

1 网络配置

工程源码	-ACZ702 v2.0 开发板 - 无
相关视频课程	无

章节导读

在本章节中，我们将讲解如何配置 PC 端、虚拟机、开发板三端网络，达到互通目的。在嵌入式 Linux 学习中，配置三端的 IP，使其网络互通对于开发者的工作流程有着重要的影响。

1.1 网络互通作用

文件传输：通过网络互通，我们可以快速地在这三端之间传输和共享文件。比如，你可能在 PC 端编写和修改代码，然后需要将这些代码发送到虚拟机或开发板上进行编译和测试。

远程控制：有时候，你可能需要在 PC 端远程操作和控制虚拟机或开发板进行各项任务。如果这三者之间的网络可以互通，你就可以使用诸如 SSH 等工具达到这个目的。

测试和调试：在开发的过程中，你可能会需要在 PC 端或虚拟机上进行模拟测试和调试工作。如果网络互通，那么在开发板上运行的程序就可以和 PC 端或虚拟机上的程序进行数据交换和通信，从而更加方便地进行测试和调试。

总的来说，配置 PC 端、虚拟机、开发板三端的网络互通可以极大地提高开发效率，增强工作流的灵活性，是嵌入式 Linux 开发中必不可少的环节。

1.2 网络连接方式

为了方便后续驱动与应用的开发，需要配置网络环境，目前我采用的连接方式是：

- PC 机与路由器，WIFI 连接
- 开发板与路由器，网线连接
- PC 机与虚拟机，桥接连接

前提须知：三者的网关必须相同，不然无法 ping 通！



图 1-1 硬件连接框架图

开发板硬件连接图，如下图所示，网口连接到路由器，串口连接电脑：

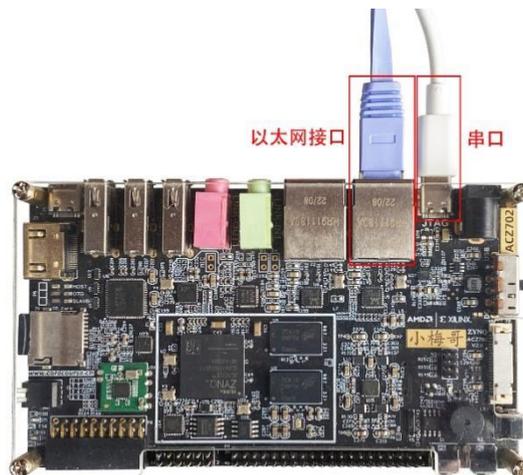


图 1-2 开发板连接图

采用上图硬件连接的优点如下：

兼容性：开发板网口直接连接到电脑可能需要对电脑的网络设置进行手动配置，包括分配 IP 地址、子网掩码、网关等，而这些设置可能需要在不同的电脑操作系统中进行不同的操作。但是，如果连接到路由器，由于路由器通常会提供 DHCP 服务，所以可以自动为开发板分配 IP 地址，大大简化了网络设置的流程。

网络共享：如果开发板直接连接到电脑，那么只能和电脑进行数据交换，而不能直接访问局域网中的其他设备或互联网（除非电脑设置为网络桥接）。但是，如果开发板连接到路由器，那么开发板就可以访问局域网中的所有设备和互联网，这对于一些需要网络功能的应用程序来说是非常有用的。

灵活性：连接到路由器可以使得开发板在任何时间都能够访问互联网或局域网中的其他设备，而无需电脑始终开机和连接。这对于一些需要长时间运行或远程访问的项目来说是非常有利的。

注意：如果你因为**缺乏路由器或者路由器网口不足**，无法将开发板连接到路由器/集线器，那么也可以**直接将开发板和电脑相连接**。

可以参考帖子：[在 VMware 中配置双网络适配器，分别桥接到电脑的两个网卡](#)

1.3 Windows 系统设置

按下 Win + R 键，这将会打开“运行”对话框，然后在对话框中输入`cmd`，最后按回车键。这样就会打开命令提示符窗口。

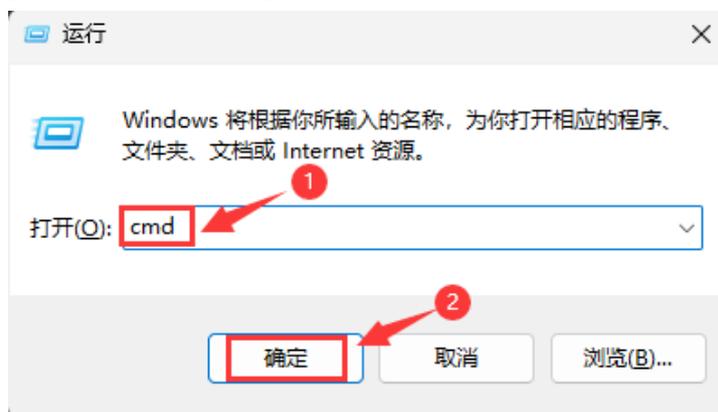


图 1-3 Windows 终端

1.3.1 查看 IP

在命令窗口中，输入：`ipconfig`，该命令主要用于显示和管理本地主机的网络接口卡（也就是网卡）的 IP 配置。



图 1-4 IP 命令

向下翻阅，找到“无线局域网适配器 WLAN”描述，记录 IPv4 地址。



图 1-5 记录 IPv4 值

1.3.2 记录网卡信息

(1) 按下快捷键 `Win + S` 打开搜索，输入"设置"。



图 1-6 找到设置

(2) 点击打开【设置】选项。



图 1-7 打开设置

(3) 点击【网络和 internet】、下滑页面点击【高级网络设置】，如下图所示；继续找到【WLAN】，可以看到相关网卡信息，记录该值，用于虚拟机配置所用。



图 1-8 记录网卡

1.3.3 防火墙配置

如果在开发板网络配置阶段，可以 ping 通虚拟机，但是无法 ping 通 PC 端，那么很可能是电脑防火墙问题，需要在电脑上勾选“允许文件和打印共享应用”通过防火墙，可以参考以下步骤操作：

(1) 打开控制面板

按下快捷键 `Win + S` 打开搜索，输入“控制面板”。



图 1-9 找到控制面板

(2) 打开"系统和安全"

如下图所示，在该界面中，找到"系统和安全"，左键点击即可。



图 1-10 打开“系统和安全”

(3) 选择"Windows Defender 防火墙"

在新出现的界面中，点击"Windows Defender 防火墙"，如下图所示。



图 1-11 打开防火墙

(4) 点击"允许应用或功能通过 Windows Defender 防火墙"

在左侧菜单栏中，找到"允许应用或功能通过 Windows Defender 防火墙"，并左键点击。



图 1-12 防火墙设置

(5) 勾选"文件和打印共享"

在新出现的界面中，向下滑动，找到"文件和打印共享"；并勾选"私有"和"公用"，使其可以通过防火墙。

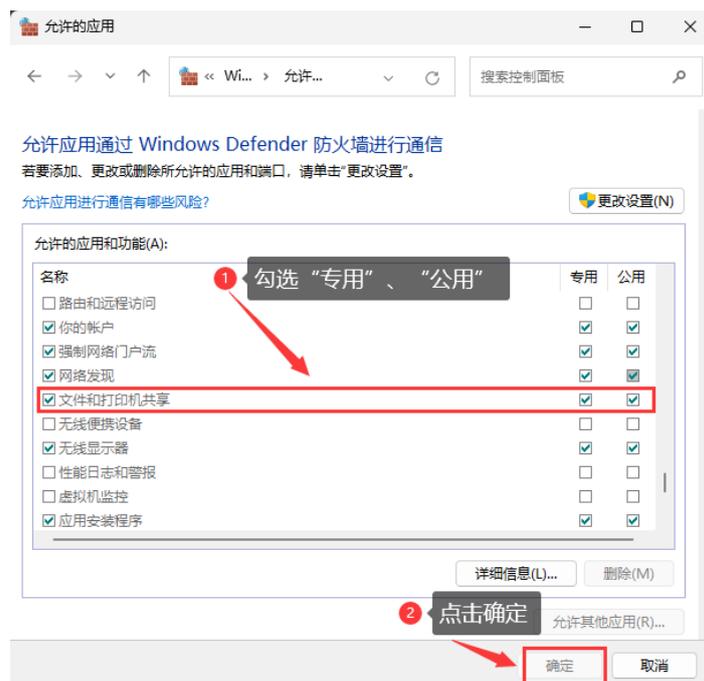


图 1-13 勾选"文件和打印共享"

1.4 虚拟机网络配置

1.4.1 VMware 网络设置

将虚拟机 Ubuntu 设置为桥接模式，便于其他电脑 ssh 访问。桥接模式，可以理解为在整个局域网中添加上一个新的电脑，整个局域网中的电脑都可以

Ping 通这台“新电脑”。

注意：如果在电脑上卸载过 VMware 软件，那么需要参考下面帖子清除，不然网络配置时可能会出现网络适配器不显示等故障。

帖子查看：[VMware 重装故障](#)

(1) 打开“虚拟网络编辑器”



图 1-14 虚拟网络编辑器

(2) 点击“更改设置”

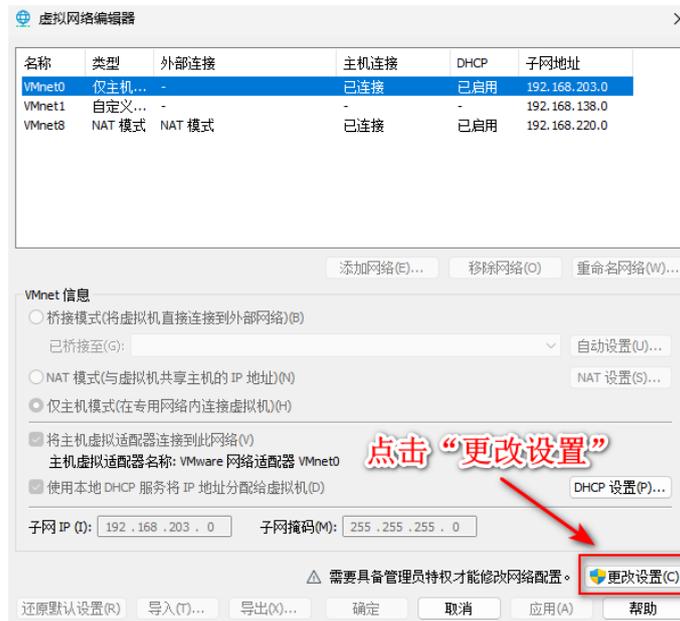


图 1-15 更改设置

(3) 添加网络



图 1-16 添加 VMnet

(4) 修改为“桥接模式”，并选择正确的网卡

注意：网卡选择要根据自己设备选择，在“Windows 系统设置”中，讲解过如何找到自己的网卡名称



图 1-17 选择正确的网卡

(5) 虚拟机设置

在 VMware 中，找到创建的虚拟机，打开其设置，如图 1-18。

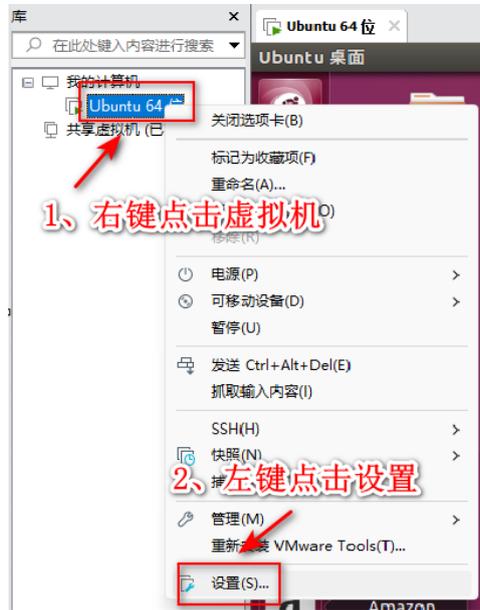


图 1-18 虚拟机设置

(6) 虚拟机适配器修改

打开网络适配器，点击“自定义:特定虚拟网络”，选择刚刚创建的 VMnet1（你的不一定是 VMnet1，可能是其他数字，请对照你前面创建的数字）

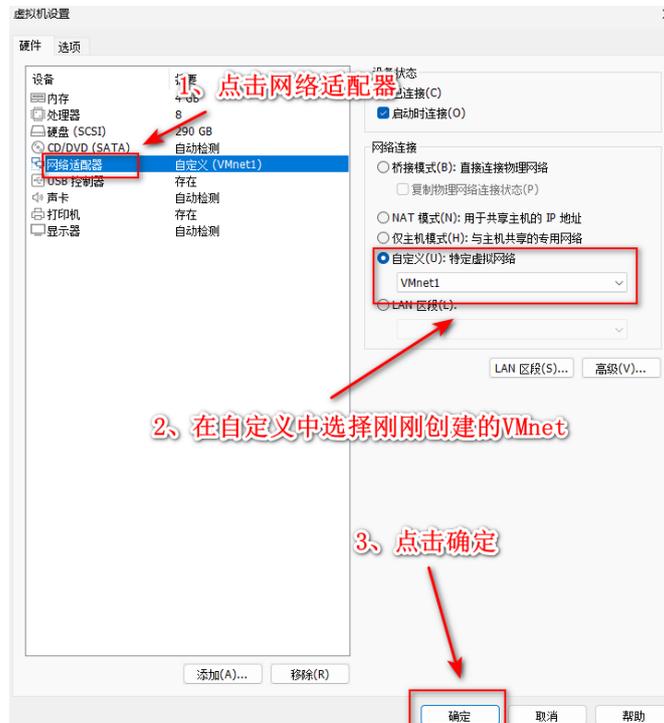


图 1-19 虚拟机适配器修改

(7) 重启网络服务

考虑大家是新手，可以直接手动重启 Ubuntu 系统；开机后桌面右键，点击“打开终端”；



图 1-20 打开终端

(8) 测试网络连接

在终端中输入命令“ping www.baidu.com”，测试网络连接；如果出现下图所示界面，表明联网成功。

```
yang@yang: ~  
yang@yang:~$ ping www.baidu.com  
PING www.a.shifen.com (153.3.238.102) 56(84) bytes of data:  
64 bytes from 153.3.238.102: icmp_seq=1 ttl=53 time=16.6 ms  
64 bytes from 153.3.238.102: icmp_seq=2 ttl=53 time=16.8 ms  
64 bytes from 153.3.238.102: icmp_seq=3 ttl=53 time=16.8 ms  
64 bytes from 153.3.238.102: icmp_seq=4 ttl=53 time=18.7 ms  
64 bytes from 153.3.238.102: icmp_seq=5 ttl=53 time=16.6 ms  
64 bytes from 153.3.238.102: icmp_seq=6 ttl=53 time=20.2 ms  
64 bytes from 153.3.238.102: icmp_seq=7 ttl=53 time=18.3 ms  
^C  
--- www.a.shifen.com ping statistics ---  
7 packets transmitted, 7 received, 0% packet loss, time 6013ms  
rtt min/avg/max/mdev = 16.629/17.764/20.258/1.288 ms  
yang@yang:~$
```

图 1-21 测试网络连接

(9) 查看 IP 地址

在 shell 中输入命令“ifconfig”，会打印出网络信息，如下图所示

```
yang@yang: ~  
yang@yang:~$ ifconfig  
ens33 Link encap:以太网 硬件地址 00:0c:29:a4:38:ae  
inet 地址:192.168.31.215 广播:192.168.31.255 掩码:255.255.255.0  
inet6 地址: fe80::3098:51e9:b5cf:a980/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 跃点数:1  
接收数据包:16438 错误:0 丢弃:0 过载:0 帧数:0  
发送数据包:11043 错误:0 丢弃:0 过载:0 载波:0 碰撞:0 发送队列长度:1000  
接收字节:1497200 (1.4 MB) 发送字节:1295850 (1.2 MB)  
  
lo Link encap:本地环回  
inet 地址:127.0.0.1 掩码:255.0.0.0  
inet6 地址: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:65536 跃点数:1  
接收数据包:254 错误:0 丢弃:0 过载:0 帧数:0  
发送数据包:254 错误:0 丢弃:0 过载:0 载波:0 碰撞:0 发送队列长度:1000  
接收字节:19120 (19.1 KB) 发送字节:19120 (19.1 KB)
```

图 1-22 查看 IP 地址

1.4.2 Ubuntu 网络设置

注意：一般是不需要自己去手动配置的，笔者多次更换过网络适配器，都不需要自己手动修改（DHCP 原因）；但是考虑到网络上有人反馈过这种问题，故添加上手动修改步骤。

(1) 打开 Ubuntu 设置

在虚拟机 Ubuntu 桌面，右侧的条形栏中找到“设置”图标，点击即可。



图 1-23 Ubuntu 设置

(2) 点击网络设置



图 1-24 点击网络设置

(3) 点击选项



图 1-25 打开网络选项

(4) 点击 IPv4 设置，在“方法”中，选择手动。

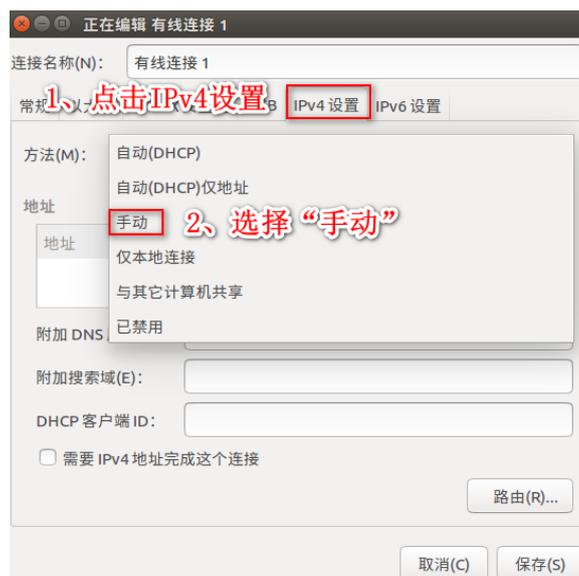


图 1-26 选择手动

(5) 添加地址

地址：192.168.31.215

子网掩码：255.255.255.0

网关：192.168.31.1

注意：网关一定要三个平台都一样，不然是不可能通信的！

图 1-27 添加地址

(6) 添加上 DNS 服务器



图 1-28 添加上 DNS 服务器

(7) 关闭网络连接，并再次打开，使前面设置的网络配置生效



图 1-29 关闭连接再打开

(8) 打开终端，输入命令“ifconfig”，判断设置是否成功；如果与前面配置相同，说明设置成功

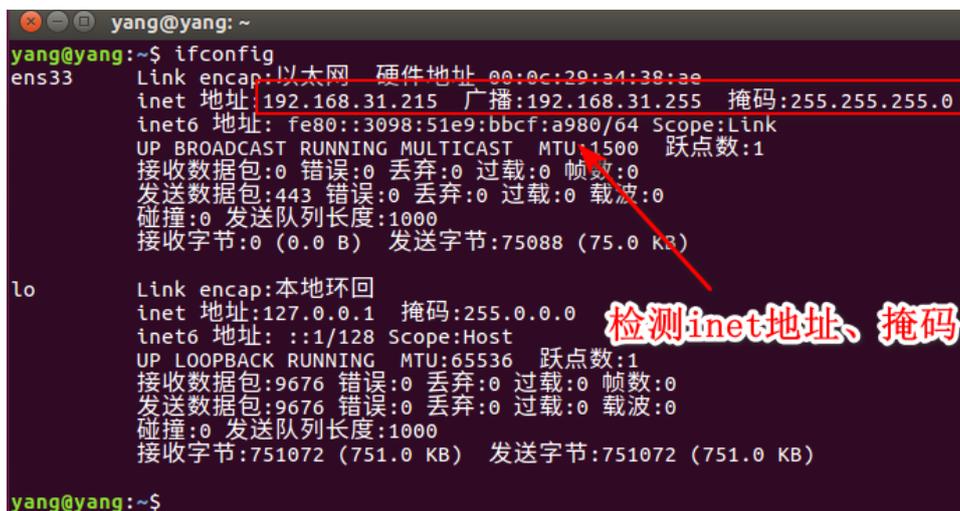


图 1-30 输入命令“ifconfig”

1.5 开发板设置

值得注意的是，开发板其实是有两个 IP，一个在 Uboot，另一个在内核；不过这里只需要设置 Uboot 阶段，当启动内核后 Uboot 也正式“消亡”了。

注意：

- 1、该步骤是修改 Uboot 环境变量，必须处于 Uboot 阶段！
- 2、网关要相同，例如” Windows 系统设置“中，查看到的 IP 地址是 192.168.31.35，那么网关就是 192.168.31.1
- 3、网络环境命令中的 IP 地址，不能已经被其他设备占有
在 U-Boot 中，在倒计时为 0 前，按下回车，开始设置环境变量。

```
U-Boot 2018.01 (Mar 06 2024 - 14:42:28 +0800) Xili
Model: ACZ702 ZYNQ Development Board
Board: Xilinx Zynq
Silicon: v3.1
I2C: ready
DRAM: ECC disabled 512 MiB
MMC: mmc@e0100000: 0 (SD)
SF: Detected w25q128fw with page size 256 Bytes, e
In: serial@e0001000
Out: serial@e0001000
Err: serial@e0001000
Net: ZYNQ GEM: e000b000, phyaddr 1, interface rg
eth0: ethernet@e000b000
Hit any key to stop autoboot: 0
Zynq>
```

图 1-31 进入 UBoot 设置

根据网关，所以我设置 Uboot 的 IP 地址为 192.168.31.31，命令如下所示：

```
setenv ipaddr 192.168.31.31
saveenv
```

将命令输入一行，回车后，再输入另一行，操作如下图所示：

```
eth0: ethernet@e000b000
Hit any key to stop autoboot: 0
Zynq> setenv ipaddr 192.168.31.31
Zynq> saveenv
Saving Environment to SPI Flash ...
SF: Detected w25q128fw with page size 256 Bytes, e
rase size 4 KiB, total 16 MiB
Erasing SPI flash ... Writing to SPI flash ... done
Zynq>
```

图 1-32 输入命令

继续输入 printenv，该命令是用于在终端展示当前的环境变量。

```
Zynq> printenv
arch=arm
baudrate=115200
bitstream_image=system.bit
bitstream_load_address=0x100000
bitstream_size=0x300000
board=zynq
board_name=zynq
```

图 1-33 输入 printenv

在打印的文字中，查找 IP 信息；如果 IP 信息与我们设置的 IP 相同，说明设置是成功的。（mobaxterm 自带搜索）

```
fileaddr=2000000
filesize=1f0
gatewayip=192.168.31.1
importbootenv=echo Importing environment from SD .
..; env import -t ${loadbootenv_addr} $filesize
initrd_high=0x20000000
ipaddr=192.168.31.31
jtagboot=echo IFIPing Linux to RAM... & tftpboot
${kernel_load_address} ${kernel_image} & tftpboot
```

配置成功

图 1-34 查找 IP 信息

1.6 验证

在设置完成后，需要验证开发板、PC 机、Ubuntu 三端是否能 ping 通。

1.6.1 开发板与 PC 机连接测试

在 uboot 中输入命令：ping 192.168.31.35；如果出现下图所示文字，说明两端是 ping 通的。（注意：不能 PC 机里面去 Ping 开发板，无法 Ping 通的！）

```
Zynq> ping 192.168.31.35
Using ethernet@e000b000 device
host 192.168.31.35 is alive
Zynq>
```

图 1-35 开发板与 PC 机测试

1.6.2 开发板与虚拟机连接测试

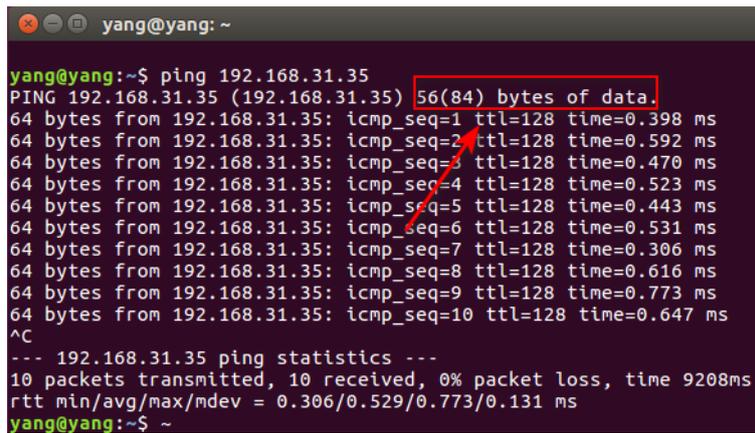
在 uboot 中输入命令：ping 192.168.31.215；如果出现下图所示文字，说明两端是 ping 通的。（注意：不能虚拟机里面去 Ping 开发板，无法 Ping 通的！）

```
Zynq> ping 192.168.31.215
Using ethernet@e000b000 device
host 192.168.31.215 is alive
Zynq>
```

图 1-36 开发板与虚拟机测试

1.6.3 虚拟机与 PC 机连接测试

在虚拟机终端中，输入命令：ping 192.168.31.35；如果出现下图所示文字，说明两端是联通的。



```
yang@yang: ~  
yang@yang:~$ ping 192.168.31.35  
PING 192.168.31.35 (192.168.31.35) 56(84) bytes of data:  
64 bytes from 192.168.31.35: icmp_seq=1 ttl=128 time=0.398 ms  
64 bytes from 192.168.31.35: icmp_seq=2 ttl=128 time=0.592 ms  
64 bytes from 192.168.31.35: icmp_seq=3 ttl=128 time=0.470 ms  
64 bytes from 192.168.31.35: icmp_seq=4 ttl=128 time=0.523 ms  
64 bytes from 192.168.31.35: icmp_seq=5 ttl=128 time=0.443 ms  
64 bytes from 192.168.31.35: icmp_seq=6 ttl=128 time=0.531 ms  
64 bytes from 192.168.31.35: icmp_seq=7 ttl=128 time=0.306 ms  
64 bytes from 192.168.31.35: icmp_seq=8 ttl=128 time=0.616 ms  
64 bytes from 192.168.31.35: icmp_seq=9 ttl=128 time=0.773 ms  
64 bytes from 192.168.31.35: icmp_seq=10 ttl=128 time=0.647 ms  
^C  
--- 192.168.31.35 ping statistics ---  
10 packets transmitted, 10 received, 0% packet loss, time 9208ms  
rtt min/avg/max/mdev = 0.306/0.529/0.773/0.131 ms  
yang@yang:~$ ~
```

图 1-37 虚拟机与 PC 机测试

到这里，三端网络设置正式结束。

1.7 总结

本章节讲解了如何配置 PC、虚拟机、开发板之间，达到互相 ping 通的目的，讲解了在嵌入式 Linux 学习与开发中，三者之间 IP 的设定和互通对于提升开发效率有着关键作用。