

# WRK 实验环境设置

清华大学电子工程系 马洪兵

(hbma@tsinghua.edu.cn)

## 1. WRK 的安装与设置

WRK 1.2 可以在两种环境下运行：

- X86(Windows Server 2003 Service Pack 1)
- AMD64(Windows XP x64 Professional)

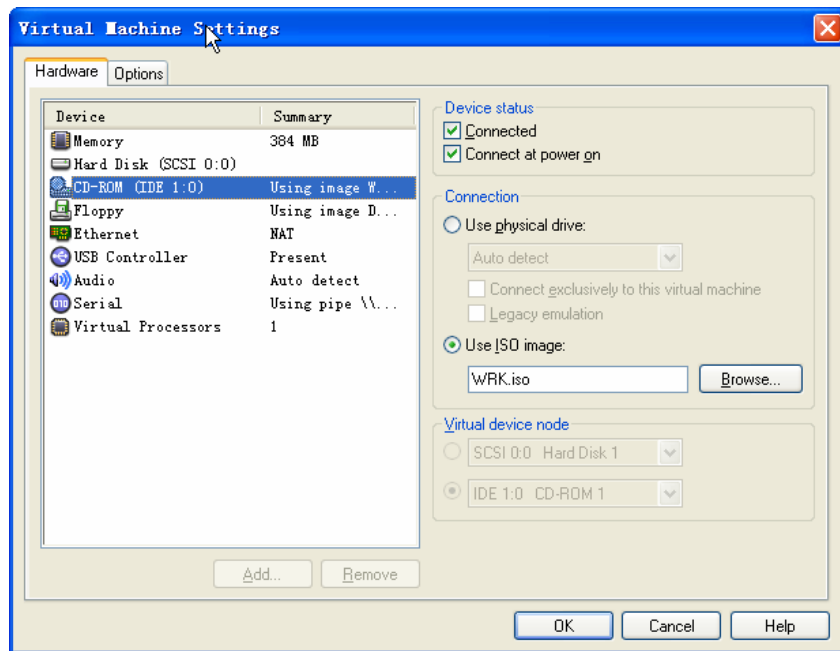
可以在运行上述操作系统的物理计算机上安装 WRK，但是，为了实验方便同时也为了保护物理计算机的操作系统，建议在虚拟机下安装 WRK。

目前最为流行的虚拟机软件有 VMware 和 Virtual PC，这两种软件都可以用来安装 WRK。以下以 VMware 和 X86(Windows Server 2003 Service Pack 1)为例说明 WRK 的安装过程。

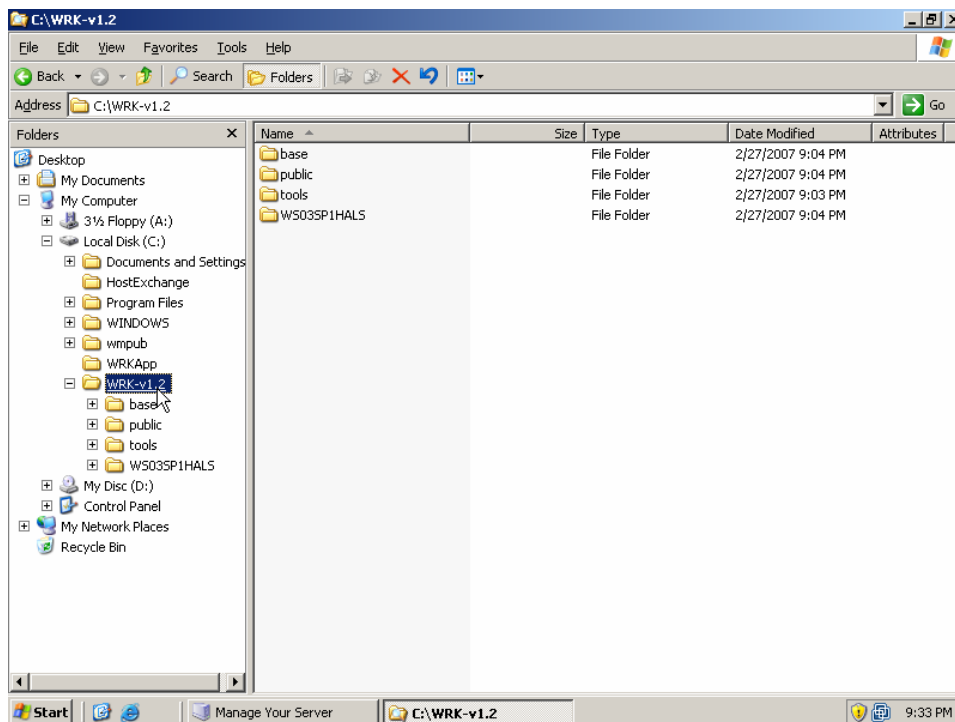
启动 VMware。在 VMware 下新建一个虚拟机，选择客户操作系统类型为 Windows Server 2003 Standard Edition（或者 Windows Server 2003 Enterprise Edition）。

在新建的虚拟机环境下安装 Windows Server 2003 Service Pack 1，安装过程与在物理计算机上安装操作系统完全相同。

下面我们要在虚拟机的 Windows Server 2003 操作系统环境下安装 WRK。首先启动虚拟机操作系统。接下来单击 VMware 的“VM”菜单，选择“Settings...”命令，在出现的 Virtual Machine Settings...对话框中选择 Hardware 选项卡，选择 CD-ROM 设备，将 WRK 光盘映像文件作为虚拟机的光盘，如下图所示。



将虚拟机光驱中 “WRK-v1.2” 目录下的内容复制到虚拟机的硬盘中，如下图所示。



打开控制台窗口，执行下述命令：

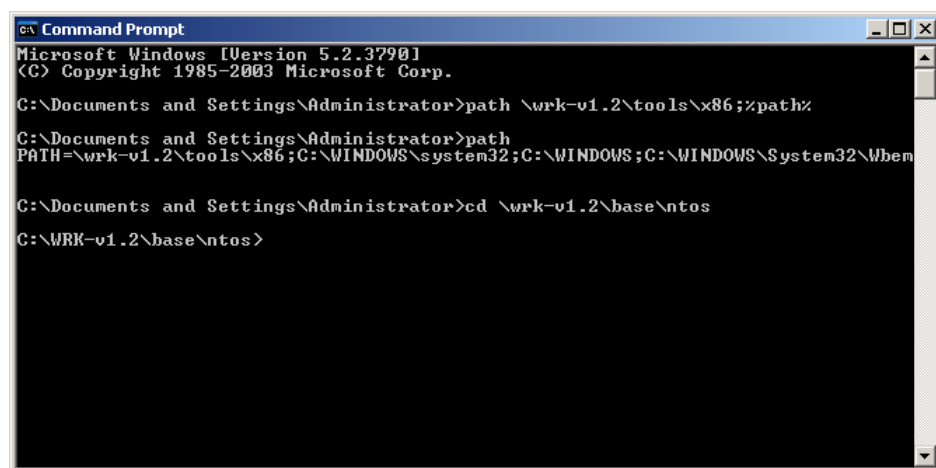
```
path \wrk-v1.2\tools\x86;%path%
```

```
cd \wrk-v1.2\base\ntos
```

其中第一条命令是设置编译环境的文件路径，可以用不带参数的 path 命令

验证路径设置的结果。

命令执行过程如下图所示。



```
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

C:\Documents and Settings\Administrator>path \wrk-v1.2\tools\x86;%path%

C:\Documents and Settings\Administrator>path
PATH=\wrk-v1.2\tools\x86;C:\WINDOWS\system32;C:\WINDOWS\System32\Wbem

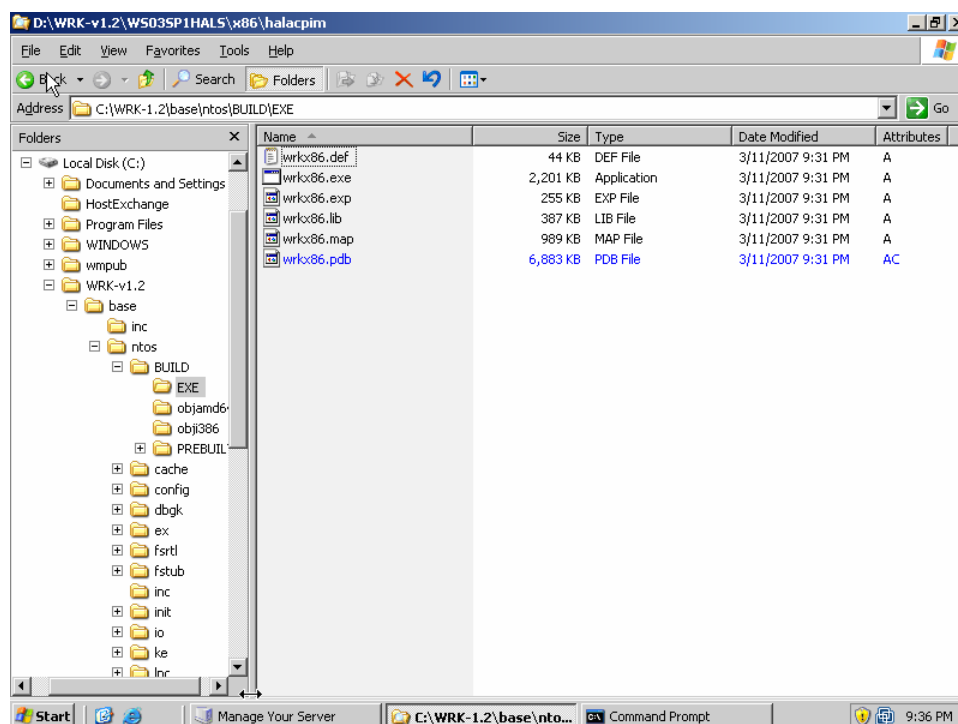
C:\Documents and Settings\Administrator>cd \wrk-v1.2\base\ntos

C:\WRK-v1.2\base\ntos>
```

执行下面的命令将对 WRK 源代码进行编译：

**nmake -nologo x86=**

编译过程大约需要 2 分钟，编译的结果是在\wrk-v1.2\base\ntos\BUILD\EXE目录下生成内核可执行文件和符号文件等，如下图所示。

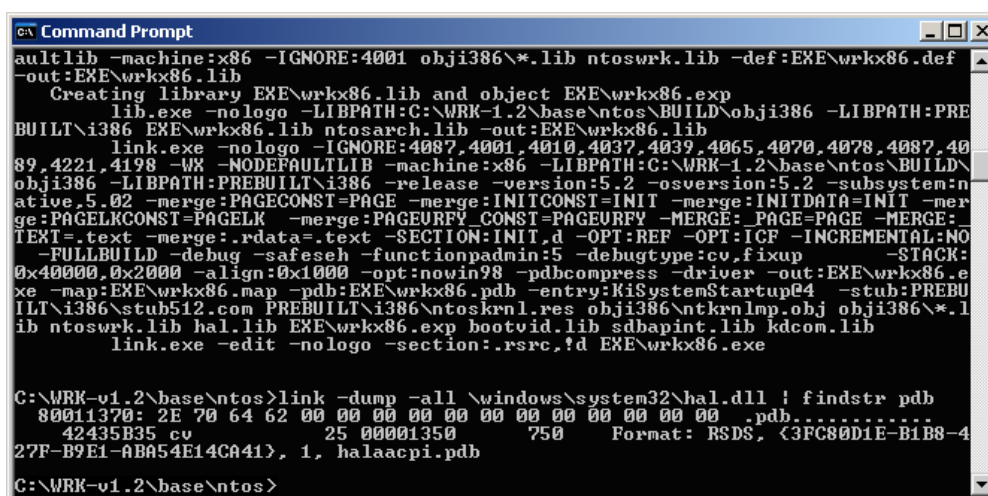


将上述目录中的 WRK 内核文件 wrkx86.exe 复制到\WINDOWS\system32\目录下。

在 X86 平台上，WRK 内核需要多处理器版本的硬件抽象层 hal.dll 的支持。在 Windows 操作系统的发行介质上，提供了许多不同的硬件抽象层文件，安装操作系统时安装程序根据机器硬件的配置选择对应的 HAL 安装到机器中，并将其一律改名为 hal.dll。为了了解机器实际的 HAL 的类型，可以在控制台窗口中执行下述命令：

```
link -dump -all \WINDOWS\system32\hal.dll | findstr pdb
```

根据屏幕输出的内容即可了解机器实际的 HAL 的类型，如下图所示。



```
C:\ Command Prompt
aullib -machine:x86 -IGNORE:4001 obji386\*.lib ntoswrk.lib -def:EXE\wrkx86.def
-out:EXE\wrkx86.lib
Creating library EXE\wrkx86.lib and object EXE\wrkx86.exp
lib.exe -nologo -LIBPATH:C:\WRK-1.2\base\ntos\BUILD\obji386 -LIBPATH:PRE
BUILT\i386 EXE\wrkx86.lib ntosarch.lib -out:EXE\wrkx86.lib
link.exe -nologo -IGNORE:4087,4001,4010,4037,4039,4065,4070,4078,4087,40
89,4221,4198 -WX -NODEFAULTLIB -machine:x86 -LIBPATH:C:\WRK-1.2\base\ntos\BUILD\
obji386 -LIBPATH:PREBUILT\i386 -release -version:5.2 -osversion:5.2 -subsystem:n
ative.5.02 -merge:PAGELKCONST=PAGELK -merge:PAGEURFY_CONST=PAGEURFY -MERGE:_PAGE=PAGE -MERGE:
TEXT=.text -merge:.rdata=.text -SECTION:INIT.d -OPT:REF -OPT:ICF -INCREMENTAL:NO
-FULLBUILD -debug -safeseh -functionpadmin:5 -debugtype:cv,fixup -STACK:
0x400000,0x2000 -align:0x1000 -opt:nowin98 -pdbcompress -driver -out:EXE\wrkx86.e
xe -map:EXE\wrkx86.map -pdb:EXE\wrkx86.pdb -entry:KiSystemStartup@4 -stub:PREBU
ILT\i386\stub512.com PREBUILT\i386\ntoskrnl.res obji386\ntkrnlmp.obj obji386\*.l
ib ntoswrk.lib hal.lib EXE\wrkx86.exp bootvid.lib sdbapint.lib kdcom.lib
link.exe -edit -nologo -section:.rsrc,td EXE\wrkx86.exe

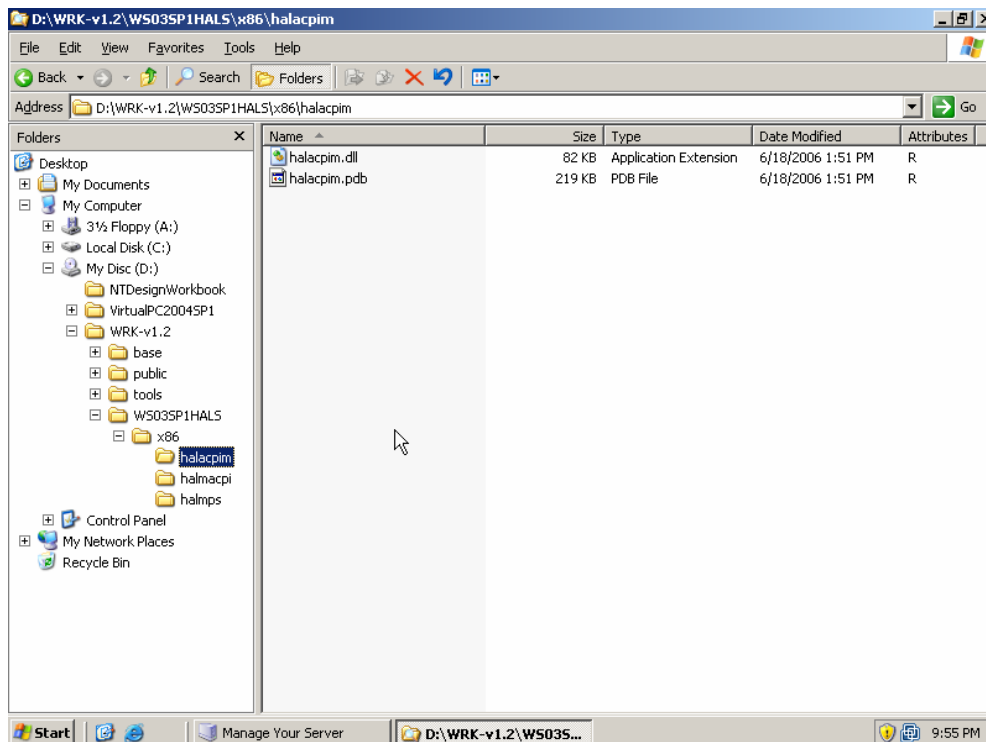
C:\WRK-v1.2\base\ntos>link -dump -all \windows\system32\hal.dll | findstr pdb
80011370: 2E 70 64 62 00 00 00 00 00 00 00 00 00 00 00 00 .pdb.....
42435B35 cv 25 00001350 750 Format: RSDS, {3FC80D1E-B1B8-4
27F-B9E1-ABA54E14CA41}, 1, halaacpi.pdb

C:\WRK-v1.2\base\ntos>
```

如果机器不是多处理器的，那么需要找到与当前 HAL 对应的多处理器 HAL 版本：

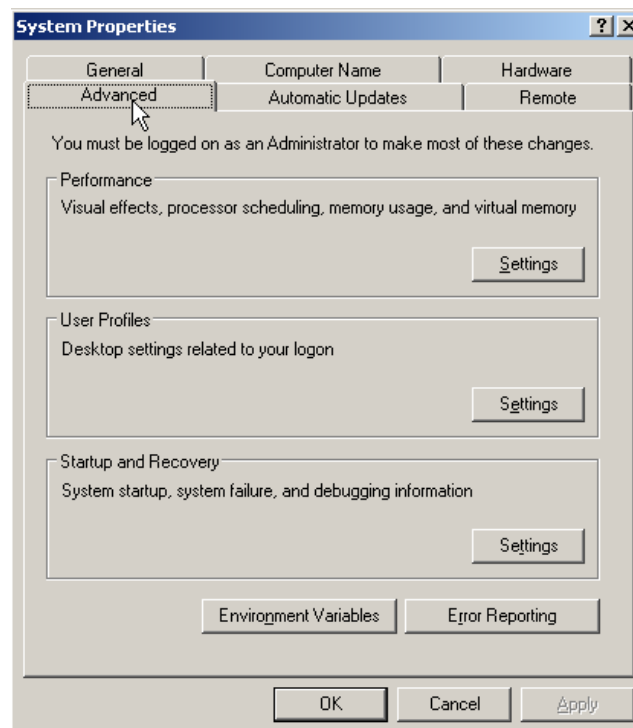
```
halacpi.dll -> halacpim.dll ; ACPI PIC-based PC [used by VirtualPC]
halaacpi.dll -> halmacpi.dll ; ACPI APIC-based PC
halapic.dll -> halmmps.dll ; MPS
```

上述多处理器 HAL 文件在 WRK 光盘的\WS03SP1HALS\x86 目录下，如下图所示。



将 WRK 内核所需要的 HAL 文件（例如 halmacpi.dll）复制到 \WINDOWS\system32\目录下。

下面我们设置 WRK 的引导选项。在“My Computer”上右击鼠标，在弹出菜单中选择“Properties”，打开 System Properties 对话框，如下图所示。

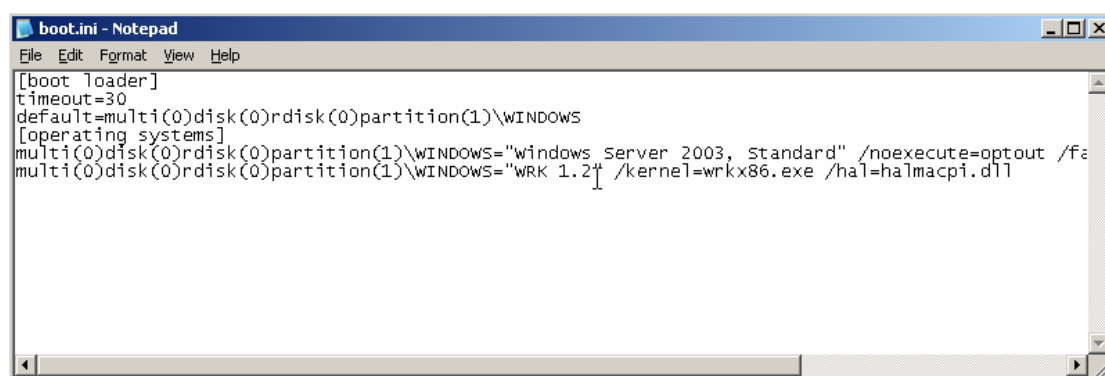


选中 Advanced 选项卡，在“Startup and Recovery”栏中单击“Settings”按钮，弹出“Startup and Recovery”，在其中单击“Edit”按钮，将在 Notepad 中打开 boot.ini 文件进行编辑。

在 boot.ini 文件中增加一个具有如下参数的引导选项：

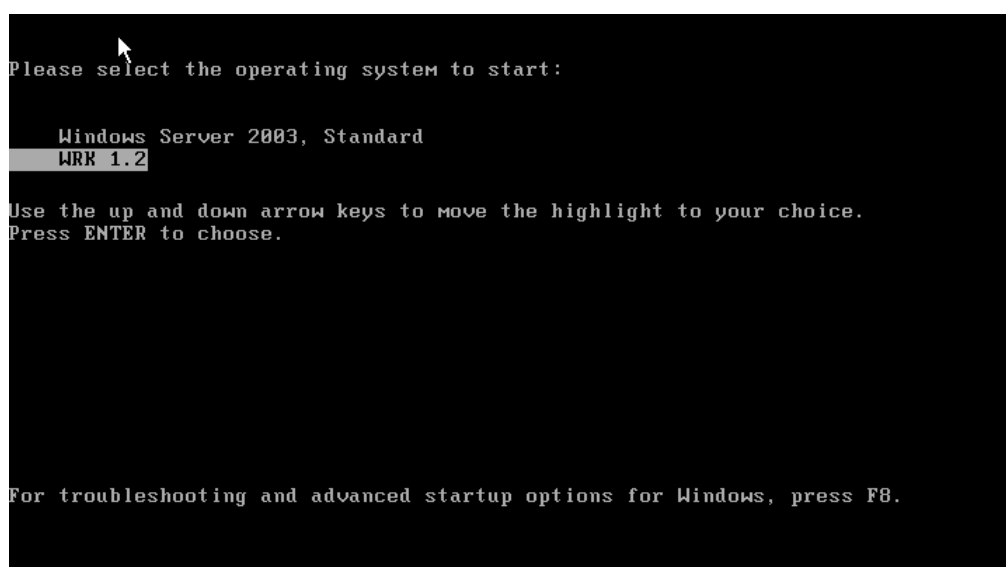
**/kernel=wrkx86.exe /hal=halmacpi.dll**

修改后的 boot.ini 文件的内容如下图所示。



至此，WRK 已经安装和设置完毕，我们可以用 WRK 内核来引导操作系统。

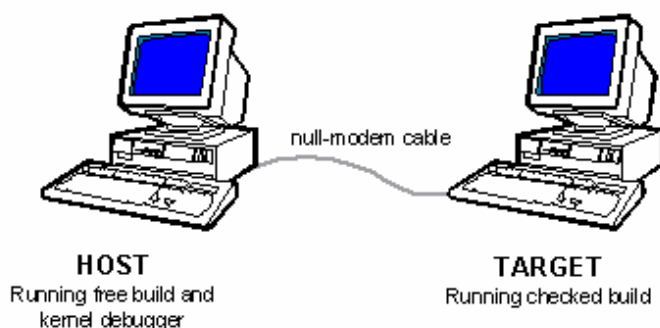
在虚拟机中重新启动操作系统，可以看到新增加了一个引导选项，选择该选项则可以用我们刚刚编译好的 WRK 内核启动系统，如下图所示。



## 2. 调试环境设置

Microsoft 公司的 Debugging Tools for Windows 是一款功能十分强大的调试器，其中包含图形用户界面的调试器 WinDbg，支持操作系统内核级调试，并且

支持 C 语言源代码级调试，可以使用源代码设置各种断点。WinDbg 的使用需要双机环境，宿主机运行调试器，目标机运行被调试的操作系统。运行 WinDbg 时，主机和目标机一般通过无 Modem 串行电缆通过串行口通讯，如下图所示。

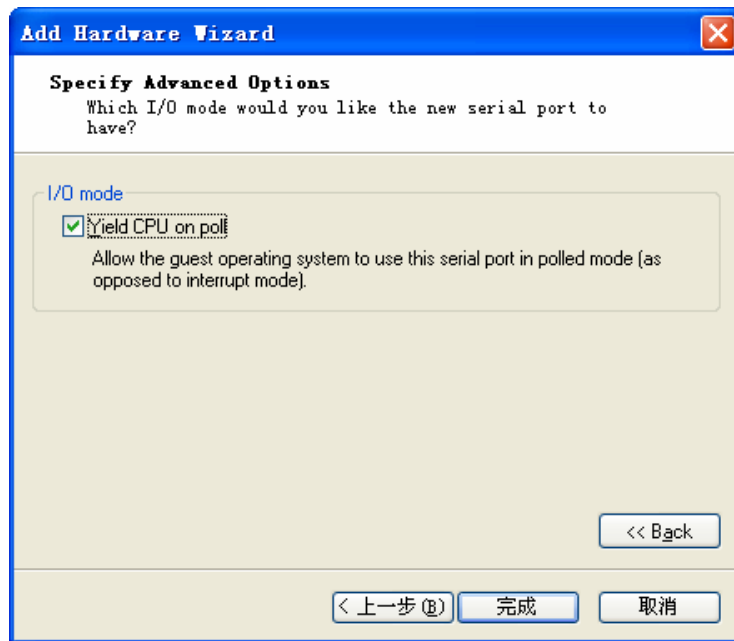


借助于 VMware 软件，我们可以在同一台计算机上的主机和虚拟机之间建立串行通信连接，从而实现调试。

下面我们为虚拟机增添一个串口：单击 VMware 的“VM”菜单，选择“Settings...”命令，在出现的对话框中选择“Add...”按钮，在接下来出现的 Add Hardware Wizard 向导中依次选择增加 Serial Port，串口类型为选择“Output to named pipe”，命名管道的设置可以采用 VMware 提供的缺省值，如下图所示。



单击上图中的“Advanced>>”按钮，在出现的对话框中选中“Yield CPU on poll”，如下图所示。



单击上图中的“完成”按钮，这样我们就为虚拟机增加了一个串口。

为了调试目标操作系统，必须设置 WinDbg 的启动参数，为此可用创建一个新的 WinDbg 快捷方式，其参数设置如下：

```
"D:\Program Files\Debugging Tools for Windows\windbg.exe" -b -k  
com:pipe,port=\\.\pipe\com_1,baud=115200,reconnect -y  
D:\Symbols\WindowsWRK;srv*D:\Symbols\WindowsWRK*http://msdl.  
microsoft.com/download/symbols -srcpath "E:\SourceFile\WRK  
1.2\base"
```

其中各个参数的意义：

- -b——一旦主机目标机之间建立起连接，立刻中断目标机
- -k——内核调试
- com——设置连接目标机的通信端口（此处为命名管道）和波特率（此处为 115200）
- -y——设置符号文件路径
- -srcpath——设置源文件路径

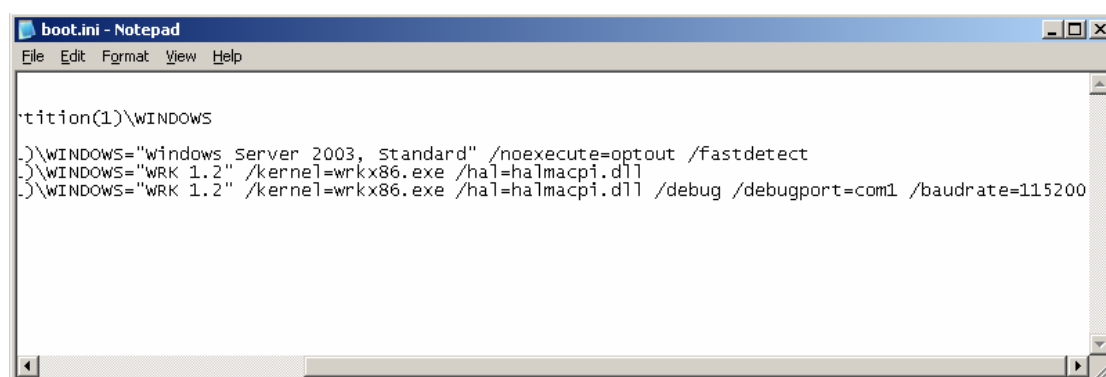
为了在调试时 WinDbg 能够找到 WRK 内核的符号文件，必须把编译 WRK 时生成的符号文件 wrkx86.pdb 复制到主机的符号文件目录中。



为了对目标机进行调试，必须在目标机操作系统中增加新的启动选项。启动虚拟机，编辑 boot.ini 文件，增加一个具有如下参数的引导选项：

```
/kernel=wrkx86.exe            /hal=halmacpi.dll            /debug  
/debugport=com1 /baudrate=115200
```

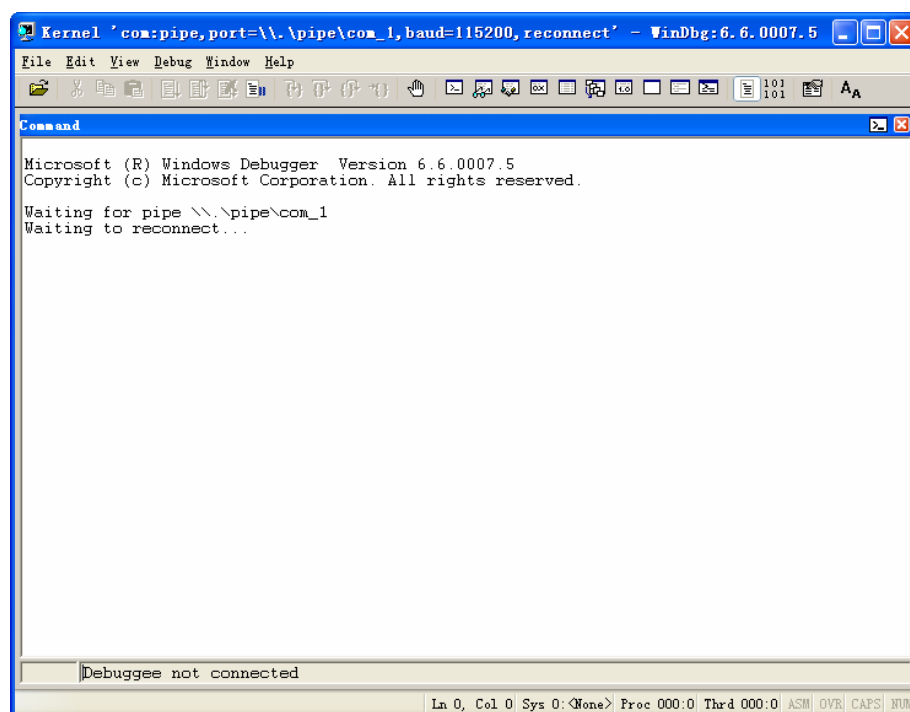
修改后的 boot.ini 文件的内容如下图所示。



修改好 boot.ini 文件后请关闭虚拟机。

利用 WinDbg 调试目标操作系统的步骤如下：

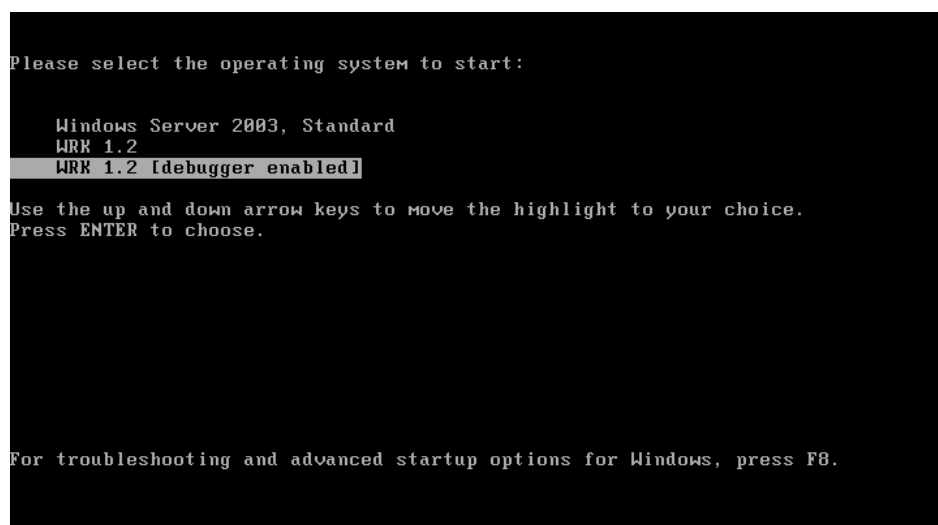
1. 启动 WinDbg，如下图所示。



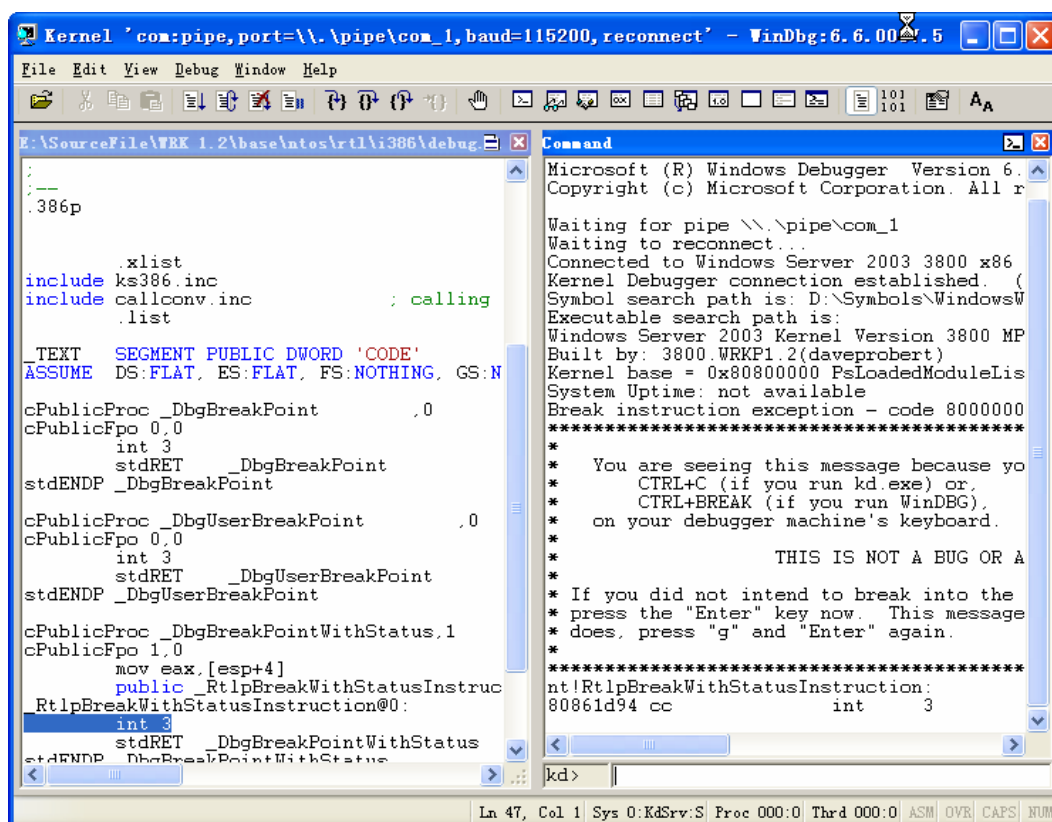
从图中可用看出，WinDbg 正在等待与目标机建立连接。

2. 启动虚拟机，在出现的三个引导选项中，选择“WRK 1.2 [debugger”

enabled]”，如下图所示。



3. 随着虚拟机操作系统的启动，主机和目标机之间建立起了连接。连接一旦建立， WinDbg 就会立刻中断目标机的操作系统，并调出断点所在的源文件。连接建立后， WinDbg 的反应如下图所示。



4. 在 WinDbg 中按下 F5 键，恢复目标机操作系统的运行。

5. 在目标机运行过程中，在 WinDbg 中按下 CTRL+BREAK 键，则立刻中

断目标机的操作系统，从而可用对目标机进行各种调试工作。在调试过程中，按下 F5 键，则恢复目标机操作系统的运行。