

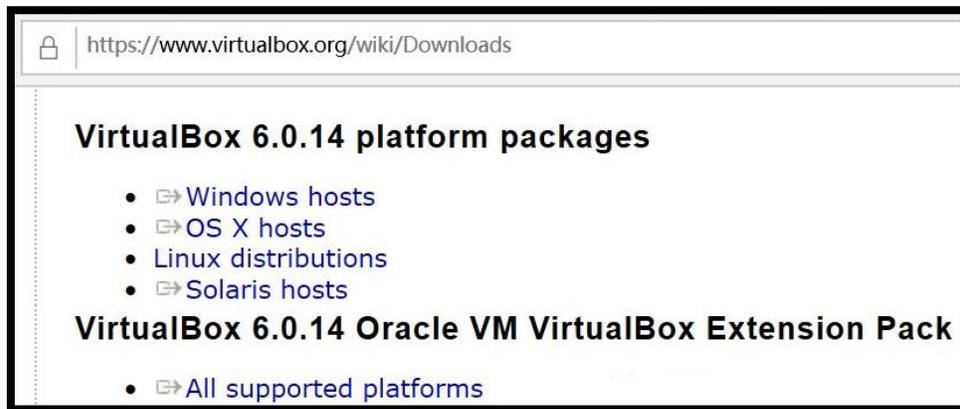
## 如何在 Virtual Box 上运行 VxWorks 7

本文介绍如何在 Virtual Box 上部署运行 [VxWorks 7](#)

### VirtualBox

官网下载 VirtualBox 和 Extension Pack

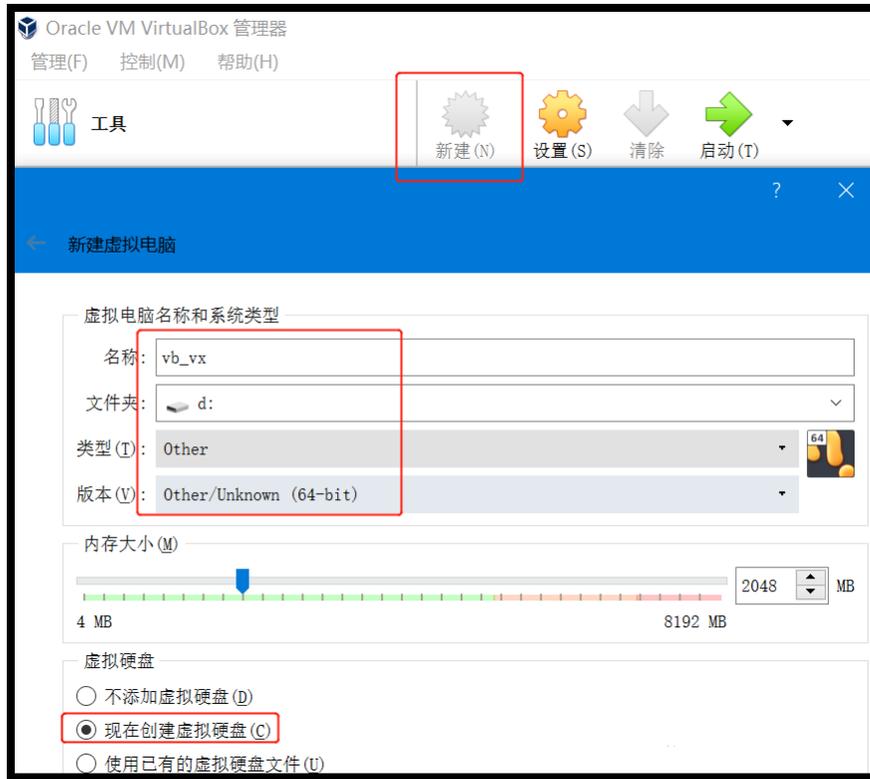
<https://www.virtualbox.org/wiki/Download>



使用默认选项安装 VirtualBox，然后在全局设定里添加扩展



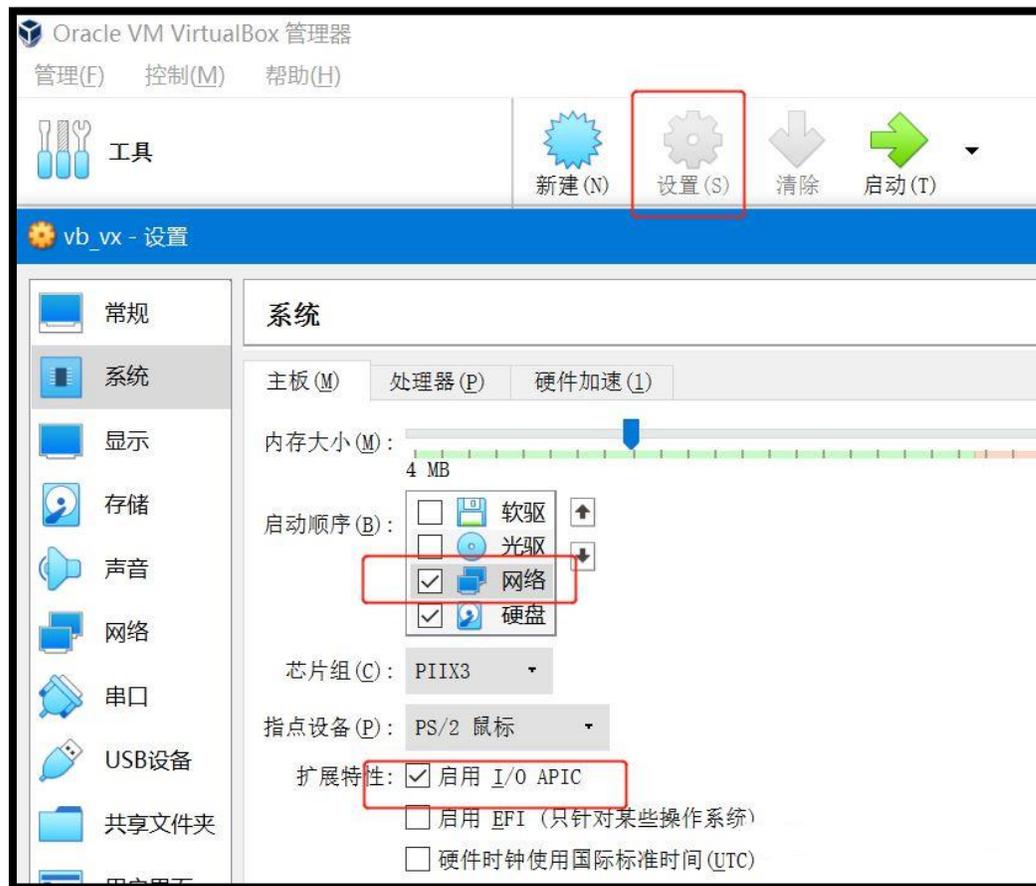
新建一个 64 位虚拟机



### 创建 HDD 硬盘



使用网络引导，启用 IO APIC



选择 Intel 网卡，使用 Host-Only 方式

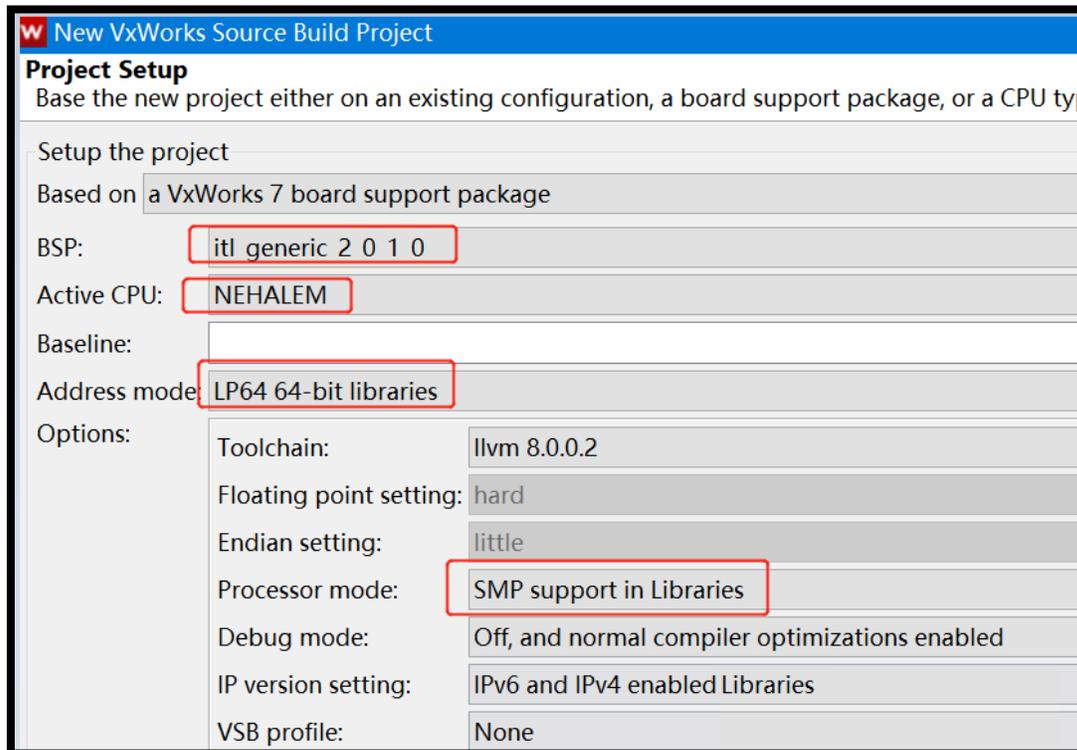


查看 Host-Only 网络地址，VB 用的网段是 192.168.56.x



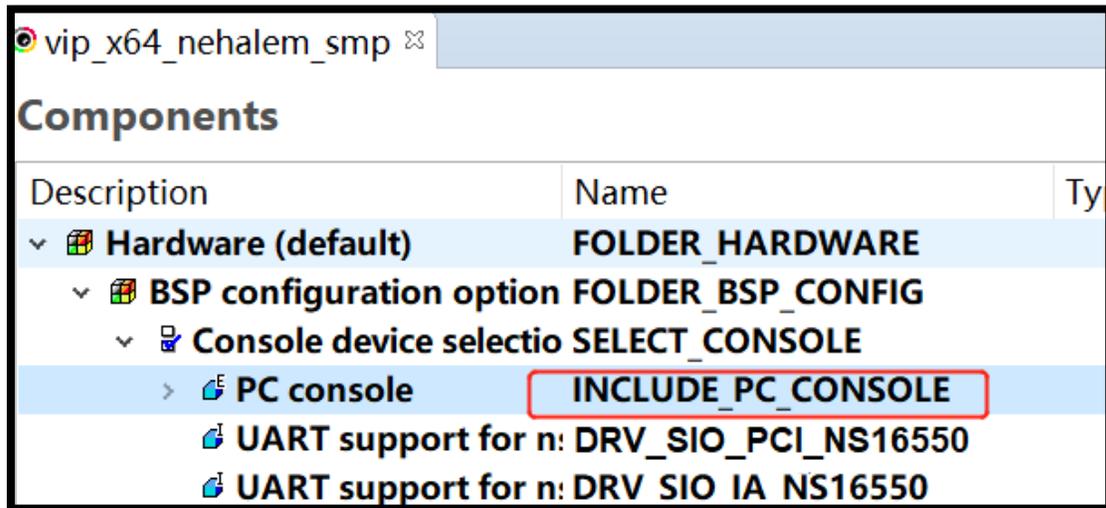
Image

在 WorkBench 里创建 VSB，使用 Intel 的 64 位 NEHALEM 处理器，SMP 或 UP 应该都可以

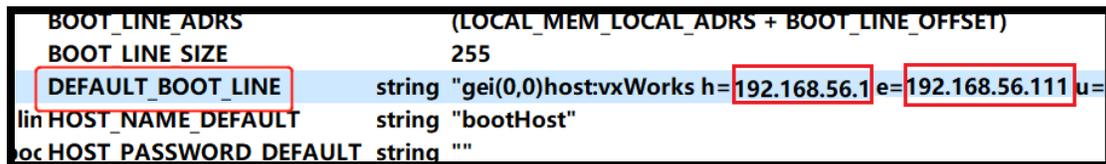


编译后，基于此 VSB，创建 VIP 添加想要的组件，

例如 INCLUDE\_PC\_CONSOLE



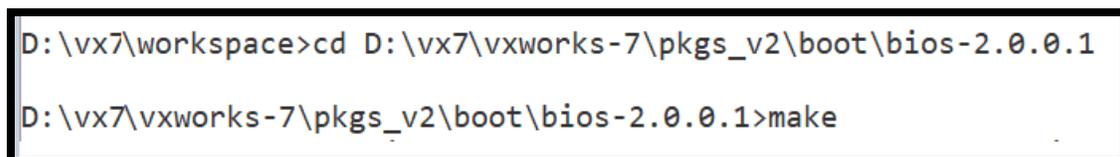
在 DEFAULT\_BOOT\_LINE 里设置 IP 地址



编译这个 VIP，将得到的 VxWorks 镜像更名为 bootapp.sys

bootloader

打开 WorkBench4 的 Terminal 窗口，在目录 vxworks-7\pkgs\_v2\boot\bios-a.b.c.d 里执行 make，得到 vxStage1Boot.bin

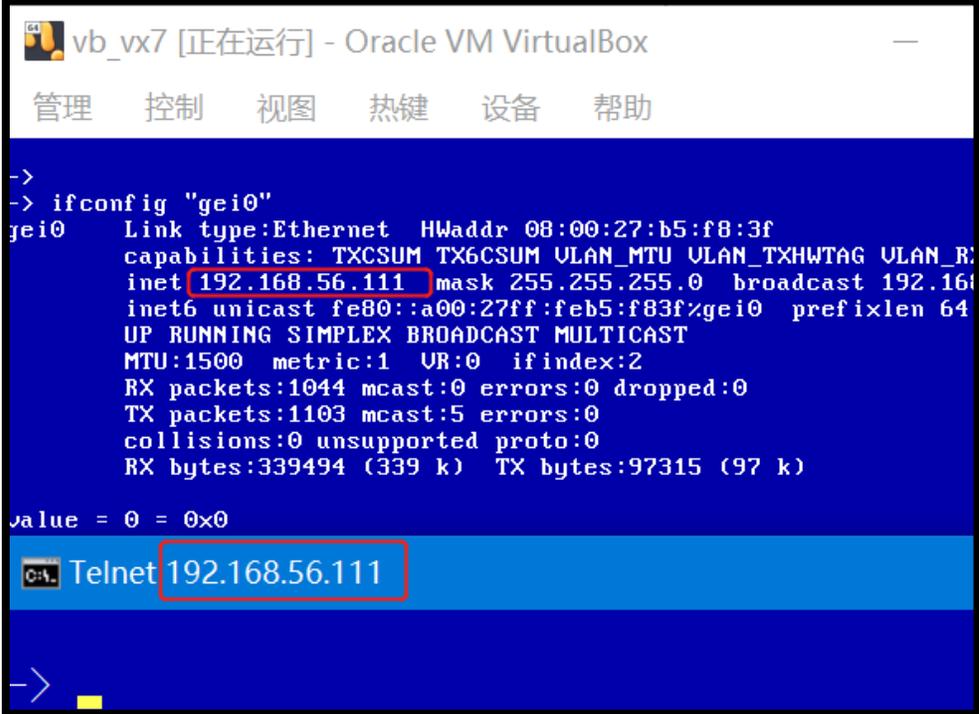


Tftp

把上文的 bootapp.sys、vxStage1Boot.bin 与 Tftp 工具放到同一目录，例如 d: 启动这个完整版的 Tftp 工具，如图设置 DHCP



网络正常



```
vb_vx7 [正在运行] - Oracle VM VirtualBox
管理 控制 视图 热键 设备 帮助

->
-> ifconfig "gei0"
gei0  Link type:Ethernet  HWaddr 08:00:27:b5:f8:3f
capabilities: TXCSUM TX6CSUM VLAN_MTU VLAN_TXHWTAG VLAN_R
inet 192.168.56.111 mask 255.255.255.0 broadcast 192.16
inet6 unicast fe80::a00:27ff:feb5:f83f%gei0 prefixlen 64
UP RUNNING SIMPLEX BROADCAST MULTICAST
MTU:1500 metric:1 UR:0 ifindex:2
RX packets:1044 mcast:0 errors:0 dropped:0
TX packets:1103 mcast:5 errors:0
collisions:0 unsupported proto:0
RX bytes:339494 (339 k) TX bytes:97315 (97 k)

value = 0 = 0x0
Telnet 192.168.56.111

->
```

硬盘正常

```
-> dosfsDiskFormat "/ata0"
Formatting /ata0 for DOSFS
value = 0 = 0x0
-> ls "/ata0"
value = 0 = 0x0
```

DKM 正常

```
dkm.c
1 /* includes */
2
3 #include "vxWorks.h"
4
5
6 void start(void) {
7     printf("hello from %s\n", __FILE__);
8 }
9
```

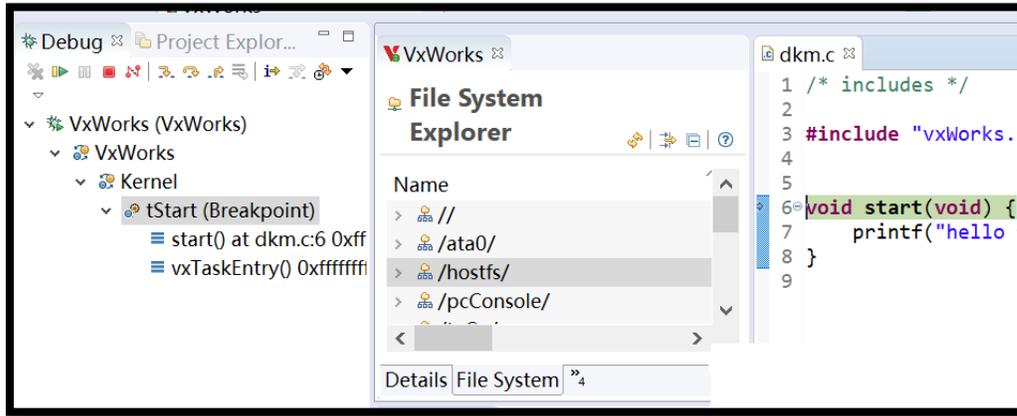
```
Telnet 192.168.56.111
->
-> ld<dkm_x64.out
value = -140737484258512 = 0xffff8000003e
-> start
hello from D:/vx7/workspace/dkm_x64/dkm.c
value = 48 = 0x30 = '0'
->
```

RTP 正常

```
rtp.c
1 /* includes */
2
3 #include <stdio.h>
4
5 int main (
6     int argc, /* number of arguments */
7     char * argv[] /* array of arguments */
8 ) {
9     printf("hello from %s\n", __FILE__);
10    return 0;
11 }
12
```

```
Telnet 192.168.56.111
-> rtpSp "rtp_x64.vxe"
value = -140737483970544 = 0xffff80000042e810
-> hello from D:/vx7/workspace/rtp_x64/rtp.c
->
```

Debugger 正常



PCI 设备列表

可以看到 VB 用的显卡也是自己定义的，Vx7 还没有支持它

```

-> vxbPciCtrlShow
Pci controller (PNP0A03:0) devId=0xffff8000000cb920
value = 52 = 0x34 = '4'
-> vxbPciDeviceShow 0xffff8000000cb920
Scanning functions of each PCI device on bus 0
bus    device    function  vendorID  deviceID  class/rev
0      0          0         0x8086    0x1237    0x06000002
0      1          0         0x8086    0x7000    0x06010000
0      1          1         0x8086    0x7111    0x01018a01
0      2          0         0x80ee    0xbeef    0x03000000
0      3          0         0x8086    0x100e    0x02000002
0      4          0         0x80ee    0xcafe    0x08800000
0      5          0         0x8086    0x2415    0x04010001
0      6          0         0x106b    0x003f    0x0c031000
0      7          0         0x8086    0x7113    0x06800008
0      11         0         0x8086    0x265c    0x0e032000
value = 0 = 0x0

```

Annotations on the right side of the table:

- HOST/PCI桥 (points to 0x1237)
- PCI/ISA桥 (points to 0x7000)
- IDE硬盘 (points to 0x7111)
- 显卡 (points to 0x100e)
- 网卡 (points to 0x7113)
- VB Guest设备 (points to 0x003f)
- 声卡 (points to 0x03000000)
- USB OHCI (points to 0x04010001)
- 电源管理 (points to 0x06800008)
- USB EHCI (points to 0x0e032000)