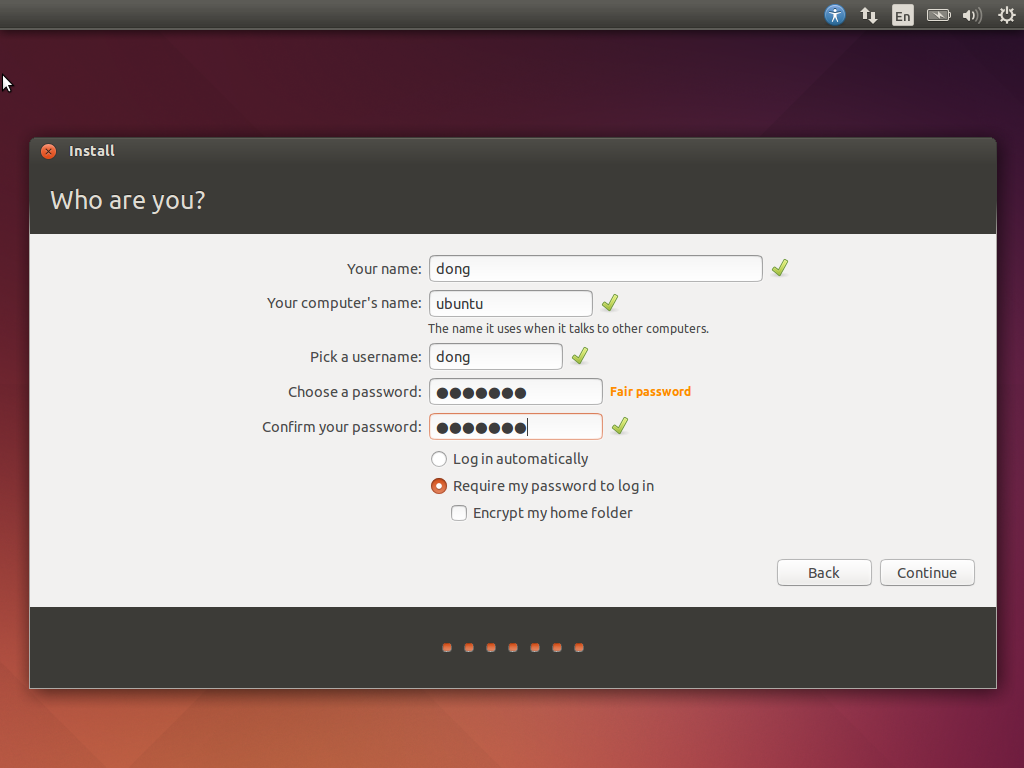
# 1环境准备

## 1.1系统安装

虚拟机VirtualBox-4.3.22-98236-Win.exe，安装时除了没有使用C盘和没有创建快速启动按钮，其他步骤一路Next。

Linux发行版本ubuntu-14.04.1-desktop-i386.iso（32位），安装时语言选择了英文

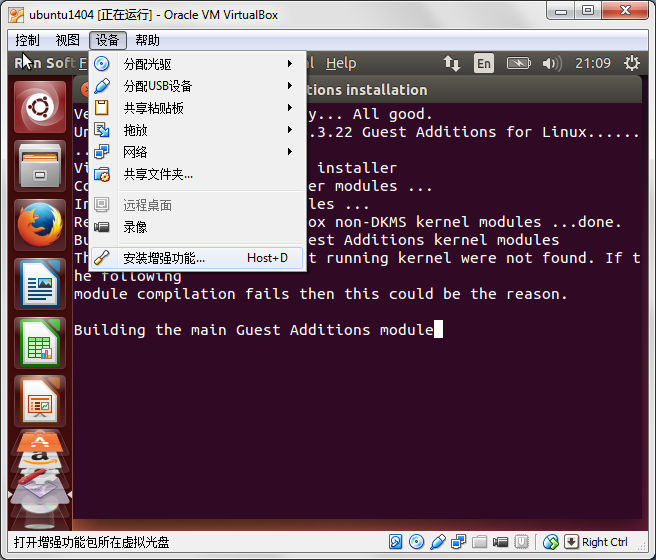
口令是1111111



安装好后如何解决分辨率问题

安装VirtualBox的增强功能即可（参考自<http://blog.sina.com.cn/s/blog_489988100102ux6e.html>）

VirtualBox->设备->安装增强功能



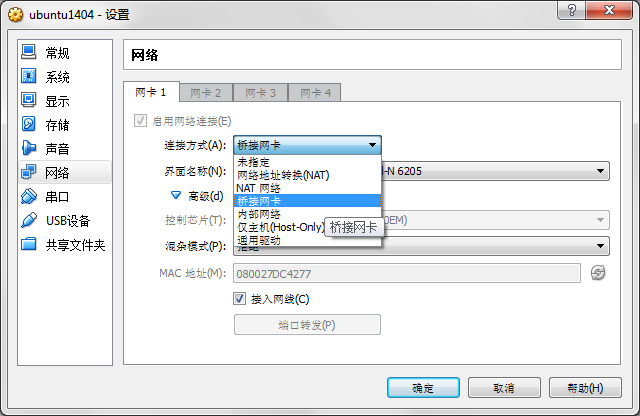
安装增强模式后，也就支持了全屏视图。

## 1.2 Samba, ssh, Telnet

**在Ubuntu的桌面环境下通过Ctrl+alt+t调出命令行。**

### 1.2.1网卡模式

虚拟机网卡的默认方式为Nat方式。为了让本局域网内其他设备（包括安装VirtualBox的这台PC）方便通过网络连接这台虚拟Ubuntu，设备的网卡选择为桥接模式。设置以后，Ubuntu重启后生效。



### 1.2.2 Samba

Samba是一个可以用来进行Linux和windows之间的文件夹共享的工具。安装在Linux上。

安装：

sudo apt-get install samba， 安装samba工具

sudo apt-get install cifs-utils，安装Cifs工具，Cifs是一个跨系统的文件互访协议，如果是Linux访问Windows共享出来的文件夹，就必须要这个了。

两个安装都跟着后面的提示按Y就可以了。

网上还有说可能需要sudo apt-get install samba-common，但是进行这个安装时，提示已经安装了。

找到/etc/samba/smb.conf,右键Gedit打开，添加如下一段

[ubuntu\_share] ------在共享网络上显示的名字

comment = ubuntu1404

browseable = yes

path = / ----从根目录共享

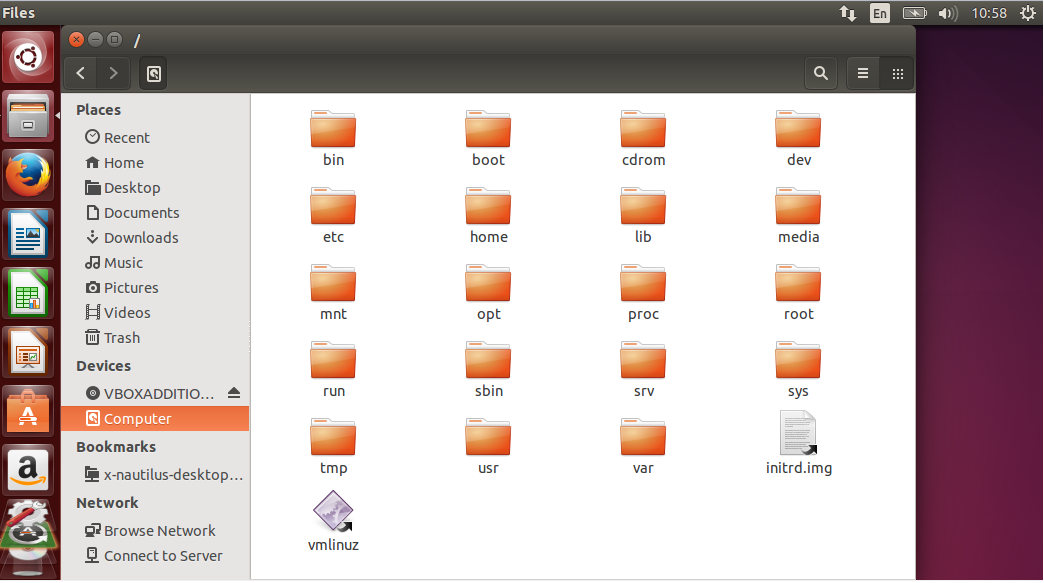
guest ok = yes ----共享出去

read only = no ----非只读

create mask = 777 ---guest创建文件后文件Mod

directory mask = 777 -----？

可以通过files->computer找到etc目录，进而找到smb.conf



保存之前，需要先用shell将该文件权限打开，sudo chmod 777 /etc/samba/smb.conf。

还需要将Gedit备份文件的功能关掉gedit->edit->preferences->edit，在这个tap中的file saving选项中，不选择生成备份文件那一项。（或者将/etc/samba的权限也设置为777应该也可以）

命令行里sudo /etc/init.d/smbd restart(或者sudo service smbd restart)，重启Samba就可以在Window访问Ubuntu共享出去的文件夹了。

sudo service smbd resart

如果windows需要编辑哪个目录或文件，要在shell里将其Mod设置为777.

**以上为windows访问linux共享。下面是Linux访问Windows共享。**

在Windows上设置共享目录，比如起名叫做*2linux*

事先在/mnt创建一个smb目录

mkdir smb

然后使用下面的命令即可

**sudo mount -t cifs -o username=administrator,password=123,nounix,noserverino //192.168.1.102/2linux /mnt/smb**

username和password按实际情况填写，关于这个也不是特别明白是怎么回事。如果windows那边有登录用户名密码，并且Windows设置共享到所有，使用这个用户名密码是肯定可以的。

卸载使用sudo umount /mnt/smb

### 1.2.3 ssh

安装一下即可

sudo apt-get install openssh-server

sudo /etc/init.d/ssh start (启动ssh) ,重启可以sudo service ssh restart

使用Ubuntu已有的用户名和Pwd即可以windows使用secureCRT等终端工具访问

### 1.2.4 telnet

telnet和ssh作用一样，就是提供一个远程全程命令行。

安装Xinetd 服务器，该服务器集成FTP、Telnet等多种服务。

$sudo apt-get install xinetd telnetd

配置文件是/etc/xinetd.conf ，按默认的就行，如下是默认的内容：

# Simple configuration file for xinetd

#

# Some defaults, and include /etc/xinetd.d/

defaults

{

# Please note that you need a log\_type line to be able to use log\_on\_success

# and log\_on\_failure. The default is the following :

# log\_type = SYSLOG daemon info

}

includedir /etc/xinetd.d

上面最后一行是指定了include的路径，在该路径下添加名为Telnet的文件，配置TELNET服务

service telnet

{

disable = no

flags = REUSE

socket\_type = stream

wait = no

user = root

server = /usr/sbin/in.telnetd

log\_on\_failure += USERID

}

启动或重启XINETD服务就可以了

sudo /etc/init.d/xinetd start（sudo /etc/init.d/xinetd restart）

可以用如下命令查看服务是否开启$ps -e | grep xinetd

### 1.2.5创建删除新用户

sudo useradd -m nuser, nuser 是用户名，-m是为其在home下创建一个目录

Sudo passwd nuser， 修改该用户密码

userdel -r nuser ， 删除nuser，-r会一起删除他的home目录

### 1.2.6建设ftp服务器

Apt 安装vsftp

sudo apt-get install vsftpd

sudo service vsftpd restart 重启成功说明安装成功

编辑/etc/vsftpd.conf进行配置。

只要把默认的文件中设置为local\_enable=YES，就可以让系统里的用户使用自己的用户名密码读自己的home文件夹了。再设置write\_enable=YES也就可以写了。（Ubuntu1804上还要设置local\_umask）

有一点绕，

chroot\_local\_user=NO

chroot\_list\_enable=NO

chroot\_list\_file=/etc/vsftpd.chroot\_list

上面chroot\_local\_user的意思是，用户可以不可以cd到他的文件夹外面去。默认貌似是可以的。

chroot\_list\_enable如果使能，就是说chroot\_list\_file里列出来用户与上一项的要求相反。chroot\_list\_file里的格式是一行一个user name。

但又有一点坑，就是如果某个用户是不能cd出去的，vsftp要求该用户的根文件夹不可写，否则登录不成功。而根文件夹又不方便设置为只读。

在下面这个配置下，atf5guser可以share出自己的用户目录，在ftp上显示为/，但不能直接写/，需要事先创建好一个可读写的Users文件夹

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

关于上面记录的小坑， 一句allow\_writeable\_chroot=YES可以解决。

<https://www.liquidweb.com/kb/error-500-oops-vsftpd-refusing-to-run-with-writable-root-inside-chroot-solved/>

chroot\_local\_user=YES

#chroot\_list\_enable=YES

# (default follows)

#chroot\_list\_file=/etc/vsftpd/chroot\_list ----如果chroot\_list\_enable为YES，则一定要有这么个文件。

allow\_writeable\_chroot=YES

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

下面是配置virtual user，这个是为了让不同的人对同一个文件夹有不同权限。否则上面就可以满足要求了。

先建立一个本地用户vftpuser,工作目录为vftpsite.

useradd -d /home/vftpsite -s /sbin/nologin vftpuser

创建一个文本文件，比如account.txt，两行一个用户，分别是用户名和密码。

apt-get install db-util 安装db\_load

db\_load -T -t hash -f account.txt /etc/vsftpd/vftpuser.db 生成ftp用的用户名密码文件。

编辑/etc/pam.d/vsftpd，把其内容改为

auth required /lib/i386-linux-gnu/security/pam\_userdb.so db=/etc/vsftpd/vftpuser

account required /lib/i386-linux-gnu/security/pam\_userdb.so db=/etc/vsftpd/vftpuser

这个文件原来是存在的，把里面文件删除，或者用#号注释掉。这个so文件和网络上搜出来的位置不一上，是用find -name找出来的。那个db文件的扩展名不要写。

vsftpd.conf文件里添加

guest\_enable=YES ------使用能guest.

guest\_username=vftpuser ------所有guest map到vftpuser

pam\_service\_name=vsftpd -----------------关联/etc/pam.d/vsftpd

user\_config\_dir=/etc/vsftpd\_user\_conf -----------该文件夹是以用户名命名的文件，如s1文件里配置用户s1的读写权限

chroot\_local\_user=YES -------后面这两个貌似如果不加，所有用户都将只有可读权限。

virtual\_use\_local\_privs=YES

/etc/vsftpd\_user\_conf下的文件内容举例：

dongcao@upl1tester-OptiPlex-7010:~$ cat account.txt

Atf5GTrialSW

5Gwin

Atf5GTrialReader

5Gguest

dongcao@upl1tester-OptiPlex-7010:~$ ls /etc/vsftpd\_user\_conf/

Atf5GTrialReader Atf5GTrialSW

dongcao@upl1tester-OptiPlex-7010:~$ cat /etc/vsftpd\_user\_conf/Atf5GTrialReader

write\_enable=NO

anon\_upload\_enable=NO

dongcao@upl1tester-OptiPlex-7010:~$

如果需要增加用户，再往account。Txt里加个用户重新生成那个DB，然后再往/etc/vsftpd\_user\_conf加个同名文件就可以了。貌似那个db\_load命令是往db里增加用户，如果要删除用户可能用别的命令。但可以直接把db文件删除再用生成的一个新的。

同样会涉及到上述的文件夹操作权限的问题。本地用户的根文件夹设置成只读是不方便的。但ftp专用的就没关系了，可以将/home/vftpsite设置成只读，但里面的子目录再设置成可写。

主要参考自下面这个，有一点是里面的命令中的-不对。Copy时要注意。

<http://yuanbin.blog.51cto.com/363003/129071/>

### 1.2.7远程桌面

下面的安装和配置完成后，windows可用mstsc工具登陆该系统。（Ubuntu1404验证）

sudo apt-get install xrdp

sudo apt-get install vnc4server

sudo apt-get install xubuntu-desktop

echo "xfce4-session" >~/.xsession

sudo service xrdp restart

# 2内核编译

本部分参考自http://qkwu.net/hw/139.html

Ubuntu1404使用的内核为3.13.0

uname -a

Linux ubuntu 3.13.0-32-generic #57-Ubuntu SMP Tue Jul 15 03:51:12 UTC 2014 i686 i686 i686 GNU/Linux

到官网去下载3.17版本<https://www.kernel.org/pub/linux/kernel/v3.x/linux-3.17.8.tar.gz>

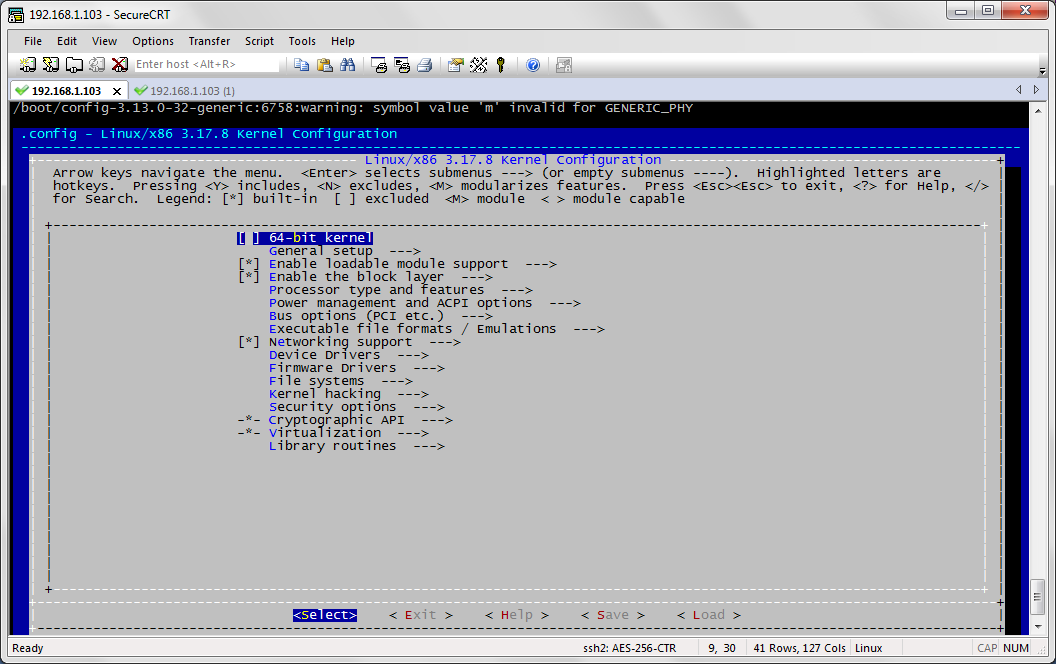
1 放到/usr/src，使用tar –zxvf解压缩tar -zxvf linux-3.17.8.tar.gz

2 切换到解压缩的目录/usr/src/linux-3.17.8。进入su

Ubuntu安装时没有指定root与口令，使用sudo passwd root来打开，这里设置好root的Pwd就可以su了。设置的pwd为123456。

3 apt-get install libncurses-dev 安装下一步需要的lib库

4 make menuconfig，生成配置文件使用默认的，直接Exit进下一个界面生成 .config



5 敲make，编译，很久很久 我了个去8、9个小时

6 make modules ，编译模块。 有Warning

7 make modules\_install  安装模块（模块拷贝到需要的目录中）

8 make install  安装内核（完成内核的安装（会复制内核至标准位置/boot..）相应的grub文件也会做更改）还是有warning

9 reboot 重启计算机，就是新内核了（启动时有错误）



再编译Moudle就会使用3.17.8的头文件与库。scull编译加载能成功，块设备（2.13.0还可以过）的又不过了。

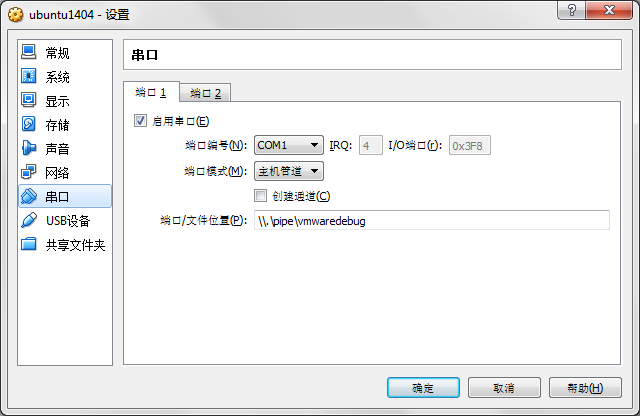
# 3 虚拟机和主机使用串口通信

## 3.1测试已安装驱动的Linux串口

LLD中对于与硬件通讯，是通过并口编程来演示的。Virtalbox提供了串口而没有提供并口。所以我们只能摸索串口编程了。

而使用virtalbox的主机管道模式，刚好方便我们在宿主机上用软件的方法去模拟虚拟机“硬件串口”读写。而如果能模拟出虚拟机的硬件写，Linux上就可以体会中断了，对不对！！

管道模式可以理解为Virtalbox把虚拟机的“硬件串口”连到了一个Windows的命名管道上。宿主机对管道的读写就分别对应了虚拟机里的写和读。



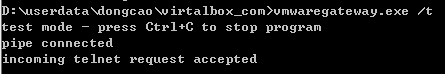
而宿主机对该管道的操作是，如果对windows编程熟悉可以写个程序来完成。这里不费这个事了，借用一个第三方的工具vmwaregateway。该工具的代码和EXE都可以在这个链接处下载。<http://www.l4ka.org/91.php>

看了下vmwaregateway的使用说明，可以在Test模式下运行，也可以注册成windows服务来完成工作。不知道什么原因的，我的Win7上注册不能成功，但好在Test模式下可以运行的很好，就不去研究为什么了。

命名管道的使用需要双方协商好一些东西，叫什么名字，谁先创建另一个再去打开等等。在这里的话vmwaregateway会创建一个叫做 [\\.\pipe\vmwaredebug](file:///\\.\pipe\vmwaredebug)的管道，所以Virtalbox只配置好这个名字，就不要再选中那个“创建通道”了。同时这种约定也意味着要先打开vmwaregateway才可以再运行虚拟机，否则Virtalbox就会抱怨说打不开那个命名管道。对于这一点，在做与串口无关的操作时，就把“启用串口”去掉，或者是把那个“创建通道”选上。

Vmwaregateway的跑起来后的工作简单点说就是，把管道再和TCP 567端串口接起来。从管道读出的数据送到TCP 567，从TCP567读出的数据再送到管道，我们读写TCP567就可以了。最简单的方式就是使用Telnet工具连接到567，就可以完成交互了，终于。需要说明的一点是Telnet只能发送Ascll字符，如果需要发送各种任意数据，只能自己写程序了。

Vmwaregateway的测试模式打开方式是通过CMD命令行打开，并指定参数/t。如果虚拟机跑起来了去连接到管道的另一端，或是用telnet工具连到Tcp567端口上，都会有提示。如下 ：



Windows XPTelnet的

Cat /proc/ioports

对应的设备是/dev/ttyS0

sudo cat /proc/tty/driver/serial 查看端口号，中断号等信息。

dong@ubuntu:~$ sudo cat /proc/tty/driver/serial

serinfo:1.0 driver revision:

0: uart:16550A port:000003F8 irq:4 tx:29 rx:27 CTS|DSR|CD

1: uart:unknown port:000002F8 irq:3

2: uart:unknown port:000003E8 irq:4

3: uart:unknown port:000002E8 irq:3

4: uart:unknown port:00000000 irq:0

。。。。。。

dong@ubuntu:~$

stty –a 查看波特率等信息

dong@ubuntu:~/drvttys$ stty -a

speed 38400 baud; rows 54; columns 197; line = 0;

intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>; eol2 = <undef>; swtch = <undef>; start = ^Q; stop = ^S; susp = ^Z; rprnt = ^R; werase = ^W; lnext = ^V; flush = ^O;

min = 1; time = 0;

-parenb -parodd cs8 -hupcl -cstopb cread -clocal -crtscts

-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl ixon -ixoff -iuclc -ixany -imaxbel -iutf8

opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0 bs0 vt0 ff0

isig icanon iexten echo echoe echok -echonl -noflsh -xcase -tostop -echoprt echoctl echoke

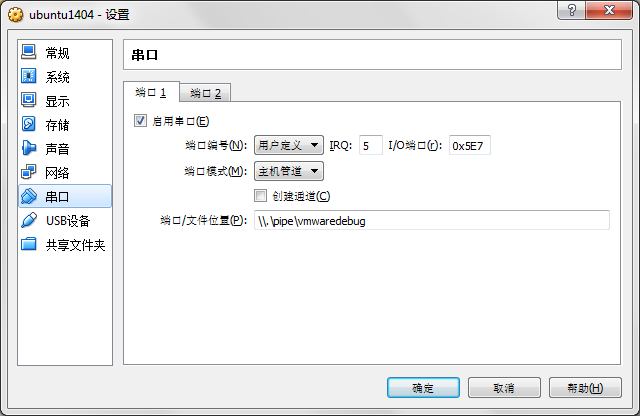
dong@ubuntu:~/drvttys$

cat /proc/interrupts

cat /proc/stat ucan grep intr

## 3.2 添加一个自定义的硬件串口

默认的硬件串口，IO端口起始编号为0x3f8，IRQ为4这种，虚拟机中的Ubuntu会将其识别出来，并用Linux默认的驱动处理。



如果把起始编号设置为其他的XX8，Linux也会将其识别出来并加载驱动。如果系统已经加载驱动了，我们就不能再对这串IO端口以及IRQ进行处理了。

如果不把起始值设置为XX8，系统就不能识别，我们写的模块也就可以操作这些IO和irq了。但发现IO的偏移并不能和网上搜出来的16550寄存器偏移对应起来，经过多次尝试，发现IO端口被自动对齐到xx8了。这个对齐是谁做的，Virtualbox还是Linux，VirtualBox的可能性大一些。

比如按上图上的0x5e7，只要在代码中申请0x5e8~0x5ef这8个IO端口，就可以按16550的寄存器操作，可以得到预期结果。16550的寄存器使用可参考下面这个文档

### [串行接口芯片16550\_百度文库](https://www.baidu.com/link?url=FsDi3YAwZdMbyM60LP-7mUG9BRThAm1ycs5YO0j3NoP3BYX2oEneWcfcyeoV5TCCE7nzDbhQ_XOukwzvhdKE7HJjmicVeWLs5wlAfaIciP3&ie=utf-8&f=3&tn=baidu&wd=%E4%B8%B2%E8%A1%8C%E6%8E%A5%E5%8F%A3%E8%8A%AF%E7%89%8716550&inputT=10055&sug=%E4%B8%B2%E8%A1%8C%E6%8E%A5%E5%8F%A3&oq=%E4%B8%B2%E8%A1%8C%E6%8E%A5&rsp=0)

http://wenku.baidu.com/link?url=FsDi3YAwZdMbyM60LP-7mUG9BRThAm1ycs5YO0j3NoP3BYX2oEneWcfcyeoV5TCCE7nzDbhQ\_XOukwzvhdKE7HJjmicVeWLs5wlAfaIciP3

波特率暂不用管，是不是传输不稳定？

Cat

Modinfo