

【Linux 学习系列二：运行 Hello World】

2019-04-02

目录

1.引言	3
2.硬件平台选择	3
3.Ubuntu 下运行 Hello World	4
4.主机和虚拟机之间复制、粘贴、共享文件夹	6
5.交叉编译	9
6.结束语	14

www.topsemic.com

1. 引言

前天写的文章《Linux 学习系列一：开发环境搭建》，放到阿莫论坛里没想到第二天就变成火帖了，看来还是有很多人对这个主题挺感兴趣的。借着大家的鼓舞，趁热打铁，今天再写一篇。今天给大家介绍的是在一个嵌入式 Linux 板子上运行 Hello World，入门学习的经典操作。依稀记得两年前，我第一次在飞思卡尔（现 NXP）IMX6 上运行出 Hello World 的那种喜悦之情。希望这篇文章也能对你有帮助。

2. 硬件平台选择

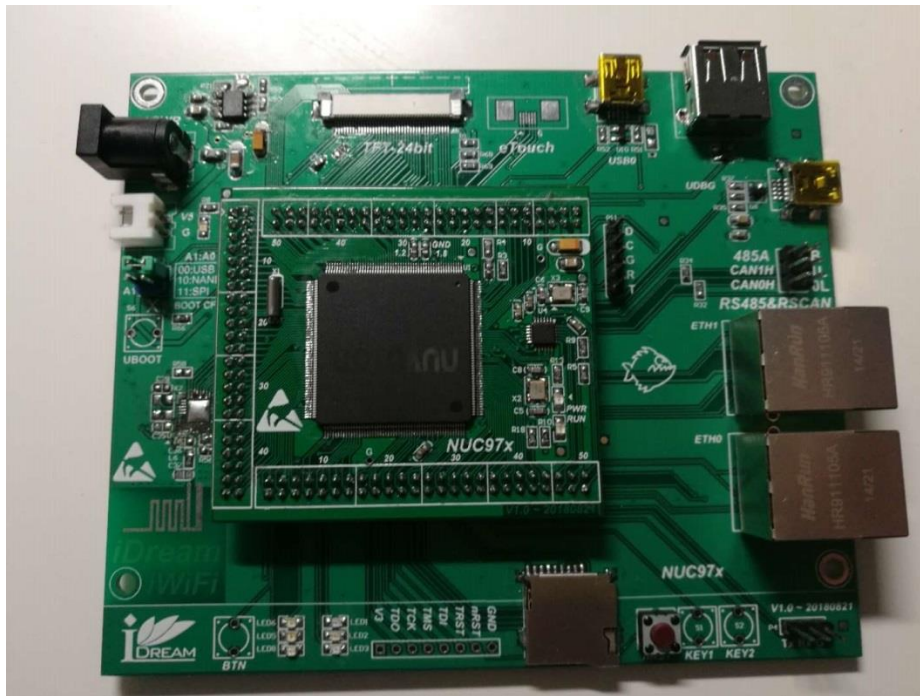
能够运行 Linux 系统的嵌入式硬件平台，我了解的有这么几个，一是三星的 2440 系列，二是 TI 的 AM335X 系列，三是 NXP 的 IMX6 系列，四是瑞芯微 RK3xxx 系列。对于初学者该如何选择呢？这个仁者见仁、智者见智，我这里选择的不是上述四种，而是新唐科技 NUC972，原因有以下几点：

1) 该芯片内置 64MB SDRAM，外部只要加一个 NAND FLASH 就能工作，所以最少只要 2 层板就可以完成布线，另外它的封装为 LQFP，自己手工就可焊接，其他的能跑 Linux 的芯片，没个 6 层、8 层板是搞不定的，另外也不好手工焊接。所以单凭这一点，就能俘获一大批原来做单片机的工程师的心，因为很多做单片机的都喜欢自己捣鼓画个板实际去调试，毕竟这样会很有成就感。

2) 该芯片为工业级芯片，其资源相当丰富，网口、串口、LCD 接口、CAN 接口等应有尽有，最关键的是它的价格相当亲民，只要大约三十块人民币，所以你如果想自己画板焊接调试，也花费不了多少钱。

3) 新唐作为台湾公司，官方给的资料也有中文资料，看起来会比较方便。

其实还有一个主要原因是，我有个前同事，嵌入式骨灰级玩家，他自己做了一块 NUC972 的板子，因为他调过，我就要过来玩了。<http://www.mcufly.com/> 这是他的个人网站，感兴趣的朋友可以看看。下图是他做的板子。



所以我后面的系列文章，都以这个 NUC972 平台为例来讲解了。如果你用的是其他的板子也没太大问题，因为很多东西都是类似的。就像你会了 STM32 单片机开发，再用其他家芯片也是很容易上手的。

3. Ubuntu 下运行 Hello World

在 NUC972 板子运行程序之前，我们先在 Ubuntu 下运行一个 Hello World。要运行此程序，需要经过编辑->编译->运行三个步骤，

第一步，编辑代码，单片机常用 KEIL 软件，此 IDE 集编辑、编译、下载、调试于一体，搞 Linux 的用什么呢？我见过的有以下几种方式：

- a) 在 Windows 下用 Source Insight、QT、VS 等软件编辑，再把文件放到虚拟机里编译
- b) 直接在 Ubuntu 里用 Eclipse 软件编辑
- c) 直接在 Ubuntu 里用 vi/vim 编辑代码 (vim 是 vi 的升级版)

这个看大家喜好，选择一种自己喜欢的就行，我其实这几种都尝试过，最后选择了在 Windows 下用 QT 的方式，原因是方案 b) 由于我电脑性能较差，在虚拟机下使用 Eclipse 一会就会卡机，放弃了；方案 c) 虽然操作起来很酷，但是对我这种菜鸟使用实在是别扭，当然用熟了会很有效率。至于为什么在 Windows 选择 QT，主要原因是我个人觉得 QT 的色彩搭配和默认字体我比较喜欢。

因为咱们这个 Hello World 没几行代码，我就直接在 Ubuntu 下用 gedit 打开编写了。复习上一篇文章的操作，在/home/topsemic 文件夹下新建一个 hello 文件夹，用 gedit 打开 hello.c ，编辑代码保存。

```
topsemic@topsemic-virtual-machine:~$ cd /home/topsemic/  
topsemic@topsemic-virtual-machine:~$ mkdir hello  
topsemic@topsemic-virtual-machine:~$ cd hello/  
topsemic@topsemic-virtual-machine:~/hello$ gedit hello.c  
topsemic@topsemic-virtual-machine:~/hello$
```

```
打开(O) ▾ [F4] 保存(S)  
#include<stdio.h>  
void main()  
{  
    printf("Hello World\n");  
}
```

第二步，编译代码，使用 gcc 指令，gcc 是 linux 系统集成的编译器，这个一装完 Ubuntu 系统里面就有。最简单的用法是 gcc hello.c，它会默认生成 a.out

```
topsemic@topsemic-virtual-machine:~/hello$ gcc hello.c  
topsemic@topsemic-virtual-machine:~/hello$ ls  
a.out hello.c  
topsemic@topsemic-virtual-machine:~/hello$
```

你如果要指定生成的应用程序名字，可以用 gcc hello.c -o hello

```
topsemic@topsemic-virtual-machine:~/hello$ gcc hello.c -o hello  
topsemic@topsemic-virtual-machine:~/hello$ ls  
a.out hello hello.c  
topsemic@topsemic-virtual-machine:~/hello$
```

这样就生成 hello 文件，如果你想知道 gcc 更多详细的用法，可以输入 man gcc 获得帮助。

```
topsemic@topsemic-virtual-machine:~/hello$ man gcc
```

```
topsemic@topsemic-virtual-machine: ~/hello
GCC(1) GNU GCC(1)
NAME
gcc - GNU project C and C++ compiler
SYNOPSIS
gcc [-c|-S|-E] [-std=standard]
    [-g] [-pg] [-Olevel]
    [-Wwarn...] [-Wpedantic]
    [-Idir...] [-Ldir...]
    [-Dmacro[=defn]...] [-Umacro]
    [-foption...] [-mmachine-option...]
    [-o outfile] [@file] infile...
Only the most useful options are listed here; see below for the
remainder. g++ accepts mostly the same options as gcc.
DESCRIPTION
When you invoke GCC, it normally does preprocessing, compilation,
assembly and linking. The "overall options" allow you to stop this
process at an intermediate stage. For example, the -c option says not
to run the linker. Then the output consists of object files output by
the assembler.
Manual page gcc(1) line 1 (press h for help or q to quit)
```

第三步，运行程序，执行 `./a.out`，`./` 表示的是当前目录，结果是打印 Hello World。

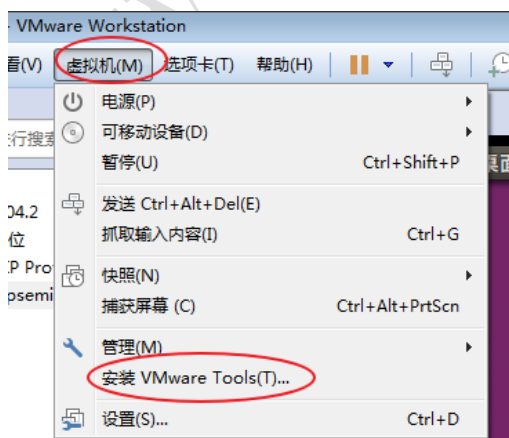
```
topsemic@topsemic-virtual-machine:~/hello$ ./a.out
Hello World
topsemic@topsemic-virtual-machine:~/hello$
```

4. 主机和虚拟机之间复制、粘贴、共享文件夹

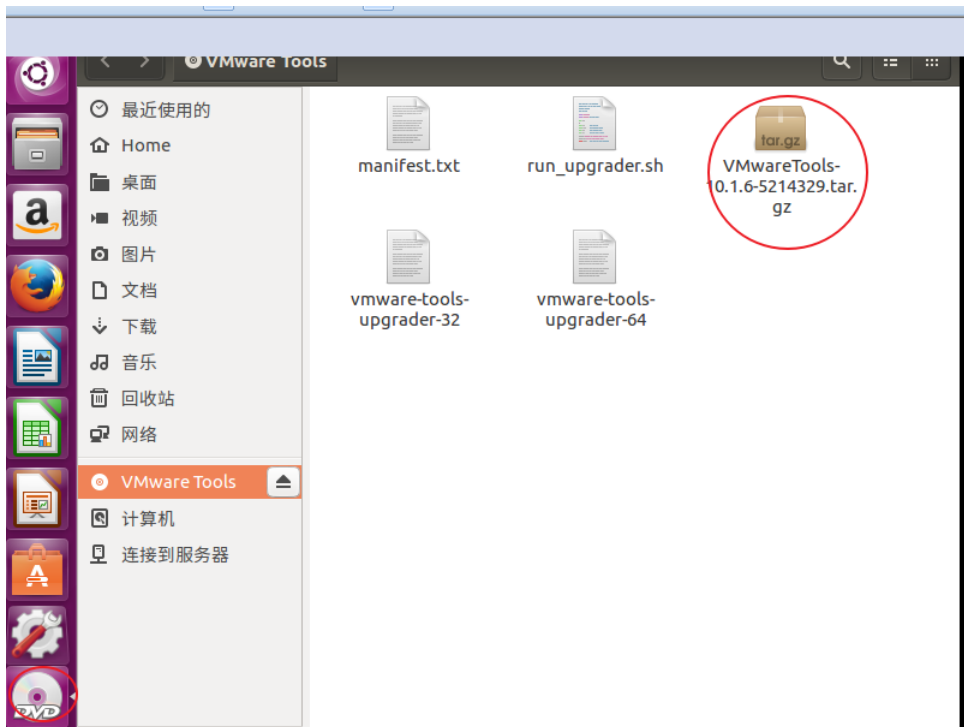
不知道你们有没有试过将 Windows 主机里的内容尝试复制粘贴到虚拟机里，在默认情况下这是不允许的。该如何实现该功能呢？

具体步骤如下：

1) 点击虚拟机-安装 VMware Tools



2) 在左侧工具栏 DVD 里会出现 VMwareTools-xxxx.tar.gz 的文件



3) 将这个文件先拷贝到/tmp 目录

```
topsemic@topsemic-virtual-machine:~$ cp /media/topsemic/VMware\ Tools/VMwareTools-10.1.6-5214329.tar.gz /tmp
```

4) 进入到/tmp 目录，解压 gz 文件，解压命令为 tar 指令

```
topsemic@topsemic-virtual-machine:~$ cd /tmp
topsemic@topsemic-virtual-machine:/tmp$ tar -xzf VMwareTools-10.1.6-5214329.tar.gz
```

方便大家解压或者压缩文件，我把常用格式的解压和压缩指令给整理出来了，主要原因是我总是记不住，记录下来就不用老上网搜索了。

格式	解压指令	压缩指令
gz	tar -zxvf file.tar.gz	tar -czf jpg.tar.gz *.jpg
bz2	tar -jxvf file.tar.gz	tar -cjf jpg.tar.bz2 *.jpg
tar	tar -xvf file.tar	tar -cvf jpg.tar *.jpg
zip	unzip file.zip	zip jpg.zip *.jpg

5) 进入到 vmware-tools-distrib 目录

```
topsemic@topsemic-virtual-machine:/tmp$ cd vmware-tools-distrib/
```

超级用户 **sudo** 执行 vmware-install.pl 文件：

```
topsemic@topsemic-virtual-machine:/tmp/vmware-tools-distrib$ sudo ./vmware-install.pl
```

开始安装，第一个 no 后面输入 yes，后面一路回车即可

```
topsemic@topsemic-virtual-machine:/tmp/vmware-tools-distrib$ sudo ./vmware-install.pl
open-vm-tools packages are available from the OS vendor and VMware recommends using open-vm-tools packages. See http://kb.vmware.com/kb/2073803 for more information.
Do you still want to proceed with this installation? [no] yes
Creating a new VMware Tools installer database using the tar4 format.

Installing VMware Tools.

In which directory do you want to install the binary files?
[/usr/bin]

What is the directory that contains the init directories (rc0.d/ to rc6.d/)?
[/etc]
```

安装成功提示如下：

```
topsemic@topsemic-virtual-machine:/tmp/vmware-tools-distrib
Creating a new initrd boot image for the kernel.
update-initramfs: Generating /boot/initrd.img-4.10.0-28-generic
Generating the key and certificate files.
Successfully generated the key and certificate files.
The configuration of VMware Tools 10.1.6 build-5214329 for Linux for this running kernel completed successfully.

You must restart your X session before any mouse or graphics changes take effect.

You can now run VMware Tools by invoking "/usr/bin/vmware-toolbox-cmd" from the command line.

To enable advanced X features (e.g., guest resolution fit, drag and drop, and file and text copy/paste), you will need to do one (or more) of the following:
1. Manually start /usr/bin/vmware-user
2. Log out and log back into your desktop session
3. Restart your X session.

Enjoy,

--the VMware team

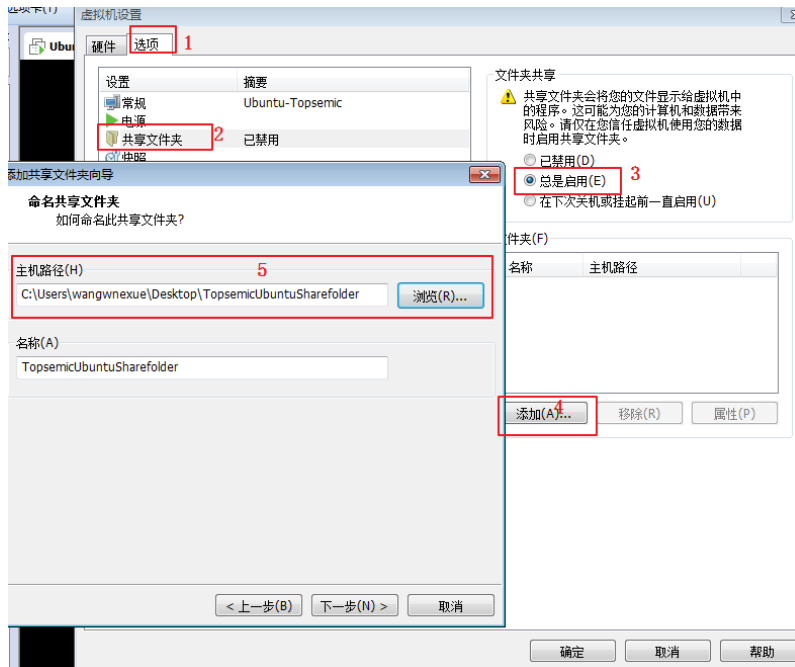
topsemic@topsemic-virtual-machine:/tmp/vmware-tools-distrib$
```

重新启动 ubuntu 系统（执行 reboot）就可以实现主机和虚拟机之间互相复制粘贴了。大家可以尝试下。



下面再介绍一个有用的东西，共享文件夹，就是在 Windows 下建立一个文件夹，虚拟机里也能同时访问它，比如这里在桌面建立 TopsemicUbuntuSharefolder 文件夹，里面随便放入一个文件。

点击虚拟机-设置，把共享文件夹打开，配置如下：



虚拟机里对应的目录是 `/mnt/hgfs/TopsemicUbuntuSharefolder` 文件夹，里面可以看到也是同样的内容。这样就免去了在主机和虚拟机之间来回复制贴粘了。

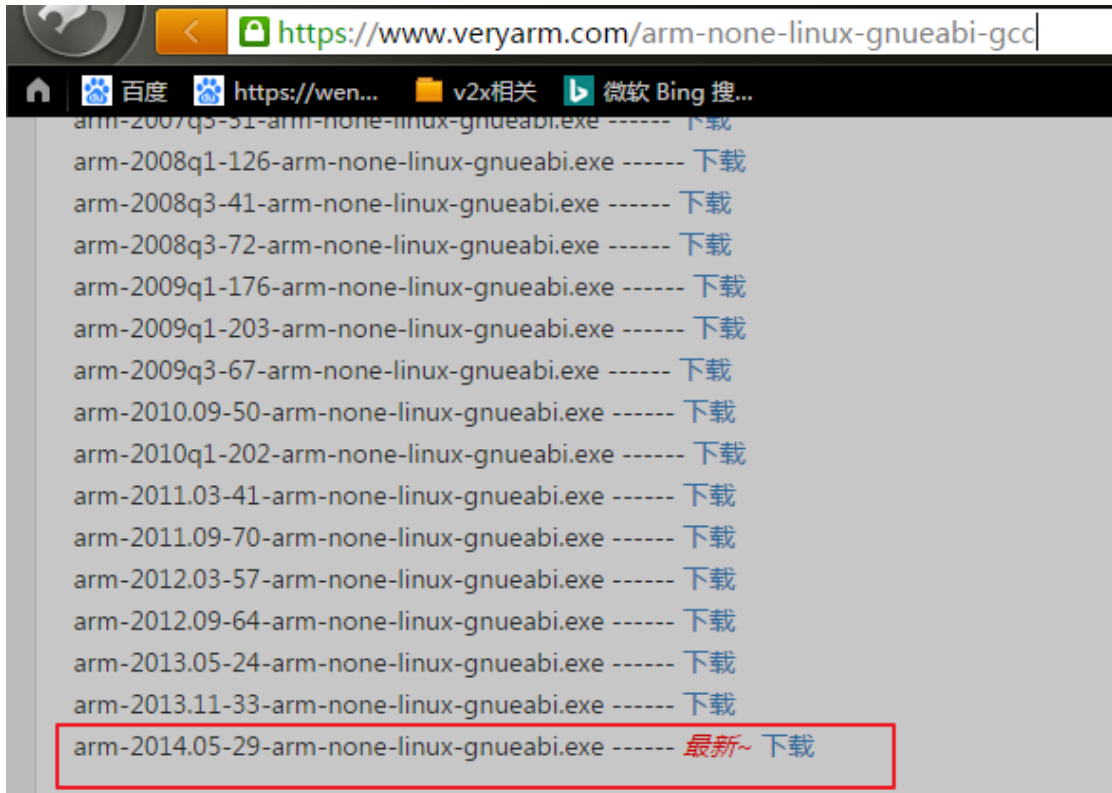
```
topsemic@topsemic-virtual-machine:/mnt/hgfs/TopsemicUbuntuSharefolder$ cd /mnt/hgfs/TopsemicUbuntuSharefolder/
topsemic@topsemic-virtual-machine:/mnt/hgfs/TopsemicUbuntuSharefolder$ ls
Linux学习系列二：运行Hello World .docx
topsemic@topsemic-virtual-machine:/mnt/hgfs/TopsemicUbuntuSharefolder$
```

另外我还见过有人用 Samba 实现虚拟机和主机共享文件夹的，我没有用过，也不知道相比 Vmware 自带的优势，所有就不在这里介绍了。

5. 交叉编译

如何让 hello world 程序在我的 NUC972 板子上运行呢？

第一步:交叉编译，也就说在 PC 机的 Ubuntu 系统里编译一个能在 NUC972 平台上运行的程序。这里就不能使用 Ubuntu 默认的 gcc 编译工具了。需要使用交叉编译工具链，我这里使用的是 `arm-2014.05-29-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2`，在 <https://www.veryarm.com/arm-none-linux-gnueabi-gcc> 网站下载的。



关于交叉编译和交叉编译链，这里多说两句。

解释什么是交叉编译之前，先要介绍下本地编译。在当前的 PC 下，X86 的 CPU 下，直接编译出来程序，可以直接在当前的环境，即 X86 的 CPU 下运行。此时的编译，可以叫做本地编译。交叉编译，是一个和本地编译相对应的概念。所谓的交叉编译，就是在一种平台上编译，编译出来的程序，是放到别的平台上运行，即编译的环境，和运行的环境，不一样，属于交叉的，此所谓交叉编译，这个概念主要和嵌入式开发有关。我们原来做单片机开发，在 Windows 使用 KEIL 或者 IAR 编译，其实也是交叉编译，因为最终它是运行在单片机里的。

要在 X86 平台编译出非 X86 平台运行的程序，就得需要交叉编译链，给定一个 CPU/SOC，如何确定其所用的交叉工具链呢？

主要逻辑是先确定该 CPU/SOC 所用的 CPU 内核，然后根据对应的 CPU 内核，去找其对应的，可用的交叉工具链。比如我们这里用的：NUC972

我们可以查到：

NUC972DF62Y - Nuvoton

www.nuvoton.com/hq/products/...control-series/nuc972df62y/?__locale=en

The NUC972DF62Y with the **ARM926EJ-S core** can operate at up to 300 MHz. It offers 216-pin LQFP package stacked with 64 MB DDR-II Memory to reduce PCB size and EMI issues.

它是基于 ARM926 的内核的。

所以我们要去找支持 ARM926 内核的交叉工具链。

进一步，通过 ARM926 处理器，我们了解到 ARM926 的内核架构是 ARMv5TEJ，所以我们去找，支持 ARMv5 的交叉编译器或者是支持 ARMv5TEJ 的交叉编译器。

常见的交叉编译工具链介绍，大家可参考：

<https://www.veryarm.com/cross-tools>，有一点大家一定要明确的是，一定要确定你用的交叉编译链得支持你的 CPU 架构。我当初编译 NUC972 就犯了大错，当时不了解这个，稀里糊涂的用了个错的交叉编译工具，卡了好多天才搞定。

下面介绍实际操作了：

1) 把下载下来的

arm-2014.05-29-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2 放到我们上面说的共享目录里，然后我们在/home/topsemic 目录新建一个 nuc972 文件夹，在 nuc972 文件夹里再新建一个 toolchain 文件夹，使用 mv 指令把 bz2 文件移过来，之后解压 bz2 文件。最后结果如下：



2) 修改环境变量，执行下面这句话

```
export PATH=$PATH:/home/topsemic/nuc972/toolchain/arm-2014.05/bin
```

为了方便起见，我把它放到/nuc972/toolchain/environment.sh 文件里了，以后要交叉编译，只需要 `source /home/topsemic/nuc972/toolchain/environment.sh` 即可。

source 指令的功能是使 Shell 读入指定的 Shell 程序文件并依次执行文件中的所有语句。

在终端上输入命令 arm- 再按 Tab 键，可以看到下图，说明环境变量设置成功了

```
topsemic@topsemic-virtual-machine:~/nuc972/toolchain$ export PATH=$PATH:/home/topsemic/nuc972/toolchain/arm-2014.05/bin
topsemic@topsemic-virtual-machine:~/nuc972/toolchain$ arm
arm2hpdL
arm-none-linux-gnueabi-addr2line
arm-none-linux-gnueabi-ar
arm-none-linux-gnueabi-as
arm-none-linux-gnueabi-c++
arm-none-linux-gnueabi-c++filt
arm-none-linux-gnueabi-cpp
arm-none-linux-gnueabi-cs
arm-none-linux-gnueabi-cs-daemon
arm-none-linux-gnueabi-elfedit
arm-none-linux-gnueabi-g++
arm-none-linux-gnueabi-gcc
arm-none-linux-gnueabi-gcc-4.8.3
arm-none-linux-gnueabi-gcc-ar
arm-none-linux-gnueabi-gcc-nm
arm-none-linux-gnueabi-gcc-ranlib
arm-none-linux-gnueabi-gcov
arm-none-linux-gnueabi-gdb
arm-none-linux-gnueabi-gprof
arm-none-linux-gnueabi-ld
arm-none-linux-gnueabi-nm
arm-none-linux-gnueabi-objcopy
arm-none-linux-gnueabi-objdump
arm-none-linux-gnueabi-ranlib
arm-none-linux-gnueabi-readelf
arm-none-linux-gnueabi-size
arm-none-linux-gnueabi-strings
arm-none-linux-gnueabi-strip
```

3) 编译

使用命令：`arm-none-linux-gnueabi-gcc hello.c -o helloworld` 编译

竟然报错了：提示 `arm-none-linux-gnueabi-gcc: 没有`

`/home/topsemic/nuc972../bin` 这个文件或目录，这是怎么回事，之前没遇到过啊。

上网一查说要需要安装 `lsb-core`，执行 `sudo apt-get install lsb-core`，安装完再编译就可以了。之前用的别人装好的虚拟机，应该都装过了，所以没有碰到这个问题。

至此，交叉编译工具就配置完成也能使用了。为了验证我们生成的 `helloworld` 确实是交叉编译的结果，我们可以试着直接虚拟机里执行下，它提示不可执行，

```
topsemic@topsemic-virtual-machine:~/hello$ ./helloworld
bash: ./helloworld: cannot execute binary file: 可执行文件格式错误
topsemic@topsemic-virtual-machine:~/hello$
```

我们在用 `file` 指令查看该文件的类型，可以看到它是 ARM 平台下的。

```
topsemic@topsemic-virtual-machine:~/hello$ file helloworld
helloworld: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), dynamically linked, interpreter /lib/ld-, for GNU/Linux 2.6.16, not stripped
topsemic@topsemic-virtual-machine:~/hello$
```

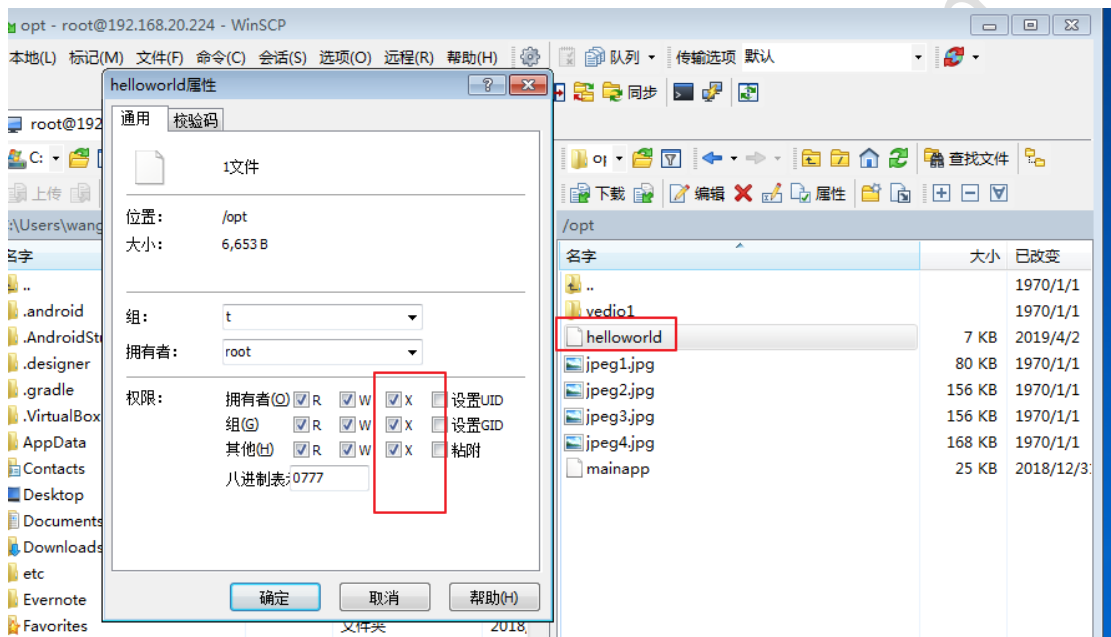
同时用 `file` 看下第 3 节编译出来的 `hello`，可以看到它是运行在 x86-64 平台下的。

```
topsemic@topsemic-virtual-machine:~/hello$ file hello
hello: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/l, for GNU/Linux 2.6.32, BuildID[sha1]=e6de77884ba59c776a95633f1d96410f97f017f2, not stripped
topsemic@topsemic-virtual-machine:~/hello$
```

4) 下载

我们要把编译生成的可执行文件最终放到 NUC972 板子里，怎么实现呢？我这里使用的方法是这样：

1) 先把 helloworld 复制到 Windows 系统，然后将板子通过网口和 PC 相连接，因为我的板子已经支持了 ssh 和 scp，同时板子的网口 IP 固定为 192.168.20.224。所以我只需要打开 Winscp 软件连接，直接将可执行文件放进去即可。Winscp 软件使用不在这里详细介绍了，大家自行百度即可。



2) 注意要修改可执行文件的权限，使它具有可执行权限，不然 SecureCRT 登录执行后，就会提示如下，Permission denied。SecureCRT 是一款支持 SSH (SSH1 和 SSH2) 的终端仿真程序，简单地说是 Windows 下登录 UNIX 或 Linux 服务器主机的软件。这个我感觉类似 Windows 下的远程连接工具。有好多类似的工具，比如 PUTTY、Xshell，大家随使用一种就行。具体使用方法也不在这里介绍了，不会的百度下即可。

```
192.168.20.224
[rootroot@wuya ~]#
[rootroot@wuya ~]#
[rootroot@wuya ~]#
[rootroot@wuya ~]#
[rootroot@wuya ~]#
[rootroot@wuya ~]# cd /opt/
[rootroot@wuya /opt]# ls
helloworld jpeg1.jpg jpeg2.jpg jpeg3.jpg jpeg4.jpg mainapp vedio1
[rootroot@wuya /opt]# ./helloworld
-sh: ./helloworld: Permission denied
[rootroot@wuya /opt]#
```

最后成功的结果如下：

```
[-] ./helloworld: permission denied  
[rootroot@wuya /opt]# ./helloworld  
Hello world  
[rootroot@wuya /opt]# █
```

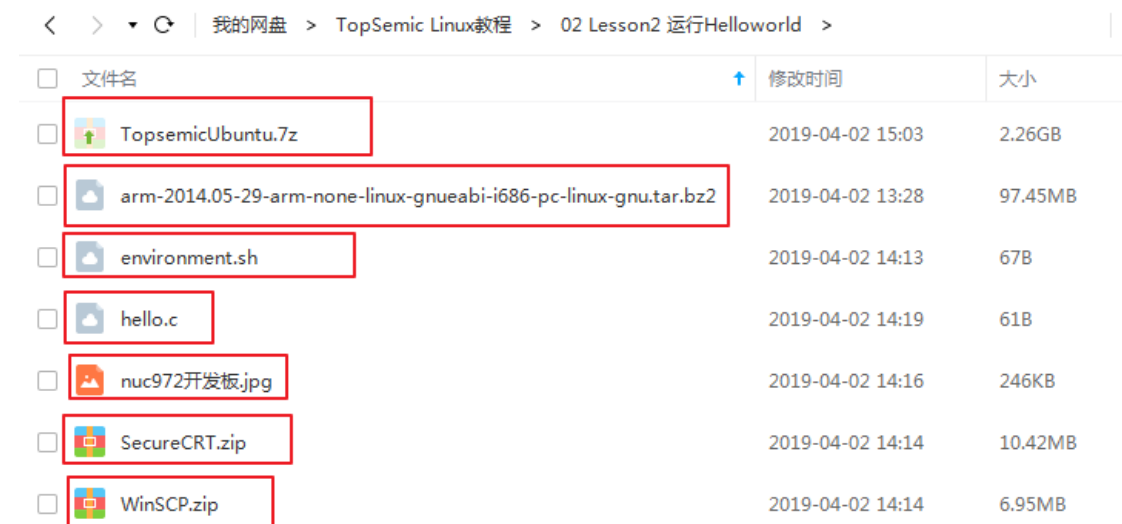
将 helloworld 放入到 NUC972 板子除了上面介绍的方法，还可以通过 NFS(网络文件系统)实现，或者先把文件放到 U 盘里，然后把 U 盘插到 NUC972 板子上，挂载拷出文件也可以。我个人觉得上面的方法比较好用。当然如何让板子支持 ssh 和 scp，在后面的文章中我会仔细再说。

6. 结束语

经过这一篇文章的介绍，想必你已经基本掌握了如何编译一个程序，熟悉了 gcc，搭建了交叉编译环境，另外也对如何把编译好的程序导入到嵌入式平台有了大概了解，同时又学习了几个新的 linux 命令。

本期相关的资料还是在百度网盘，链接：

<https://pan.baidu.com/s/1pN5Jc8L2FbrD8YgX6yHWhQ> 提取码：lke3；(02 Lesson2 运行 Helloworld 目录) 里面，同时把经过这一期操作后的虚拟机也放进去了，开机密码 topsemic，如果大家不想麻烦去安装 vmware-tools 就可以直接用了。或者自己的操作哪里不对，也可以做个比较。



文件名	修改时间	大小
TopsemicUbuntu.7z	2019-04-02 15:03	2.26GB
arm-2014.05-29-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2	2019-04-02 13:28	97.45MB
environment.sh	2019-04-02 14:13	67B
hello.c	2019-04-02 14:19	61B
nuc972开发板.jpg	2019-04-02 14:16	246KB
SecureCRT.zip	2019-04-02 14:14	10.42MB
WinSCP.zip	2019-04-02 14:14	6.95MB

有任何问题欢迎大家留言交流讨论。

下期会给大家介绍 NUC972 平台的移植。