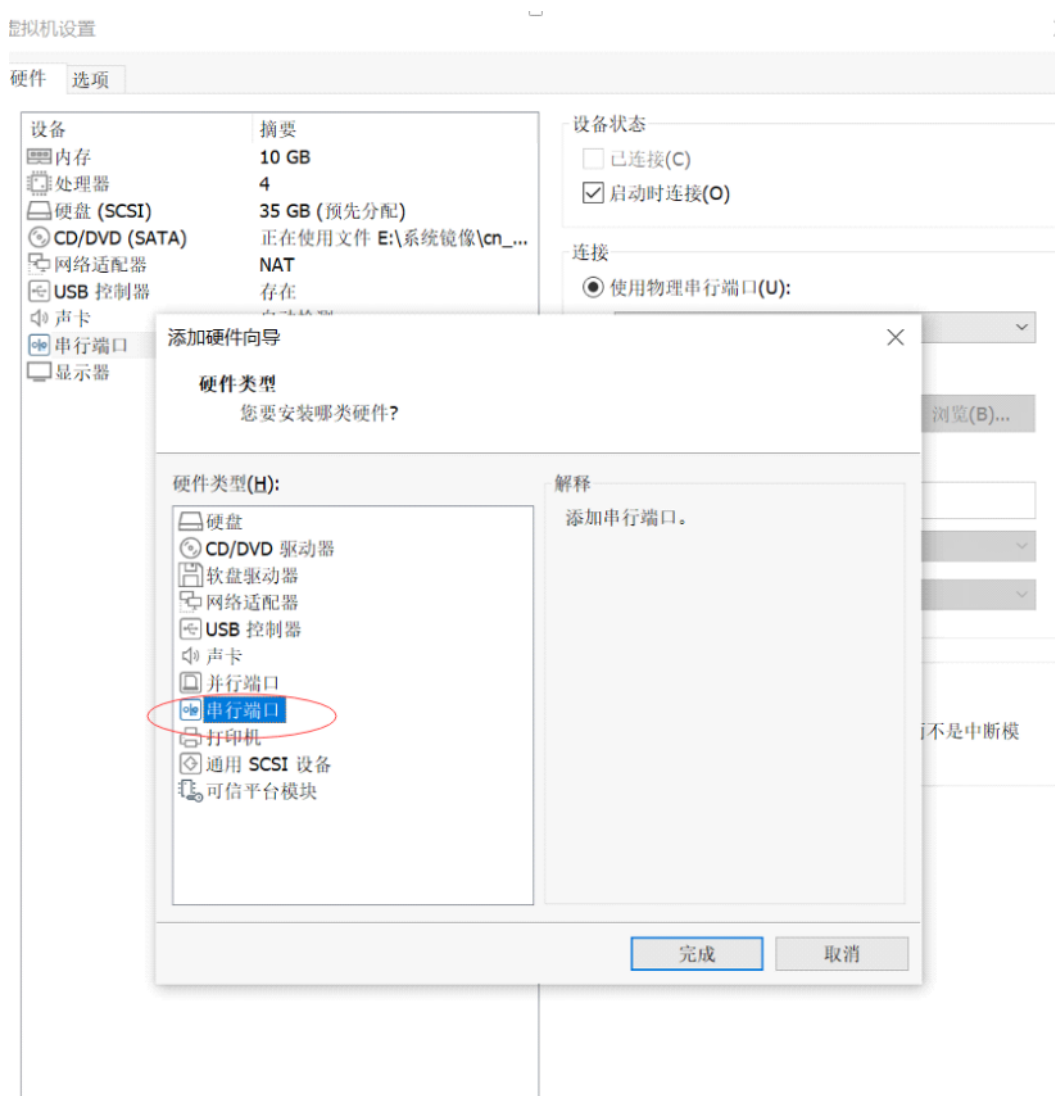


环境: VS2019+win7虚拟机

1.虚拟机设置

1.1 添加串行端口

虚拟机设置中，先移除打印机，然后添加**串口**。(打印机可能会占用COM1端口，所以先移除，当然使用其他端口也可以，但相关设置都要更改为相应的端口)



设置如下

管道地址: [\\.\pipe\com_1](#)

设备状态

☐ 已连接(C)
☒ 启动时连接(Q)

连接

☐ 使用物理串行端口(U):

自动检测

☐ 使用输出文件(S):

浏览(B)...

☒ 使用命名的管道(N):

\\.\pipe\com_1

该端是服务器。

另一端是应用程序。

I/O 模式








☒ 轮询时主动放弃 CPU(Y)
 允许客户机操作系统在轮询模式下 (而不是中断模式)使用此串行端口。

1.2. 设置测试模式

在本地WDK的安装目录下找到 **Remote** 文件夹，复制相应的 **WDK Test Target Setup.msi** 文件到虚拟机中打开。

« Program Files (x86) » Windows Kits » 10 » Remote » x86

搜索"x86"

名称	修改日期	类型	大小
 9bcb3fab78e80d68be28892ea7ad46c3.cab	2021/10/11 17:39	360压缩 CAB 文件	
 5509e4710313421be8d5e7cfbfde4d30.cab	2021/10/11 17:38	360压缩 CAB 文件	
 b1a8ecf890b4ffddd05e8c9422f55db7.cab	2021/10/11 17:39	360压缩 CAB 文件	
 ec9d39539c27e8cf5ad39bffce00c34e.cab	2021/10/11 17:40	360压缩 CAB 文件	
 WDK Test Target Setup x86-x86_en-us.msi	2021/10/11 17:29	Windows Installer ...	
 Windows SDK DirectX x86 Remote-x86_en-us.msi	2021/10/11 17:40	Windows Installer ...	
 Windows SDK for Windows Store Apps DirectX x86 Remote-x86_en-us.msi	2021/10/11 17:40	Windows Installer ...	

在虚拟机管理员权限运行cmd，输入命令bcdedit /copy {current} /d DebugEntry

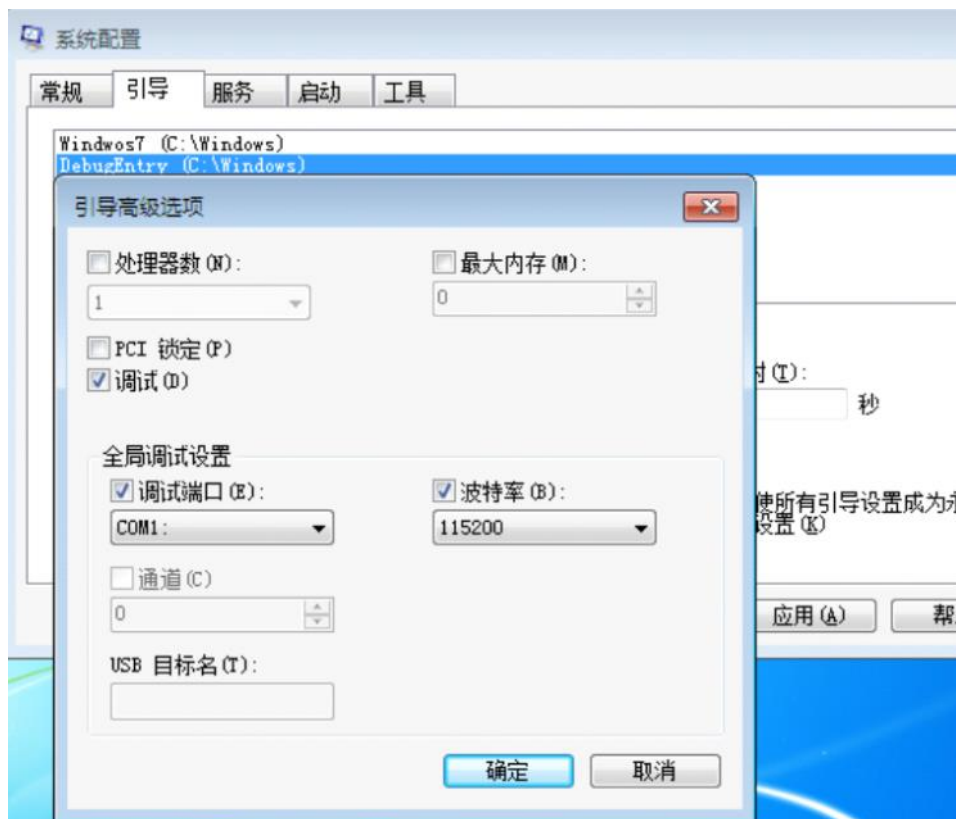
```

管理员: C:\Windows\System32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Windows\system32>bcdedit /copy {current} /d DebugEntry
已将该项成功复制到 {1b59be87-1a27-11ec-928d-eea614122b2e}。

C:\Windows\system32>
  
```

之后打开msconfig，选择引导选项卡，找到我们刚刚创建的DebugEntry并点击高级选项，进行如下配置：



重启，选择 DebugEntry 启动

管理员权限运行cmd，输入 **bcdedit /set testsigning on**和 **bcdedit /Debug on**，开启测试模式。

输入 **bcdedit /dbgsettings** 查看设置情况如下。

```
管理员: C:\Windows\System32\cmd.exe

C:\Windows\system32>bcdedit /set testsigning on
操作成功完成。

C:\Windows\system32>bcdedit /Debug on
操作成功完成。

C:\Windows\system32>bcdedit /dbgsettings
debugtype          Serial
debugport          1
baudrate            115200
操作成功完成。

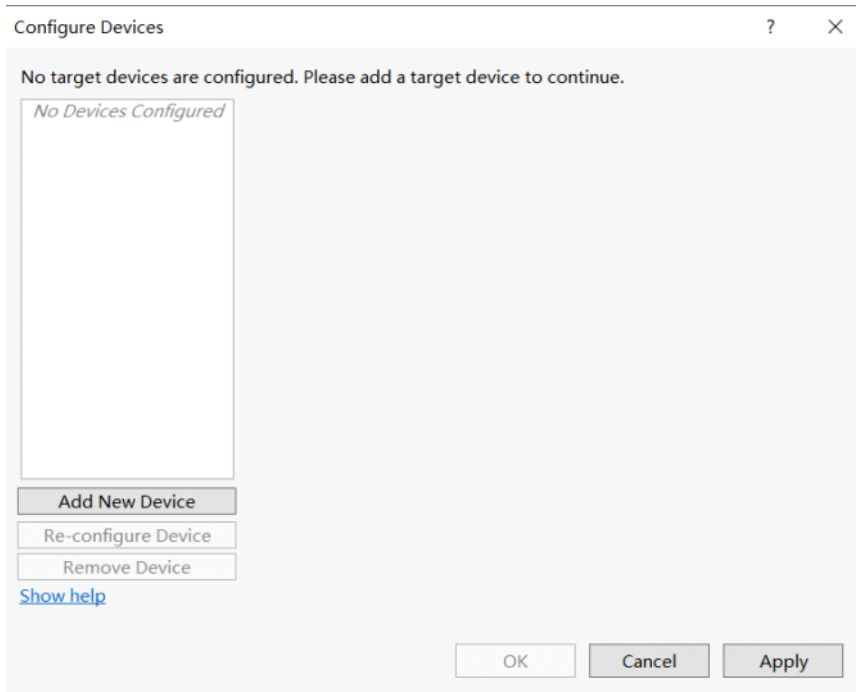
C:\Windows\system32>a
```

2.VS2019设置

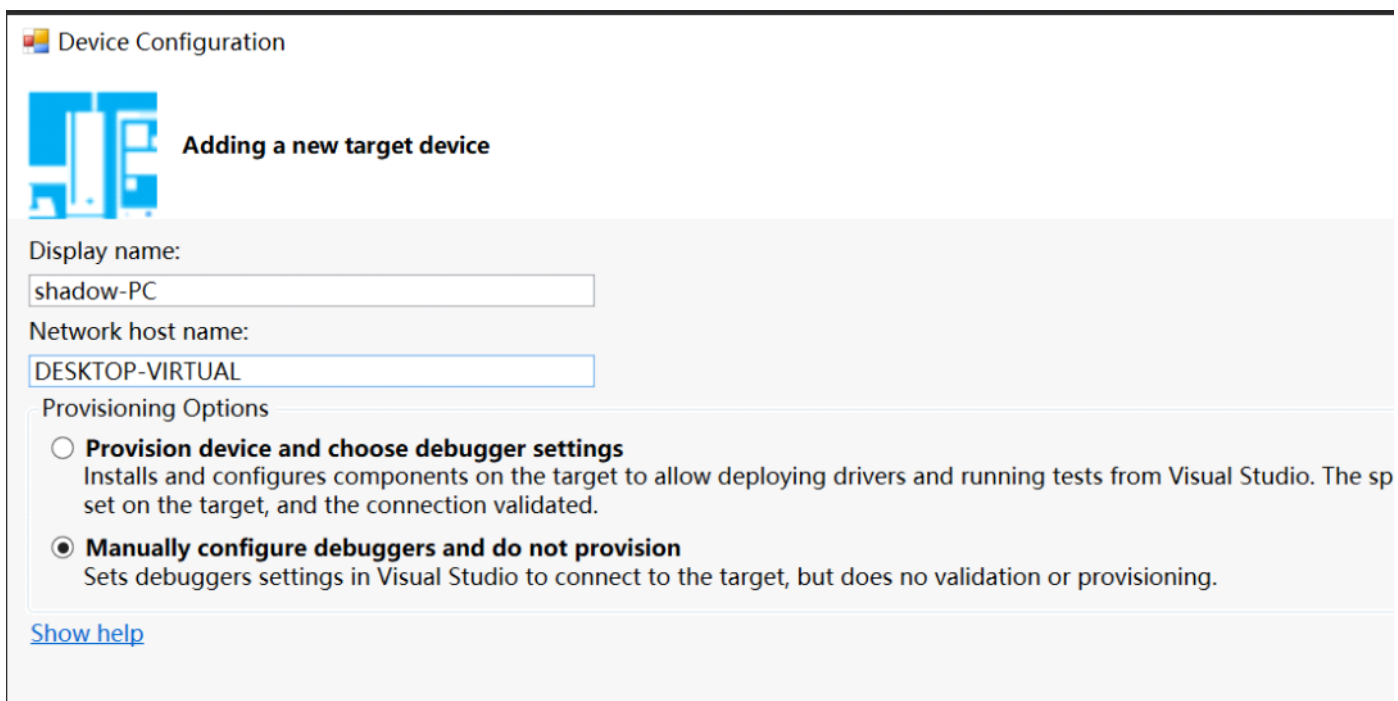
来自 <https://blog.csdn.net/m0_48995611/article/details/112211268>

添加新设备

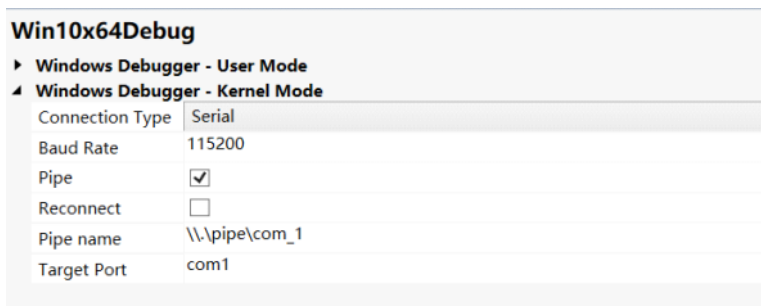
打开 扩展 - Diver - Test - Configure Devices..., 选择 Add New Device。



输入名称，然后下一步。注意：Network host name必须为虚拟机的计算机名。



Kernel Mode中设置串口通信等信息，如下图：



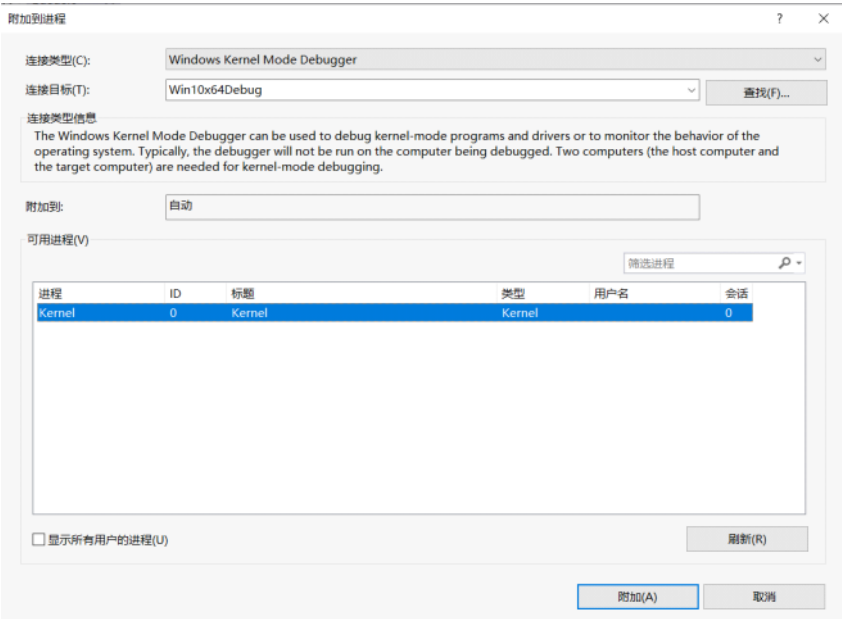
注意：Pipe name要与虚拟机串口设置的管道名称一致。

主机和虚拟机均把防火墙关闭，反正只要保证双方可以相互ping通即可。

3.双机调试

VS2019中，选择 调试 - 附加到进程，连接类型选择 Windows Kernel Mode Debugger，连接目标为之前新建的Device，附加。

注意：附加前要开启虚拟机



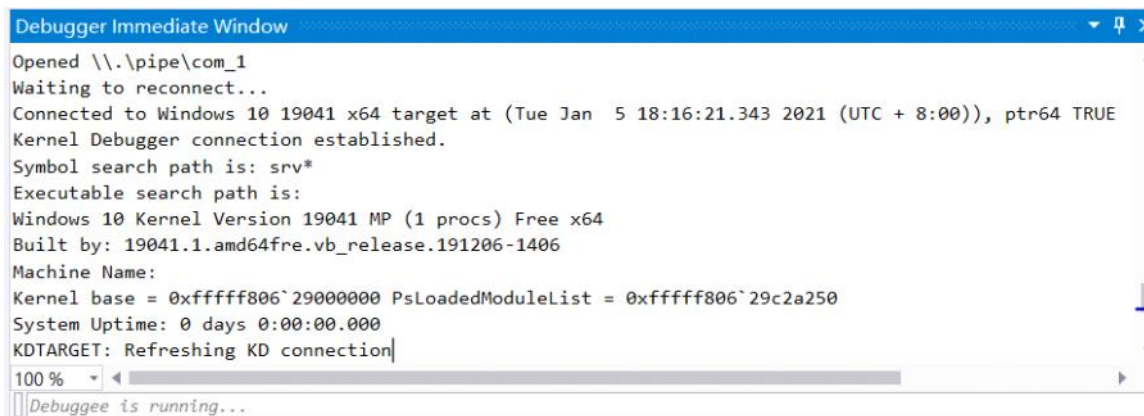
Debug窗口会显示以下字样，表示已打开管道并等待重连。(注意是 Opened \\.\pipe\com_1)



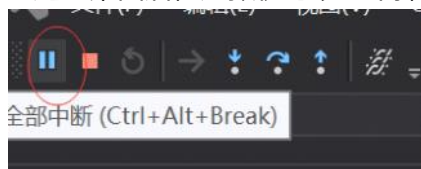
接着重新启动虚拟机，等待虚拟机进入桌面。Debug窗口显示如下：

注意：1.这里可能会花较长时间，并不是因为卡死了。

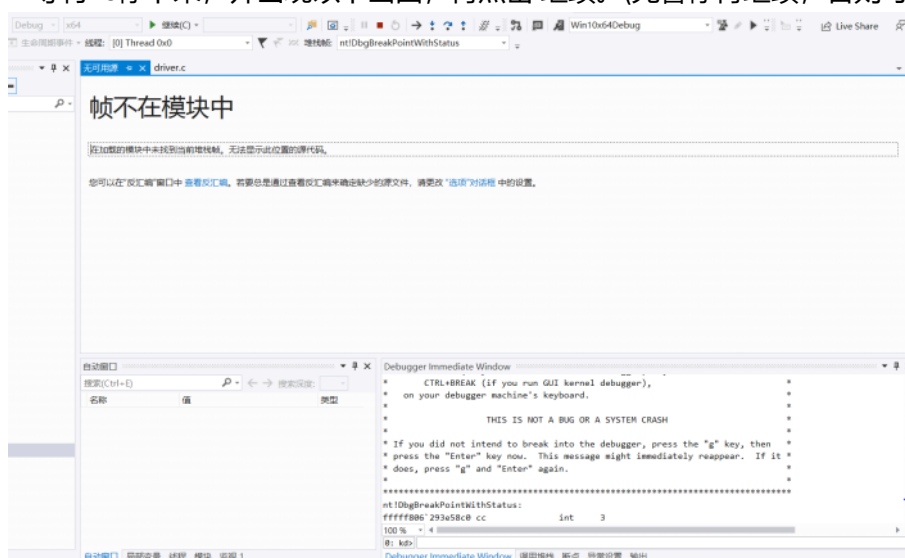
注意：2.重启虚拟机的时候记得禁用驱动程序强制签名。(或者使用testsigning)



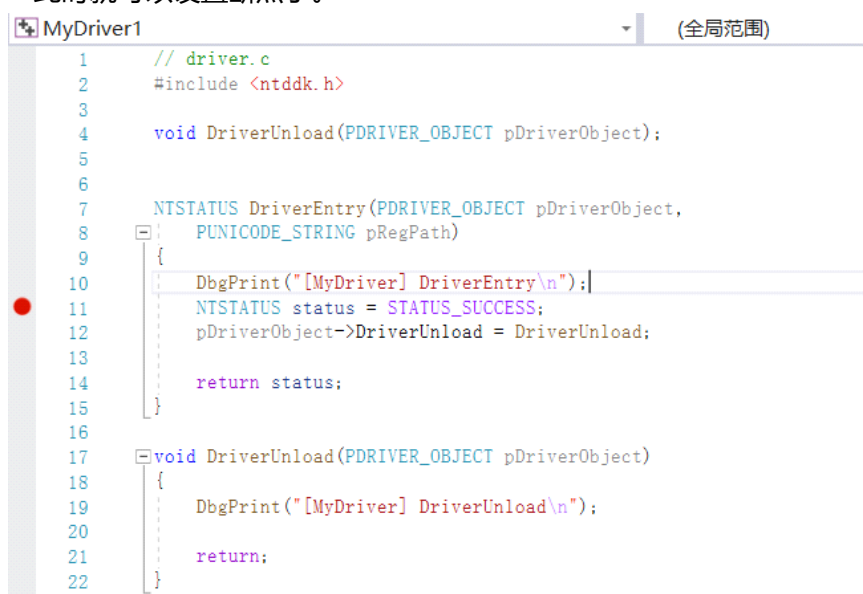
进入桌面后，虚拟机可以正常操作。在vs2019中按下暂停



等待vs停下来，并出现以下画面，再点击 继续。(先暂停再继续，否则可能断不下来)



此时就可以设置断点了。



注意：如果vs很长时间都没有响应，可以用任务管理器结束vs,然后重新打开再附加到进

程。

等待虚拟机可以正常操作后，将主机生成的驱动程序拷贝到虚拟机中，用相关程序加载，就可以成功命中断点。

参考资料：

https://blog.csdn.net/m0_48995611/article/details/112211268