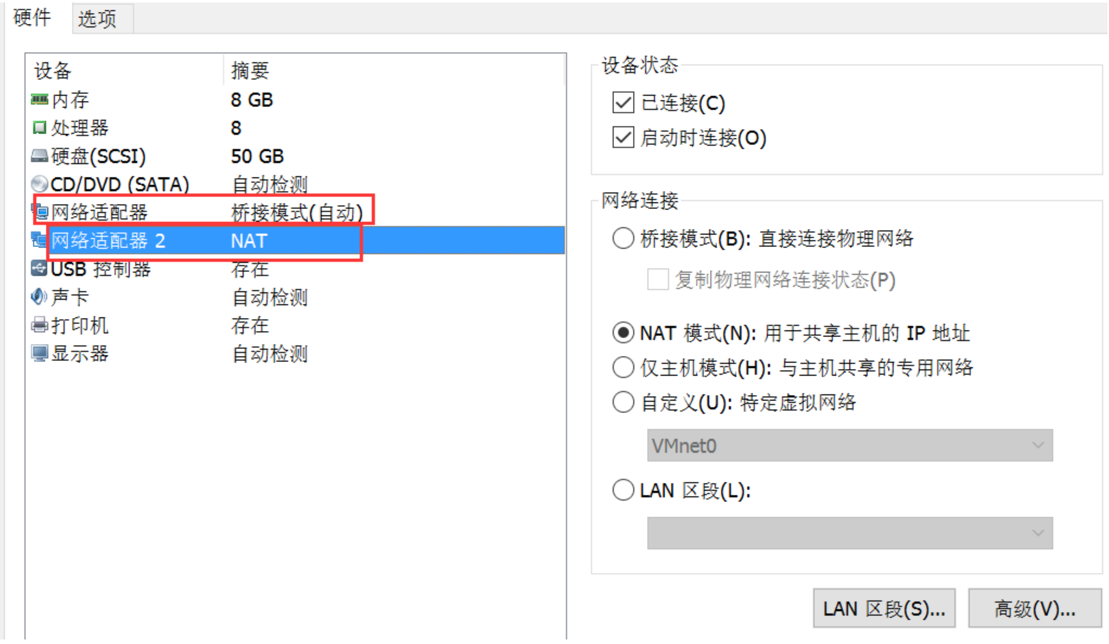
Dpdk 环境搭建

工具准备： vmware + Ubuntu 16.04 Server 版本

1.vmware 添加两块网卡



硬件 选项

设备	摘要
内存	8 GB
处理器	8
硬盘(SCSI)	50 GB
CD/DVD (SATA)	自动检测
网络适配器	桥接模式(自动)
网络适配器 2	NAT
USB 控制器	存在
声卡	自动检测
打印机	存在
显示器	自动检测

设备状态

☑ 已连接(C)

☑ 启动时连接(O)

网络连接

○ 桥接模式(B): 直接连接物理网络

□ 复制物理网络连接状态(P)

● NAT 模式(N): 用于共享主机的 IP 地址

○ 仅主机模式(H): 与主机共享的专用网络

○ 自定义(U): 特定虚拟网络

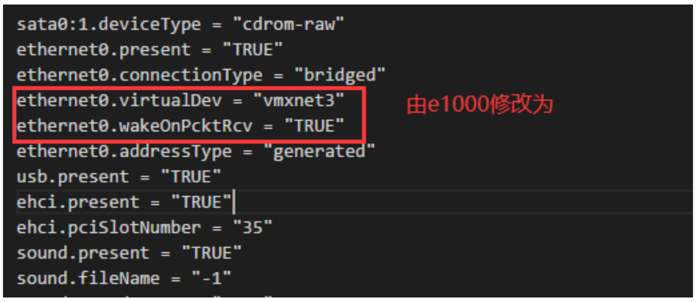
VMnet0

○ LAN 区段(L):

桥接网卡作为 DPDK 运行的网卡

NAT 网卡作为 ssh 连接的网卡

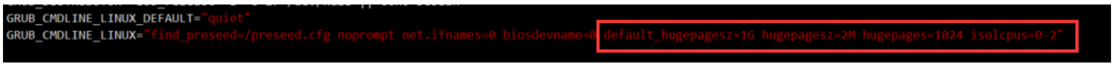
2.修改网卡配置信息



```
sata0:1.deviceType = "cdrom-raw"
ethernet0.present = "TRUE"
ethernet0.connectionType = "bridged"
ethernet0.virtualDev = "vmxnet3"
ethernet0.wakeOnPcktRcv = "TRUE"
ethernet0.addressType = "generated"
usb.present = "TRUE"
ehci.present = "TRUE"
ehci.pciSlotNumber = "35"
sound.present = "TRUE"
sound.fileName = "-1"
```

将 ethernet0.virtualDev 由 e1000 修改 vmxnet3，因为 vmware 的 vmxnet3 支持多队列网卡

3.修改 ubuntu 系统的启动参数



```
GRUB_CMDLINE_LINUX_DEFAULT="quiet"
GRUB_CMDLINE_LINUX="find_preseed=/preseed.cfg noprompt net.ifnames=0 biosdevname=0 default_hugepagesz=1G hugepagesz=2M hugepages=1024 isolcpus=0-2"
```

物理机：

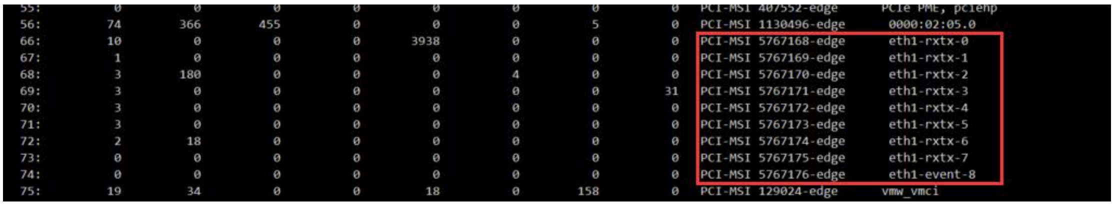
default_hugepages=1G hugepagesz=1G hugepages=20 isolcpus=0-7

虚拟机：

default_hugepages=1G hugepagesz=2M hugepages=1024 isolcpus=0-2

4.查看系统是否支持多队列网卡

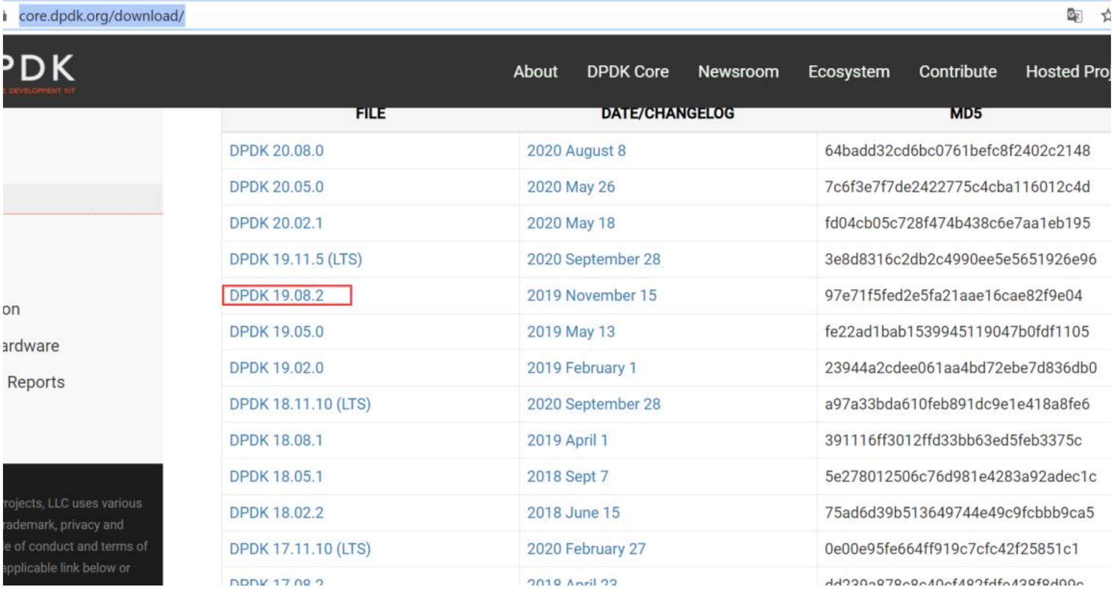
执行：cat /proc/interrupts



```
 55:      0      0      0      0      0      0      0      0   PCI-MSI 407552-edge     PCIe PME, pciehp
 56:     74    366    455      0      0      0      5      0   PCI-MSI 1130496-edge     0000:02:05.0
 66:     10      0      0      0   3938      0      0      0   PCI-MSI 5767168-edge     eth1-rxtx-0
 67:      1      0      0      0      0      0      0      0   PCI-MSI 5767169-edge     eth1-rxtx-1
 68:      3    180      0      0      0      4      0      0   PCI-MSI 5767170-edge     eth1-rxtx-2
 69:      3      0      0      0      0      0      0     31   PCI-MSI 5767171-edge     eth1-rxtx-3
 70:      3      0      0      0      0      0      0      0   PCI-MSI 5767172-edge     eth1-rxtx-4
 71:      3      0      0      0      0      0      0      0   PCI-MSI 5767173-edge     eth1-rxtx-5
 72:      2     18      0      0      0      0      0      0   PCI-MSI 5767174-edge     eth1-rxtx-6
 73:      0      0      0      0      0      0      0      0   PCI-MSI 5767175-edge     eth1-rxtx-7
 74:      0      0      0      0      0      0      0      0   PCI-MSI 5767176-edge     eth1-event-8
 75:     19     34      0      0     18      0    158      0   PCI-MSI 129024-edge     vmw_vmci
```

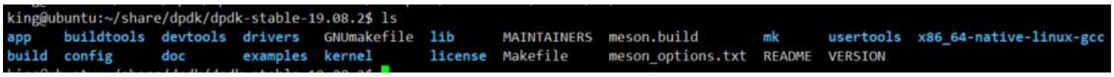
5.编译 DPDK

下载 dpdk ：https://core.dpdk.org/download/



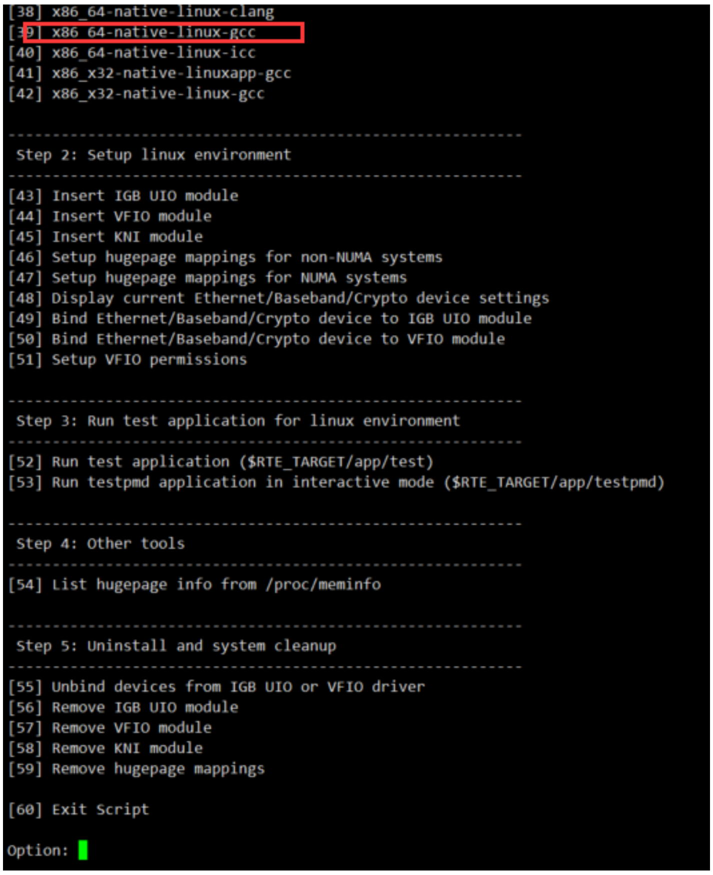
FILE	DATE/CHANGELOG	MD5
DPDK 20.08.0	2020 August 8	64badd32cd6bc0761befc8f2402c2148
DPDK 20.05.0	2020 May 26	7c6f3e7f7de2422775c4cba116012c4d
DPDK 20.02.1	2020 May 18	fd04cb05c728f474b438c6e7aa1eb195
DPDK 19.11.5 (LTS)	2020 September 28	3e8d8316c2db2c4990ee5e5651926e96
DPDK 19.08.2	2019 November 15	97e71f5fed2e5fa21aae16cae82f9e04
DPDK 19.05.0	2019 May 13	fe22ad1bab1539945119047b0fdf1105
DPDK 19.02.0	2019 February 1	23944a2cdee061aa4bd72ebe7d836db0
DPDK 18.11.10 (LTS)	2020 September 28	a97a33bda610feb891dc9e1e418a8fe6
DPDK 18.08.1	2019 April 1	391116ff3012ffd33bb63ed5feb3375c
DPDK 18.05.1	2018 Sept 7	5e278012506c76d981e4283a92adec1c
DPDK 18.02.2	2018 June 15	75ad6d39b513649744e49c9fcbbb9ca5
DPDK 17.11.10 (LTS)	2020 February 27	0e00e95fe664ff919c7cfc42f25851c1
DPDK 17.08.2	2018 April 23	dd230a878c8c40cf482fdfe438f8d90c

随便挑选版本，不同版本直接子系统接口会有差异，建议选择 dpdk 19.08.2



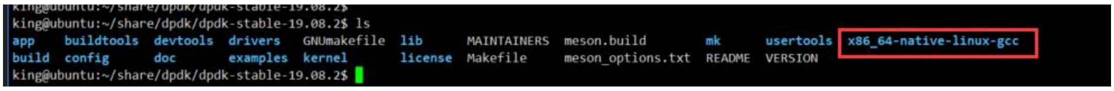
```
king@ubuntu:~/share/dpdk/dpdk-stable-19.08.2$ ls
app    buildtools  devtools  drivers   GNUmakefile  lib      MAINTAINERS  meson.build        mk      usertools  x86_64-native-linux-gcc
build  config      doc       examples  kernel       license  Makefile     meson_options.txt  README  VERSION
```

可以通过 usertools/dpdk-setup.sh



```
[38] x86_64-native-linux-clang
[39] x86_64-native-linux-gcc
[40] x86_64-native-linux-icc
[41] x86_x32-native-linuxapp-gcc
[42] x86_x32-native-linux-gcc

----------------------------------------------------------
 Step 2: Setup linux environment
----------------------------------------------------------
[43] Insert IGB UIO module
[44] Insert VFIO module
[45] Insert KNI module
[46] Setup hugepage mappings for non-NUMA systems
[47] Setup hugepage mappings for NUMA systems
[48] Display current Ethernet/Baseband/Crypto device settings
[49] Bind Ethernet/Baseband/Crypto device to IGB UIO module
[50] Bind Ethernet/Baseband/Crypto device to VFIO module
[51] Setup VFIO permissions

----------------------------------------------------------
 Step 3: Run test application for linux environment
----------------------------------------------------------
[52] Run test application ($RTE_TARGET/app/test)
[53] Run testpmd application in interactive mode ($RTE_TARGET/app/testpmd)

----------------------------------------------------------
 Step 4: Other tools
----------------------------------------------------------
[54] List hugepage info from /proc/meminfo

----------------------------------------------------------
 Step 5: Uninstall and system cleanup
----------------------------------------------------------
[55] Unbind devices from IGB UIO or VFIO driver
[56] Remove IGB UIO module
[57] Remove VFIO module
[58] Remove KNI module
[59] Remove hugepage mappings

[60] Exit Script

Option:
```

64 位系统选择 39.



```
king@ubuntu:~/share/dpdk/dpdk-stable-19.08.2$
king@ubuntu:~/share/dpdk/dpdk-stable-19.08.2$ ls
app    buildtools  devtools  drivers   GNUmakefile  lib      MAINTAINERS  meson.build        mk      usertools  x86_64-native-linux-gcc
build  config      doc       examples  kernel       license  Makefile     meson_options.txt  README  VERSION
king@ubuntu:~/share/dpdk/dpdk-stable-19.08.2$
```

编译完会多出 x86_64-native-linux-gcc 的文件夹

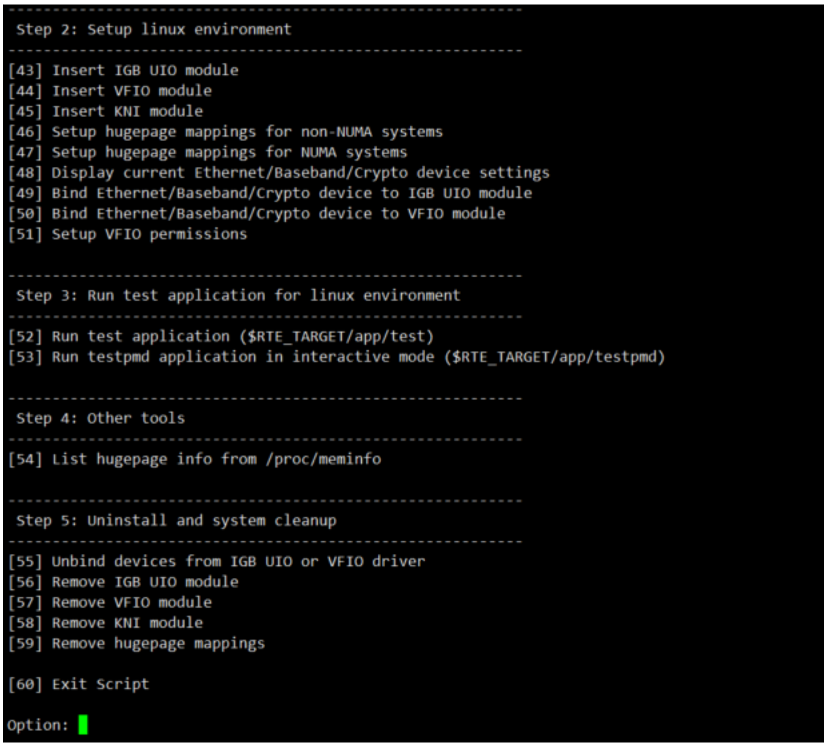
6.设置 DPDK 的环境变量

```
# export RTE_SDK=/home/dpdk
# export RTE_TARGET=x86_64-native-linux-gcc
```

7.执行 testpmd

执行

```
# /usertools/dpdk-setup.sh
```



```
----------------------------------------------------------
 Step 2: Setup linux environment
----------------------------------------------------------
[43] Insert IGB UIO module
[44] Insert VFIO module
[45] Insert KNI module
[46] Setup hugepage mappings for non-NUMA systems
[47] Setup hugepage mappings for NUMA systems
[48] Display current Ethernet/Baseband/Crypto device settings
[49] Bind Ethernet/Baseband/Crypto device to IGB UIO module
[50] Bind Ethernet/Baseband/Crypto device to VFIO module
[51] Setup VFIO permissions

----------------------------------------------------------
 Step 3: Run test application for linux environment
----------------------------------------------------------
[52] Run test application ($RTE_TARGET/app/test)
[53] Run testpmd application in interactive mode ($RTE_TARGET/app/testpmd)

----------------------------------------------------------
 Step 4: Other tools
----------------------------------------------------------
[54] List hugepage info from /proc/meminfo

----------------------------------------------------------
 Step 5: Uninstall and system cleanup
----------------------------------------------------------
[55] Unbind devices from IGB UIO or VFIO driver
[56] Remove IGB UIO module
[57] Remove VFIO module
[58] Remove KNI module
[59] Remove hugepage mappings

[60] Exit Script

Option:
```

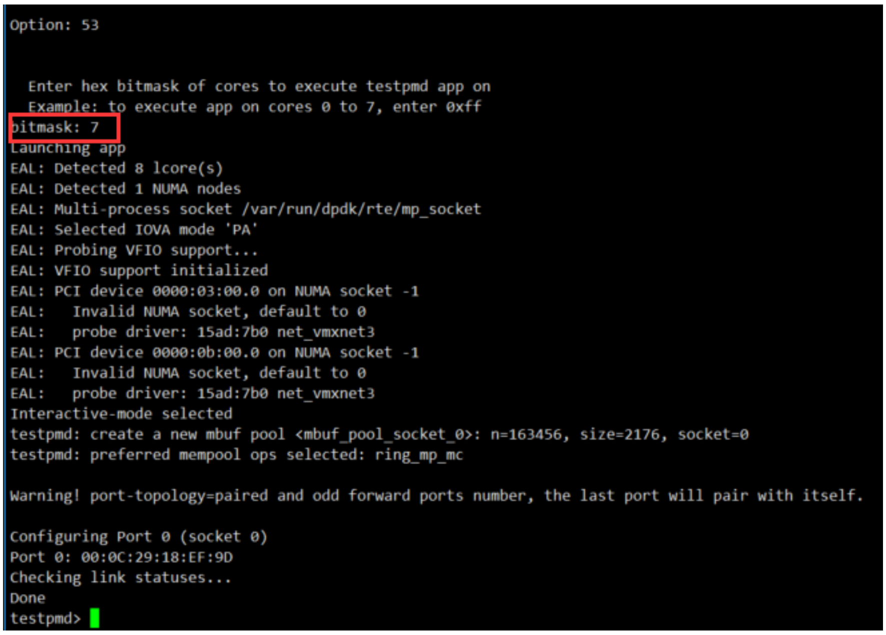
选择 43 插入 IGB_UIO 模块， 选择网卡为 vmxnet3 会加载此模块

选择 44 插入 VFIO 模块，选择网卡为 e1000 会加载此模块

选择 49 绑定 igb_uio 模块， 也可以退出，通过命令来执行。

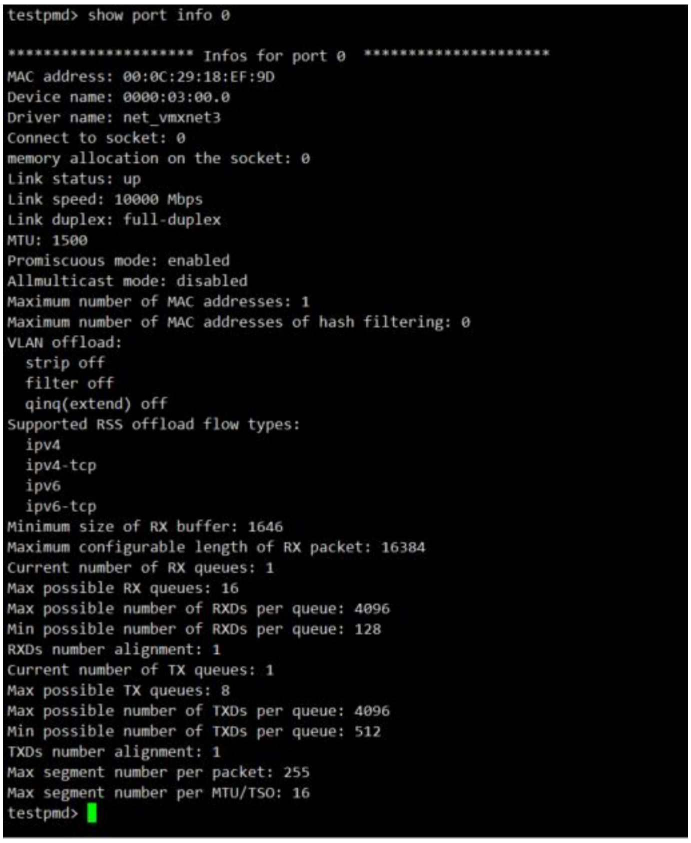
```
# ifconfig eth0 down
# /usertools/dpdk-devbind.py --bind=igb_uio eth0
```

选择 53 运行 testpmd



```
Option: 53


  Enter hex bitmask of cores to execute testpmd app on
  Example: to execute app on cores 0 to 7, enter 0xff
bitmask: 7
Launching app
EAL: Detected 8 lcore(s)
EAL: Detected 1 NUMA nodes
EAL: Multi-process socket /var/run/dpdk/rte/mp_socket
EAL: Selected IOVA mode 'PA'
EAL: Probing VFIO support...
EAL: VFIO support initialized
EAL: PCI device 0000:03:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
EAL: PCI device 0000:0b:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
Interactive-mode selected
testpmd: create a new mbuf pool <mbuf_pool_socket_0>: n=163456, size=2176, socket=0
testpmd: preferred mempool ops selected: ring_mp_mc

Warning! port-topology=paired and odd forward ports number, the last port will pair with itself.

Configuring Port 0 (socket 0)
Port 0: 00:0C:29:18:EF:9D
Checking link statuses...
Done
testpmd>
```

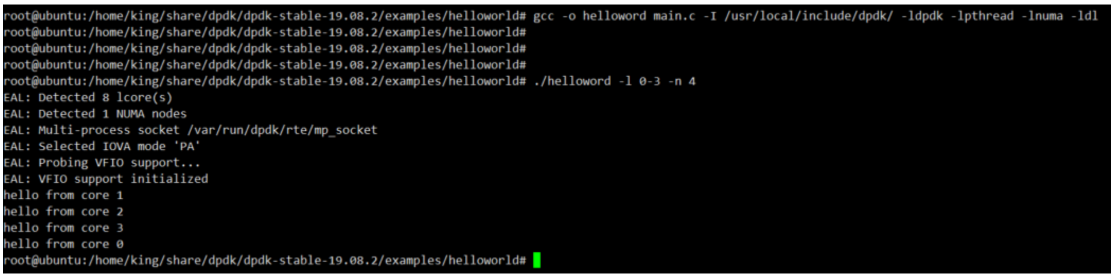
```
> show port info 0
```



```
testpmd> show port info 0

********************* Infos for port 0  *********************
MAC address: 00:0C:29:18:EF:9D
Device name: 0000:03:00.0
Driver name: net_vmxnet3
Connect to socket: 0
memory allocation on the socket: 0
Link status: up
Link speed: 10000 Mbps
Link duplex: full-duplex
MTU: 1500
Promiscuous mode: enabled
Allmulticast mode: disabled
Maximum number of MAC addresses: 1
Maximum number of MAC addresses of hash filtering: 0
VLAN offload:
  strip off
  filter off
  qinq(extend) off
Supported RSS offload flow types:
  ipv4
  ipv4-tcp
  ipv6
  ipv6-tcp
Minimum size of RX buffer: 1646
Maximum configurable length of RX packet: 16384
Current number of RX queues: 1
Max possible RX queues: 16
Max possible number of RXDs per queue: 4096
Min possible number of RXDs per queue: 128
RXDs number alignment: 1
Current number of TX queues: 1
Max possible TX queues: 8
Max possible number of TXDs per queue: 4096
Min possible number of TXDs per queue: 512
TXDs number alignment: 1
Max segment number per packet: 255
Max segment number per MTU/TSO: 16
testpmd> 
```

8.编译 DPDK 程序

进入 example/helloworld ,
可以直接 make,
也可以通过 gcc 命令编译
# gcc -o helloword main.c -I /usr/local/include/dpdk/ -ldpdk -lpthread -lnuma -ldl



```
root@ubuntu:/home/king/share/dpdk/dpdk-stable-19.08.2/examples/helloworld# gcc -o helloword main.c -I /usr/local/include/dpdk/ -ldpdk -lpthread -lnuma -ldl
root@ubuntu:/home/king/share/dpdk/dpdk-stable-19.08.2/examples/helloworld#
root@ubuntu:/home/king/share/dpdk/dpdk-stable-19.08.2/examples/helloworld#
root@ubuntu:/home/king/share/dpdk/dpdk-stable-19.08.2/examples/helloworld#
root@ubuntu:/home/king/share/dpdk/dpdk-stable-19.08.2/examples/helloworld# ./helloword -l 0-3 -n 4
EAL: Detected 8 lcore(s)
EAL: Detected 1 NUMA nodes
EAL: Multi-process socket /var/run/dpdk/rte/mp_socket
EAL: Selected IOVA mode 'PA'
EAL: Probing VFIO support...
EAL: VFIO support initialized
hello from core 1
hello from core 2
hello from core 3
hello from core 0
root@ubuntu:/home/king/share/dpdk/dpdk-stable-19.08.2/examples/helloworld# 
```

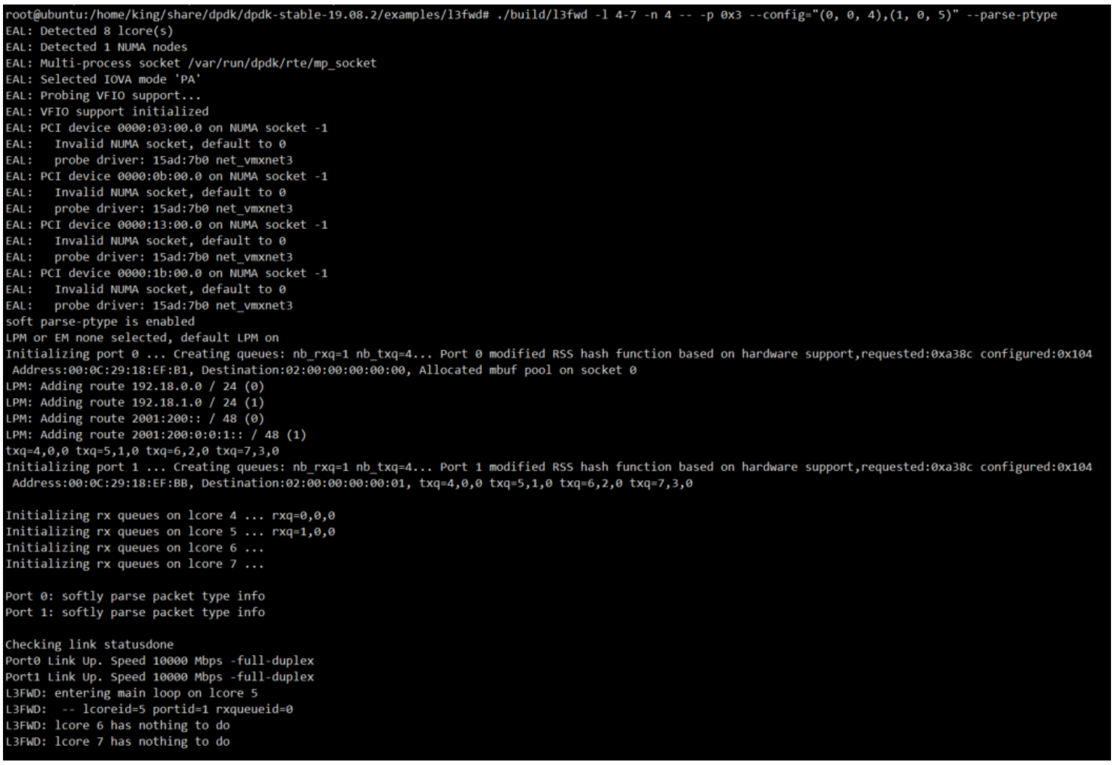
9. 运行 DPDK 案例

Kni 运行：

```
./build/kni -l 4-7 -n 4 -- -P -p 0x3 -m --config="(0, 4, 6),(1, 5, 7)"
```

L3fwd 运行

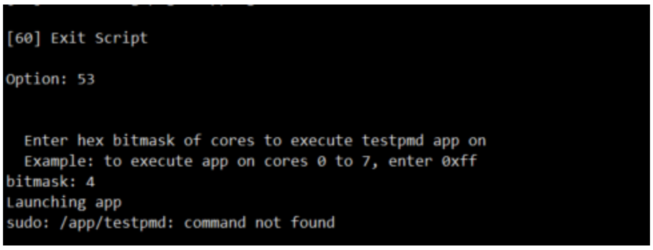
```
# ./build/l3fwd -l 4-7 -n 4 -- -p 0x3 --config="(0, 0, 4),(1, 0, 5)" --parse-ptype
```



```
root@ubuntu:/home/king/share/dpdk/dpdk-stable-19.08.2/examples/l3fwd# ./build/l3fwd -l 4-7 -n 4 -- -p 0x3 --config="(0, 0, 4),(1, 0, 5)" --parse-ptype
EAL: Detected 8 lcore(s)
EAL: Detected 1 NUMA nodes
EAL: Multi-process socket /var/run/dpdk/rte/mp_socket
EAL: Selected IOVA mode 'PA'
EAL: Probing VFIO support...
EAL: VFIO support initialized
EAL: PCI device 0000:03:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
EAL: PCI device 0000:0b:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
EAL: PCI device 0000:13:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
EAL: PCI device 0000:1b:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
soft parse-ptype is enabled
LPM or EM none selected, default LPM on
Initializing port 0 ... Creating queues: nb_rxq=1 nb_txq=4... Port 0 modified RSS hash function based on hardware support,requested:0xa38c configured:0x104
 Address:00:0C:29:18:EF:B1, Destination:02:00:00:00:00:00, Allocated mbuf pool on socket 0
LPM: Adding route 192.18.0.0 / 24 (0)
LPM: Adding route 192.18.1.0 / 24 (1)
LPM: Adding route 2001:200:: / 48 (0)
LPM: Adding route 2001:200:0:0:1:: / 48 (1)
txq=4,0,0 txq=5,1,0 txq=6,2,0 txq=7,3,0
Initializing port 1 ... Creating queues: nb_rxq=1 nb_txq=4... Port 1 modified RSS hash function based on hardware support,requested:0xa38c configured:0x104
 Address:00:0C:29:18:EF:BB, Destination:02:00:00:00:00:01, txq=4,0,0 txq=5,1,0 txq=6,2,0 txq=7,3,0

Initializing rx queues on lcore 4 ... rxq=0,0,0
Initializing rx queues on lcore 5 ... rxq=1,0,0
Initializing rx queues on lcore 6 ...
Initializing rx queues on lcore 7 ...

Port 0: softly parse packet type info
Port 1: softly parse packet type info

Checking link statusdone
Port0 Link Up. Speed 10000 Mbps -full-duplex
Port1 Link Up. Speed 10000 Mbps -full-duplex
L3FWD: entering main loop on lcore 5
L3FWD:  -- lcoreid=5 portid=1 rxqueueid=0
L3FWD: lcore 6 has nothing to do
L3FWD: lcore 7 has nothing to do
```

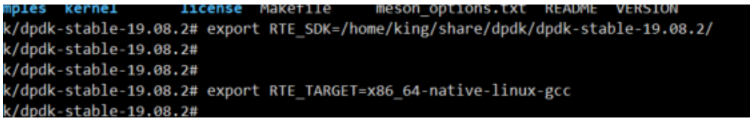
常见错误

错误 1：

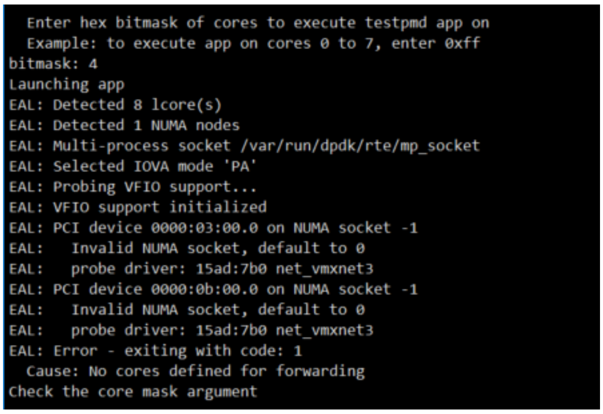


```
[60] Exit Script

Option: 53


  Enter hex bitmask of cores to execute testpmd app on
  Example: to execute app on cores 0 to 7, enter 0xff
bitmask: 4
Launching app
sudo: /app/testpmd: command not found
```

没有设置环境变量，请查看本篇《设置 dpdk 的环境变量》



```
mples  kernel       license  Makefile     meson_options.txt  README  VERSION
k/dpdk-stable-19.08.2# export RTE_SDK=/home/king/share/dpdk/dpdk-stable-19.08.2/
k/dpdk-stable-19.08.2#
k/dpdk-stable-19.08.2#
k/dpdk-stable-19.08.2# export RTE_TARGET=x86_64-native-linux-gcc
k/dpdk-stable-19.08.2#
```



```
  Enter hex bitmask of cores to execute testpmd app on
  Example: to execute app on cores 0 to 7, enter 0xff
bitmask: 4
Launching app
EAL: Detected 8 lcore(s)
EAL: Detected 1 NUMA nodes
EAL: Multi-process socket /var/run/dpdk/rte/mp_socket
EAL: Selected IOVA mode 'PA'
EAL: Probing VFIO support...
EAL: VFIO support initialized
EAL: PCI device 0000:03:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
EAL: PCI device 0000:0b:00.0 on NUMA socket -1
EAL:   Invalid NUMA socket, default to 0
EAL:   probe driver: 15ad:7b0 net_vmxnet3
EAL: Error - exiting with code: 1
  Cause: No cores defined for forwarding
Check the core mask argument
```

bitmask 请选择 7, bit:111