

# 基于 TI-Davinci DM6446 开发平台的网络文件系统(NFS)搭建和测试

邹腾剑 邓少彬

(西南科技大学 四川 绵阳 621000)

**[摘要]** DM6446 开发平台是美国德州仪器(Texas Instruments)公司生产的一款基于 DSP 和 ARM 的开发平台,该平台适用于图像处理、多媒体处理、通讯系统,数字信号处理以及嵌入式系统等领域。本文研究的是如何通过配置 DM6446 开发平台的 bootloader 来搭建一个网络文件系统,从而使 DM6446 开发平台能够通过网络共享 PC 机上的资源,最后通过一个测试程序来验证网络文件系统的功能。

**[关键词]** DM6446 NFS bootloader Texas Instruments

中图分类号:TP606

文献标识码:A

文章编号:1009-914X(2009)35-0204-02

## 1 DVEVM 开发环境的搭建

命令行说明:

host\$: 主机 linux 下输入命令(su user、su root 切换普通用户和超级用户)

EVM#: U-boot 控制窗口输入命令

target\$: 目标板系统下输入命令.

### 1.1 通过虚拟机安装 Linux 系统

目标板上已经安装好了 Montavista Linux Pro v4.0

我们可以先在 PC 上安装以下支持使用 DVEVM 的 linux 主机操作系统,可以用虚拟机安装 linux,也可以单独安装 linux:

我们的方法是通过安装 VMware Workstation 虚拟机,启动虚拟机后,再通过虚拟光驱加载三个 redhat9.0 的镜像安装包,从而实现在虚拟机上安装 linux 系统。

### 1.2 在虚拟机的 linux 系统中安装相应的软件

按以下步骤安装主机 linux 软件:

首先从 DAVINCI 四张光盘中拷贝出以下文件暂时存放在 /tmp 下:

\* mvl\_setuplinux\_#\_#\_#.bin(##代表版本号)

\* mvl\_target\_setuplinux\_#\_#\_#.bin

\* mvl\_lsp\_setuplinux\_#\_#\_#.bin

\* dvevm\_setuplinux\_#\_#\_#.bin

(1) 进入用户主机 linux 的 /tmp 下执行 .bin 文件安装(可以默认安装在 /opt 下):

\* host\$ ./mvl\_setuplinux\_#\_#\_#.bin

\* host\$ ./mvl\_target\_setuplinux\_#\_#\_#.bin

\* host\$ ./mvl\_lsp\_setuplinux\_#\_#\_#.bin

注意:如果在安装的过程提示不能用图形化安装,可以用 ./mvl\_setuplinux\_#\_#\_#.bin -console 在控制台按照提示安装(普通用户不能执行切换到 root 执行,下同)

(2) 执行上面的文件后确认以下文件出现在 /opt/mv\_pro\_4.0 下:

\* mvltools4.0-no-target.tar.gz;

\* mvl4.0-target\_path.tar.gz;

\* DaVinciLSP#.#.tar.gz。

(4) 在 /opt/mv\_pro\_4.0 下解压:

\* host\$ tar xzf mvltools4.0-no-target.tar.gz

\* host\$ mvl4.0-target\_path.tar.gz

\* host\$ DaVinciLSP#.#.tar.gz

在 /opt/mv\_pro\_4.0 下建立了 Montavista 目录结构。

### 1.3 安装 DVEVM 软件

\* host\$ cd /tmp

\* host\$ ./dvevm\_setuplinux\_#\_#\_#.bin (可能要在控制台安装,参照第二步当中所用到的方法)

注意:当出现提示符时,不要使用默认安装,安装在用户所使用的目录里。比如:你的普通用户账户名叫 user,你就可以安装在 /home/user 下,在 user 下就有 dvevm\_#\_#\_# 这个目录。到此为止,你可以将 /tmp/ 下的 .bin 文件删除。以免占用过多的空间。

### 1.4 开辟一个共享的文件系统,供目标板使用

尽管评估板硬件驱动器中已经包含了一个文件系统,但是我们可以为主

机 linux 下增加一个供目标板使用的网络文件系统(NFS),为开发提供方便。

先将目标文件系统装在主机 linux 某个目录下:

(1) 使用普通用户账号(user)登入主机 linux

(2) 执行以下命令,为 Montavista 文件系统准备位置

host\$ cd ~

host\$ mkdir -p workdir/filesys

host\$ cd workdir/filesys

(3) 切换到超级用户

host\$ su root

(4) 复制目标文件系统到共享区域:

host\$ cp -a /opt/mv\_pro\_4.0/montavista/pro/devkit/arm/v5tle/target/\* ./

host\$ chown -R user opt

(5) 用命令 gedit /etc/exportfs 或 vi /etc/exportfs 在主机 linux 下编辑 /etc/exportfs 文件 添加以下行:

/home/user/workdir/filesys \*(rw,no\_root\_squash,no\_all\_squash, sync) 保存退出

(6) 依然在 root 用户下,将 NFS 服务器改变为上述(5)中配置

host\$ /usr/sbin/exportfs -a

host\$ /sbin/service nfs restart

### 1.5 测试共享文件系统

(1) host\$ /sbin/ifconfig 获得主机 linux 的 ip 地址,eth0 后的第一个地址

(2) 用两根网线把 pc 机和目标板同时连入路由器外网,板子和 pc 之间用串口线相连 打开超级终端(进入超级终端时的设置只要把第一行的波特率从 2400 为 115200)

(3) DVEVM 板加电,在超级终端控制窗口上按任意键,终止自动引导。

在控制窗口设置环境变量(用 NFS 来引导评估板启动):(IP:dhcp)

EVM# setenv nfshost 192.168.0.113 (该地址是步骤(1)中获得的地址)

EVM# setenv rootpath /home/user/workdir/filesys

EVM# setenv bootargs console=ttyS0,115200n8 noinitrd rw ip=dhcp root=/dev/nfs nfsroot=\$(nfshost):\$(rootpath),nolock mem=120M

EVM# s2veenv

EVM# boot

此时启动开发平台上的操作系统,开发平台上的文件系统就是主机 linux 下 /home/user/workdir/filesys 目录下的文件系统映射过去的,而不是自身硬盘驱动器下的文件系统。比如我们在主机 linux 下 /home/user/workdir/filesys 目录下创建了一个文件夹 aaa,在超级终端进去的开发平台系统中 /mnt 下自动生成 aaa 文件夹。达到共享的目的,方便开发。

### 2 静态 IP 启动

从 NFS 引导文件系统(IP:static) (注:用一根专用网线把主机 linux 和目标板连起来,启动时网线水晶头闪烁)

环境变量的设置



DVEVM 板加电, 在超级终端控制窗口上按任意键, 终止自动引导

```
EVM# setenv serverip 192.168.1.115    (设置主机linux的静态ip)
EVM# setenv ipaddr 192.168.1.112    (设置板子ip)
EVM# setenv gateway 192.168.1.97
EVM# setenv netmask 255.255.255.0
EVM# setenv dns1 156.117.126.7
EVM# setenv dns2 157.170.1.5
EVM# setenv myip $(ipaddr):$(gateway):$(netmask):$(dns1):$(dns2)::off
EVM# setenv nfspath /home/user/workdir/filesys
EVM# setenv bootcmd bootm 0x2050000
EVM# setenv bootargs console=ttyS0,115200n8 noinitrd rw ip=$(myip) root=/dev/nfs nfsroot=$(serverip):$(nfspath),nolock mem=120M
EVM# saveenv
EVM# boot
```

注:引导启动的时候,如果出现looking for serverip 192.168.1.113 failed……

此时我们可以到 主机linux下 设置eth0 的ip为 192.168.1.113 具体如下:

```
host$ /sbin/ifconfig
host$ /sbin/ifconfig eth0 down
host$ /sbin/ifconfig eth0 192.168.1.113
host$ /sbin/ifconfig (查看eth0 后是否有出现192.168.1.113)
host$ /sbin/service network restart (出现...ok 有failed)
重新按上述步骤设置,也可以先到application->system setting->network设置eth0:192.168.1.113)
```

### 3 测试应用程序

自己拷贝已有的应用程序工具包workshop到/home/user/ 进入主机linux应用程序目录:/home/user/workshop 进入某个lab ##-## 目录 ./runxdc.sh install (如果./runxdc.sh 后面没有install,就只生成在本身目录下的app下,没有传到目标板上,需要手动拷贝到共享目录下)

这时候会出现error:can't find the path :/opt/montavista/.... 解决方法:

- 1)进入/opt/workshop:用命令gedit setpath.sh打开编辑里面有一行:export MONTAVISTA\_DEVKIT=/opt/montavista/pro/devkit,把它改为:export MONTAVISTA\_DEVKIT=/opt/mv\_pro\_4.0/montavista/pro/devkit
- 2). 把/opt/mv\_pro\_4.0下的montavista移到/opt下再运行./runxdc.sh install 具体如下:

```
host$ cd /
host$ mv -R /opt/mv_pro_4.0/montavista /opt (需要以root用户来执行)
```

如果还有错误:no rule to make target 'install'.换一个以“lab”开头的目录进去后./runxdc.sh install 直到找到可以运行成功的目录下文件。(有的会有错误:no rule to make target 'install',因为它的app下的package.bld中没有定义install,可以对比“能运行./runxdc.sh install的app下的package.bld”,用gedit package.bld分别查看,在有错误的app下的package.bld中加上定义install的内容,这样就可以执行后面的install了)

自动把编译好的可执行文件拷贝到共享文件系统/home/user/workdir/filesys/opt/workshop(先要进入板子系统/opt下或者主机linux下的/home/user/workdir/filesys/opt/下建立文件夹workshop),此时你就可以在超级终端进入板子系统:/opt/workshop下执行生成的可执行文件。

注意:以上的这些操作必须保证PC机上的linux上安装GCC和交叉编译器,可以再application下的添加删除程序补充安装。

已有的应用程序工具包workshop拷贝到主机linux目录/home/user/下、在主机linux下的/home/user/workdir/filesys/opt/下建立文件

夹workshop, /home/user/workshop/lab ##-## 目录 ./runxdc.sh install 生成的可执行文件自动映射到/home/user/workdir/filesys/opt/workshop 也可以自己设置路径让生成的可执行文件自动映射到/home/user/workdir/filesys/下的其它目录(但是一定要映射在/home/user/workdir/filesys/下的共享目录,板子上系统才可以识别到。比如/home/user/workdir/filesys/mnt)

那怎样设置路径的让生成的可执行文件自动映射到/home/user/workdir/filesys/下的其它目录了呢?下面介绍一下方法:

打开/home/user/workshop 里面有一个setpaths.sh 和一个config.bld 用命令gedit setpaths.sh(只编辑这一个好像也可以)和gedit config.bld 查看内容,最后几行出现:export /home/user/workdir/filesys/opt/workshop之类的,你把这行最后的/opt/workshop改成其它目录,比如改成/mnt,执行./runxdc.sh install后,自动把编译好的可执行文件拷贝到共享文件系统/home/user/workdir/filesys/mnt下了,进入板子系统/mnt下可以看到多出几个文件,执行即可。

(注意:如果你之前已经执行过该文件,需要先./runxdc.sh clean 再执行./runxdc.sh install才能看到/mnt下的新的结果)

### 4 结果以及分析

在NFS搭建完毕之后,我们在PC机上的linux对应的目录下新建一个文件,然后通过超级终端来查看,结果发现对应的目录下已经新建的一个文件,这就说明了DM6446开发平台启动的文件系统是PC机上linux对应目录下的那个文件系统,这种方法可以使得开发平台与PC机共享有用的资源,节约了开发成本,同时降低了开发难度。

将测试程序拷贝到网络文件系统之后,我们通过交叉编译某些实例程序,然后在开发平台的环境下运行,我们可以看到对应的实验结果,这说明我们可以在PC机上编写程序和调试程序,最后将可执行文件移植到开发平台上去执行,这种方法既方便又节约成本,同时由于资源的共享,还可以缩短产品的开发周期。

### 参考文献

- [1] 晓飞,刘彬,张辉.基于ARM 嵌入式Linux 应用开发与实例教程[M].北京:清华大学出版社,2007.
- [2] 韦东山.嵌入式linux应用开发完全手册.人民邮电出版社,2008.