktio\_15 -c 1 -m 0

32位ODP层2转发：

./sec\_app32 -i pktio\_15 -c 1 -m 0 -s 0

# odp工程文件目录介绍



build\_32：ODP工程32位编译目录；

build\_64：ODP工程64位编译目录；

config：doc：helper：m4：pkgconfig：test：暂时没用到，可不用关注；

example：demo程序存放路径，在创建ODP的应用程序时参考；

include：ODP对外API接口申明文件；

kernel\_drv：内核态驱动，需要与使用的内核进行联编；

kernel：存放环境搭建时用到的image，DTS，文件系统；

platform：ODP平台源文件

user\_drv：用户态驱动源文件；