

嵌入式软件开发技术与工具实验报告四

黎炜桁

一、实验目的

- 学会下载安装Linux及相关工具到目标机或目标机仿真环境。
- 学会使用交叉编译器编译应用程序，并能够将可执行程序下载到目标机运行。
- 学会使用交叉调试工具gdb-gdbserver调试应用程序。

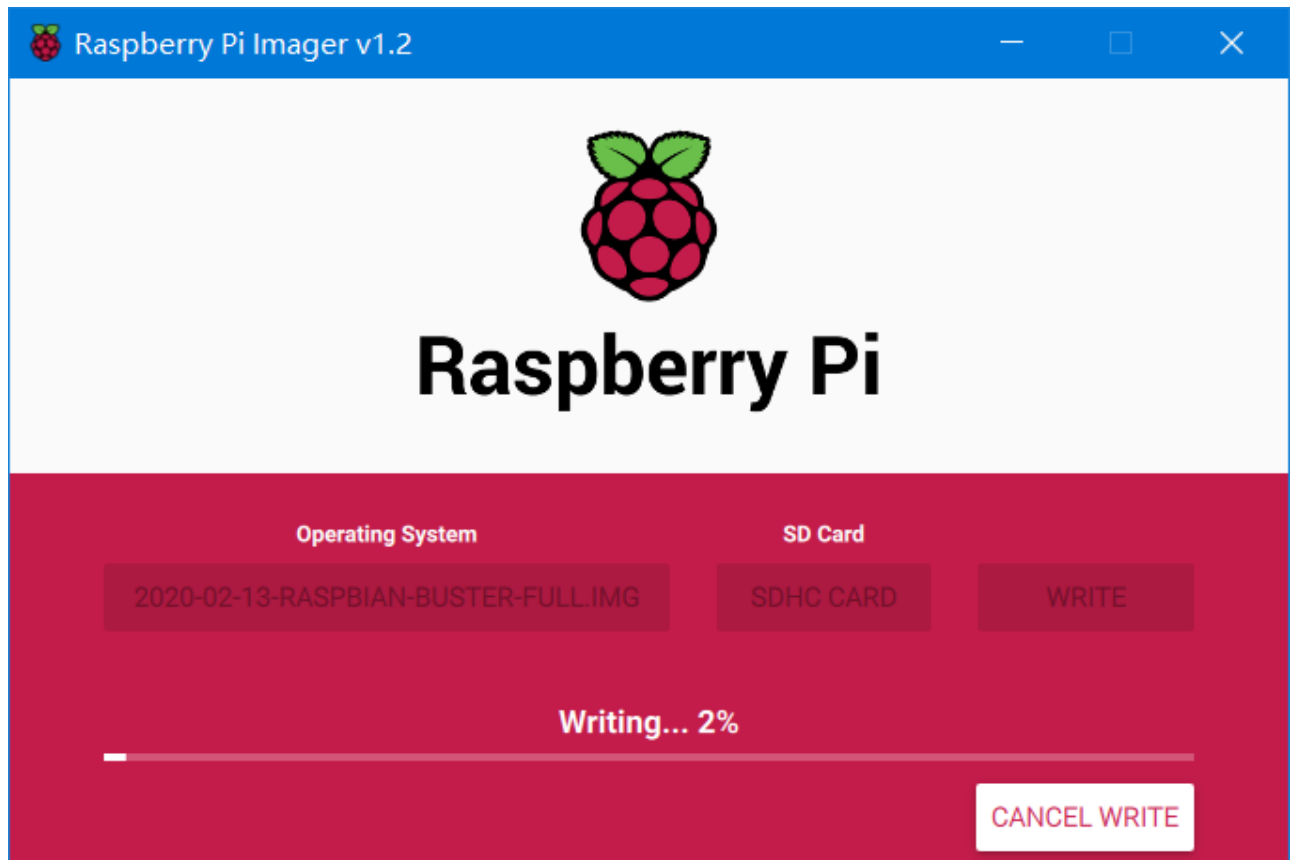
二、实验内容

1. Host-Target方案1：下载安装Linux系统到开发板，其中包括bootloader、内核、模块、库、文件系统等；建立host-target连接，常见连接有SSH，VNC，Putty。
2. Host-Target方案2：下载安装目标机仿真环境qemu，安装qemu树莓派系统。
3. 构建并测试开发板交叉编译器。
4. 建立gdb-gdbserver交叉调试环境并能够调试应用程序。

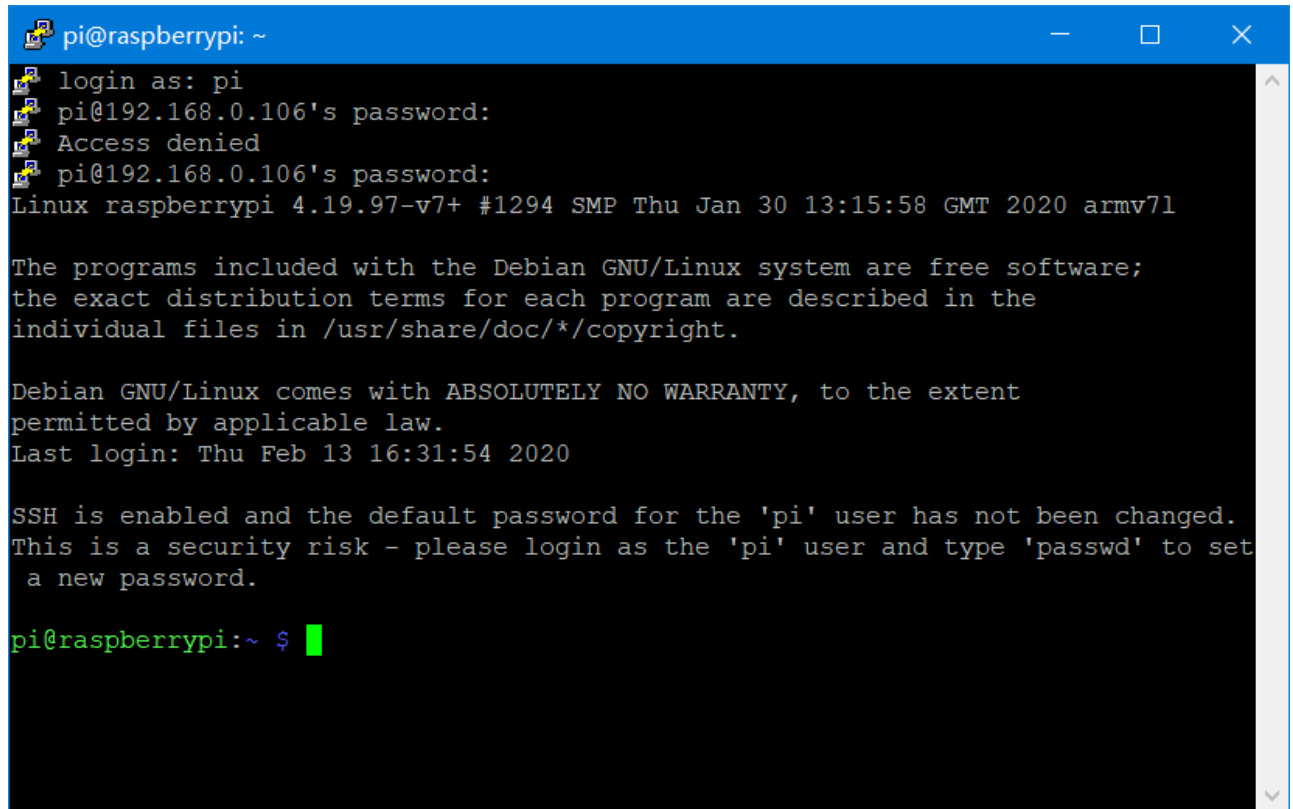
三、实验过程与结果

1. 下载安装Linux系统到开发板

- 从树莓派官网下载Linux镜像，写入SD卡中



- 在SD卡根目录创建ssh文件，使能ssh，使用Putty连接树莓派目标端



```
pi@raspberrypi: ~
login as: pi
pi@192.168.0.106's password:
Access denied
pi@192.168.0.106's password:
Linux raspberrypi 4.19.97-v7+ #1294 SMP Thu Jan 30 13:15:58 GMT 2020 armv7l

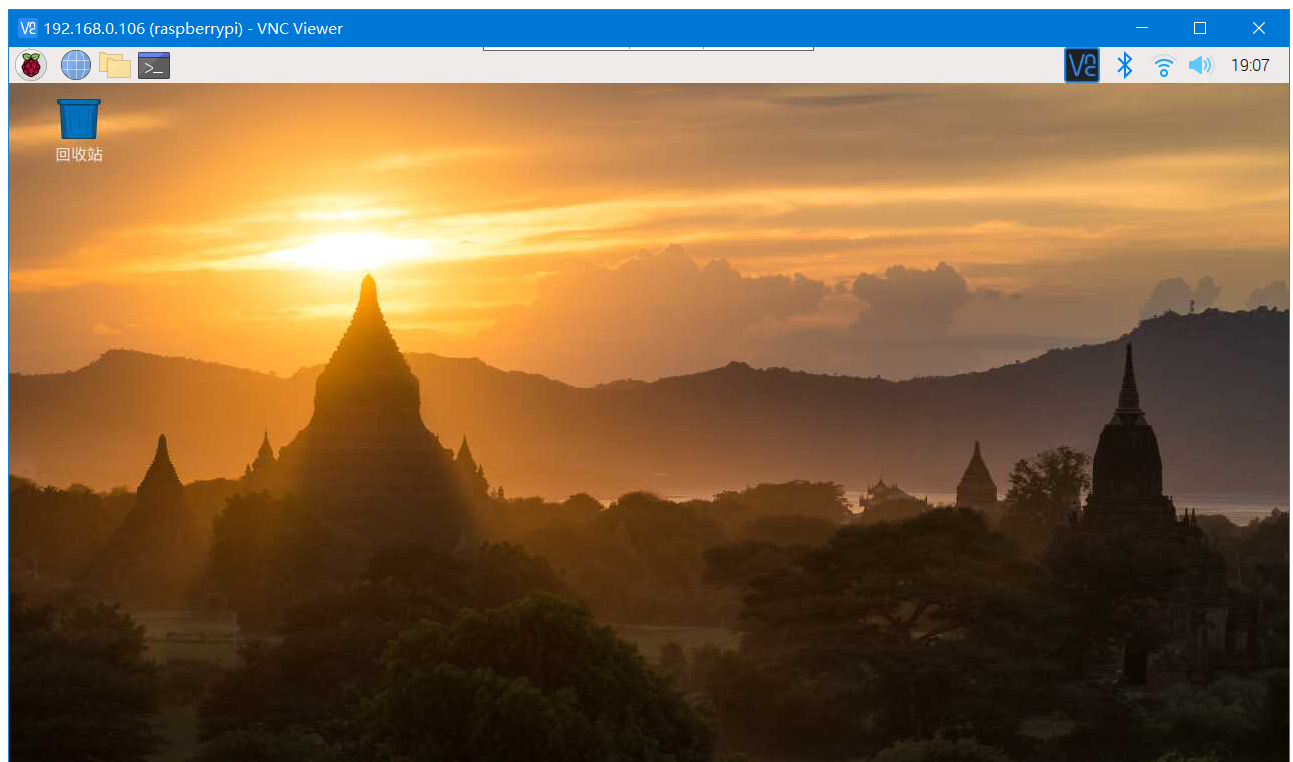
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Feb 13 16:31:54 2020

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~ $
```

- `sudo raspi -config` 打开VNC功能，使用VNC viewer远程登陆图形化界面，完成树莓派各项配置



2. 构建并测试开发板交叉编译环境

- 下载交叉编译工具tools.git
- 将编译器路径加入到系统环境变量,并使用`source ~/.bashrc`应用设置。

```

. .bashrc (~) - VIM
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

# export PATH=/home/liweiheng/rasppi/tools/arm-bcm2708/gcc-linaro-arm-linux-gnueabi-hf-raspbian-x64/bin
export PATH=$PATH:/usr/tools/arm-bcm2708/gcc-linaro-arm-linux-gnueabi-hf-raspbian-x64/bin
# export PATH=$PATH:$HOME/rasppi/tools/arm-bcm2708-linux-gnueabi/bin

```

- 输入 `arm-linux-gnueabi-hf-gcc -v` 查看交叉编译gcc版本，输出正确版本信息，表明交叉编译环境配置成功。

```

root@liweiheng-ThinkPad-X1-Carbon-5th: /usr/tools/arm-bcm2708
arm-linux-gnueabi-hf-gcov-5
root@liweiheng-ThinkPad-X1-Carbon-5th: /usr/tools/arm-bcm2708# arm-linux-gnueabi-hf-gcc -v
Using built-in specs.
COLLECT_GCC=arm-linux-gnueabi-hf-gcc
COLLECT_LTO_WRAPPER=/usr/lib/gcc-cross/arm-linux-gnueabi-hf/5/lto-wrapper
Target: arm-linux-gnueabi-hf
Configured with: ../src/configure -v --with-pkgversion='Ubuntu/Linaro 5.4.0-6ubuntu1~16.04.9' --with-bugurl=file:///usr/share/doc/gcc-5/README.Bugs --enable-languages=c,ada,c++,java,go,d,fortran,objc,obj-c++ --prefix=/usr --program-suffix=-5 --enable-shared --enable-linker-build-id --libexecdir=/usr/lib --without-include-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --with-sysroot=/ --enable-clocale=gnu --enable-libstdc++-debug --enable-libstdc++-time=yes --with-default-libstdc++-abi=new --enable-gnu-unique-object --disable-libitm --disable-libquadmath --enable-plugin --with-system-zlib --disable-browser-plugin --enable-java-awt=gtk --enable-gtk-cairo --with-java-home=/usr/lib/jvm/java-1.5.0-gcj-5-armhf-cross/jre --enable-java-home --with-jvm-root-dir=/usr/lib/jvm/java-1.5.0-gcj-5-armhf-cross --with-jvm-jar-dir=/usr/lib/jvm-exports/java-1.5.0-gcj-5-armhf-cross --with-arch-directory=arm --with-ecj-jar=/usr/share/java/eclipse-ecj.jar --disable-libgcj --enable-objc-gc --enable-multiarch --enable-multilib --disable-sjlj-exceptions --with-arch=armv7-a --with-fpu=vfpv3-d16 --with-float=hard --with-mode=thumb --disable-werror --enable-multilib --enable-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=arm-linux-gnueabi-hf --program-prefix=arm-linux-gnueabi-hf- --includedir=/usr/arm-linux-gnueabi-hf/include
Thread model: posix
gcc version 5.4.0 20160609 (Ubuntu/Linaro 5.4.0-6ubuntu1~16.04.9)
root@liweiheng-ThinkPad-X1-Carbon-5th: /usr/tools/arm-bcm2708#

```

- 建立hello.c文件，验证交叉编译正确性

```

#include<stdio.h>

int main(){

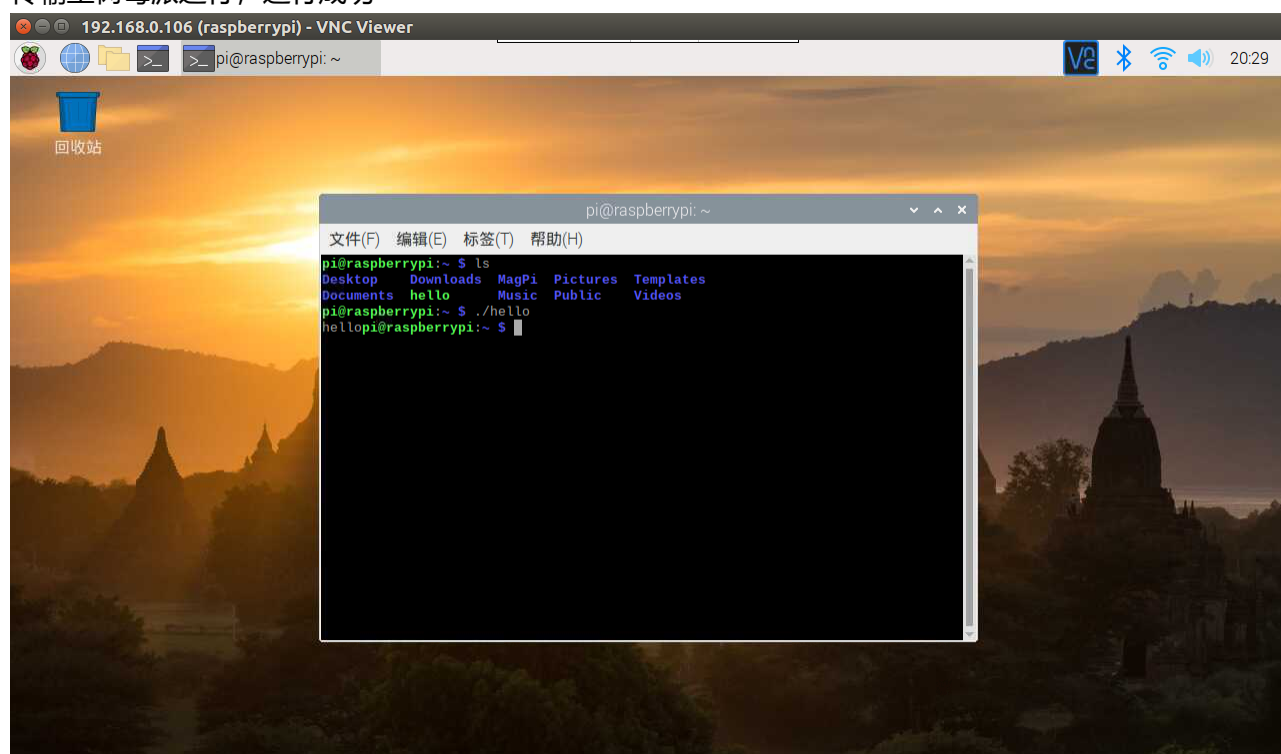
```

```
printf("hello");  
return 0;  
}
```

- 使用交叉编译工具进行编译，获得输出文件hello

```
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ arm-linux-gnueabihf-gcc hel  
lo.c -o hello  
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ ls  
hello  hello.c  
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$
```

- 传输至树莓派运行，运行成功

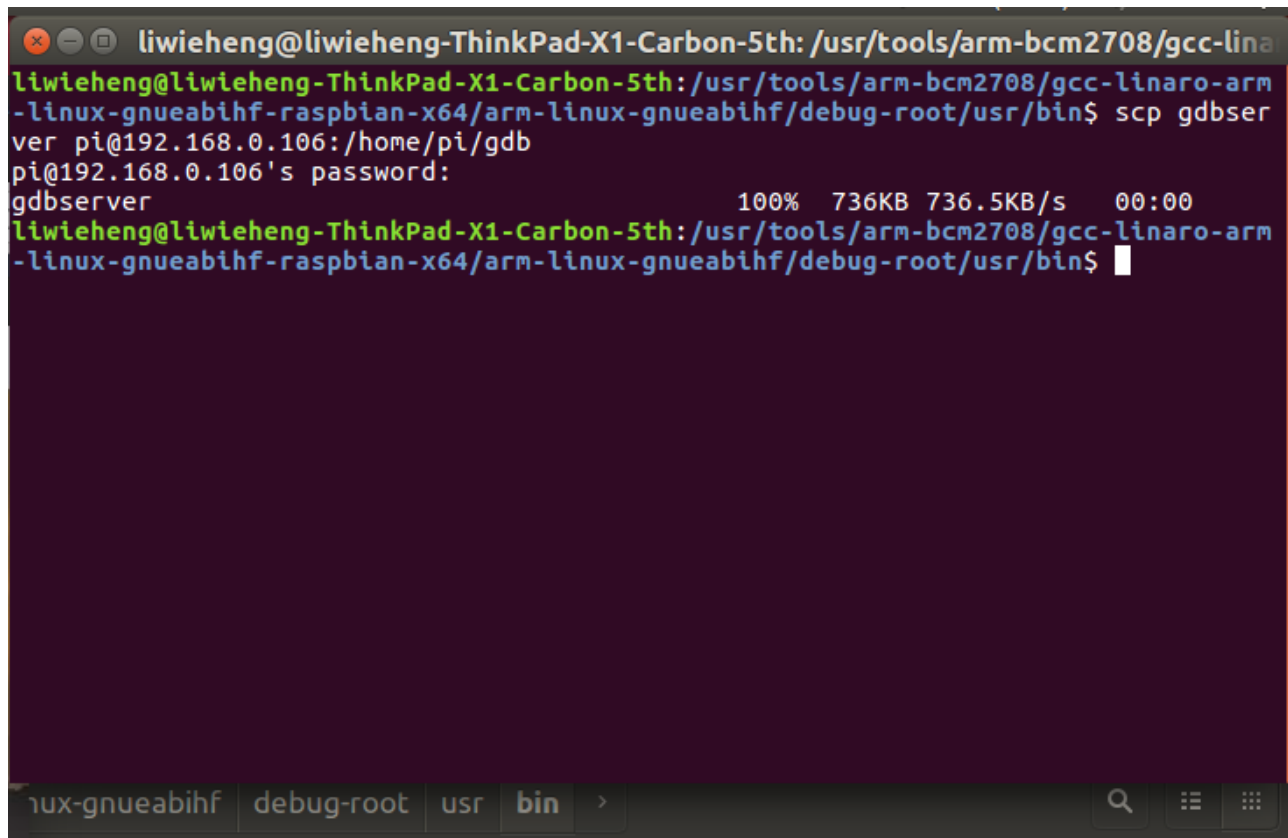


- 总结

交叉编译产生的执行文件不可在主机端运行，下载到目标端后运行成功，说明交叉编译工具是针对目标端环境进行编译的。

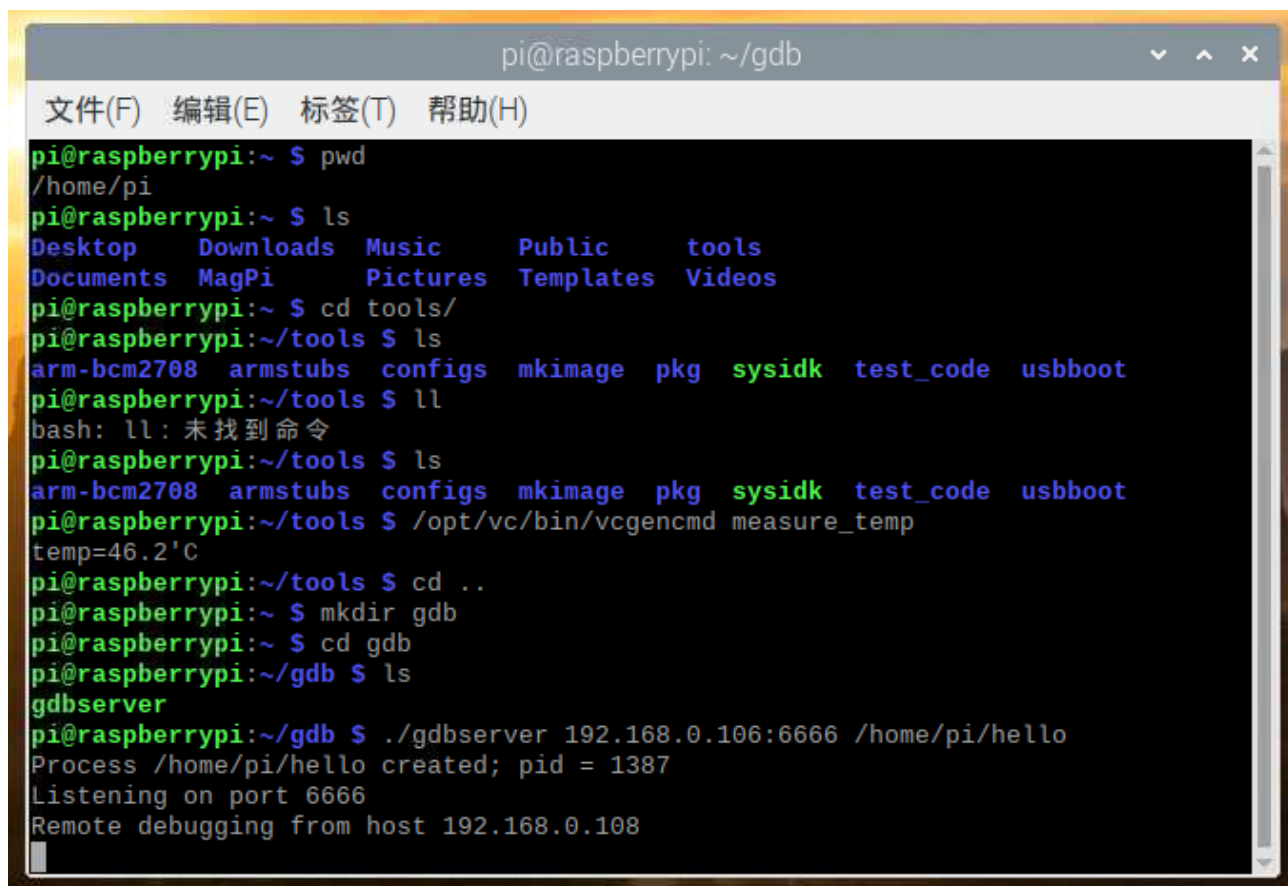
3. 建立gdb-gdbserver交叉调试环境

- 下载的交叉编译工具链中已包含了gdb-gdbserver相关工具，位于tools/arm-bcm2708/gcc-linaro-arm-linux-gnueabihf-raspbian-x64/arm-linux-gnueabihf/debug-root/usr/bin 目录下，可直接下载到树莓派目标端上即可使用



```
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th: /usr/tools/arm-bcm2708/gcc-linaro-arm-linux-gnueabihf-raspbian-x64/arm-linux-gnueabihf/debug-root/usr/bin$ scp gdbserver pi@192.168.0.106:/home/pi/gdb
pi@192.168.0.106's password:
gdbserver                               100% 736KB 736.5KB/s   00:00
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th: /usr/tools/arm-bcm2708/gcc-linaro-arm-linux-gnueabihf-raspbian-x64/arm-linux-gnueabihf/debug-root/usr/bin$
```

- 树莓派上启动gdbserver，打开6666端口作为调试使用



```
pi@raspberrypi: ~/gdb
文件(F) 编辑(E) 标签(T) 帮助(H)
pi@raspberrypi:~$ pwd
/home/pi
pi@raspberrypi:~$ ls
Desktop  Downloads  Music      Public     tools
Documents MagPi      Pictures   Templates  Videos
pi@raspberrypi:~$ cd tools/
pi@raspberrypi:~/tools$ ls
arm-bcm2708  armstubs  configs  mkimage  pkg  sysidk  test_code  usbboot
pi@raspberrypi:~/tools$ ll
bash: ll: 未找到命令
pi@raspberrypi:~/tools$ ls
arm-bcm2708  armstubs  configs  mkimage  pkg  sysidk  test_code  usbboot
pi@raspberrypi:~/tools$ /opt/vc/bin/vcgencmd measure_temp
temp=46.2'C
pi@raspberrypi:~/tools$ cd ..
pi@raspberrypi:~$ mkdir gdb
pi@raspberrypi:~$ cd gdb
pi@raspberrypi:~/gdb$ ls
gdbserver
pi@raspberrypi:~/gdb$ ./gdbserver 192.168.0.106:6666 /home/pi/hello
Process /home/pi/hello created; pid = 1387
Listening on port 6666
Remote debugging from host 192.168.0.108
```

- 主机端Ubuntu使用-g关键字重新编译hello.c，支持gdb调试


```
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th: ~/rasppi
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~$ cd rasppi/
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ gedit hello.c

(gedit:8897): Gtk-WARNING **: Attempting to read the recently used resources file at '/home/liwieheng/.local/share/recently-used.xbel', but the parser failed: 打开文件“/home/liwieheng/.local/share/recently-used.xbel”失败: 权限不够.
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ arm-linux-gnueabihf-gcc -g hello.c -o hello
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ scp hello pi@192.168.0.106:/home/pi
pi@192.168.0.106's password:
hello                                100% 9320      9.1KB/s   00:00
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$
```

- 主机端Ubuntu启动gdb，调试hello。使用target remote命令连接至目标端，b 4在第四行加入断点，c执行

```

liwieheng@liwieheng-ThinkPad-X1-Carbon-5th: ~/rasppi
Thread model: posix
gcc version 5.4.0 20160609 (Ubuntu/Linaro 5.4.0-6ubuntu1~16.04.9)
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ rm hello
rm: 是否删除有写保护的普通文件 'hello'?
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ arm-linux-gnueabi-gcc hel
lo.c -o hello
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ ls
hello hello.c
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ scp hello pi@192.168.0.106:
/home/pi
pi@192.168.0.106's password:
hello                                100% 8248      8.1KB/s   00:00
liwieheng@liwieheng-ThinkPad-X1-Carbon-5th:~/rasppi$ arm-linux-gnueabi-gdb
GNU gdb (crosstool-NG linaro-1.13.1+bzr2650 - Linaro GCC 2014.03) 7.6.1-2013.10
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "--host=x86_64-build_unknown-linux-gnu --target=arm-l
inux-gnueabi".
For bug reporting instructions, please see:
<https://bugs.launchpad.net/gcc-linaro>.
(gdb) target remote 192.168.0.106:6666
Remote debugging using 192.168.0.106:6666
warning: Could not load vsyscall page because no executable was specified
try using the "file" command first.
0x76fcea30 in ?? ()
(gdb)

```

```

liwieheng@liwieheng-ThinkPad-X1-Carbon-5th: ~/rasppi
Continuing.

Breakpoint 1, main () at hello.c:6
6          printf("hello %d",i);
(gdb) p i
$1 = 1
(gdb) c
Continuing.

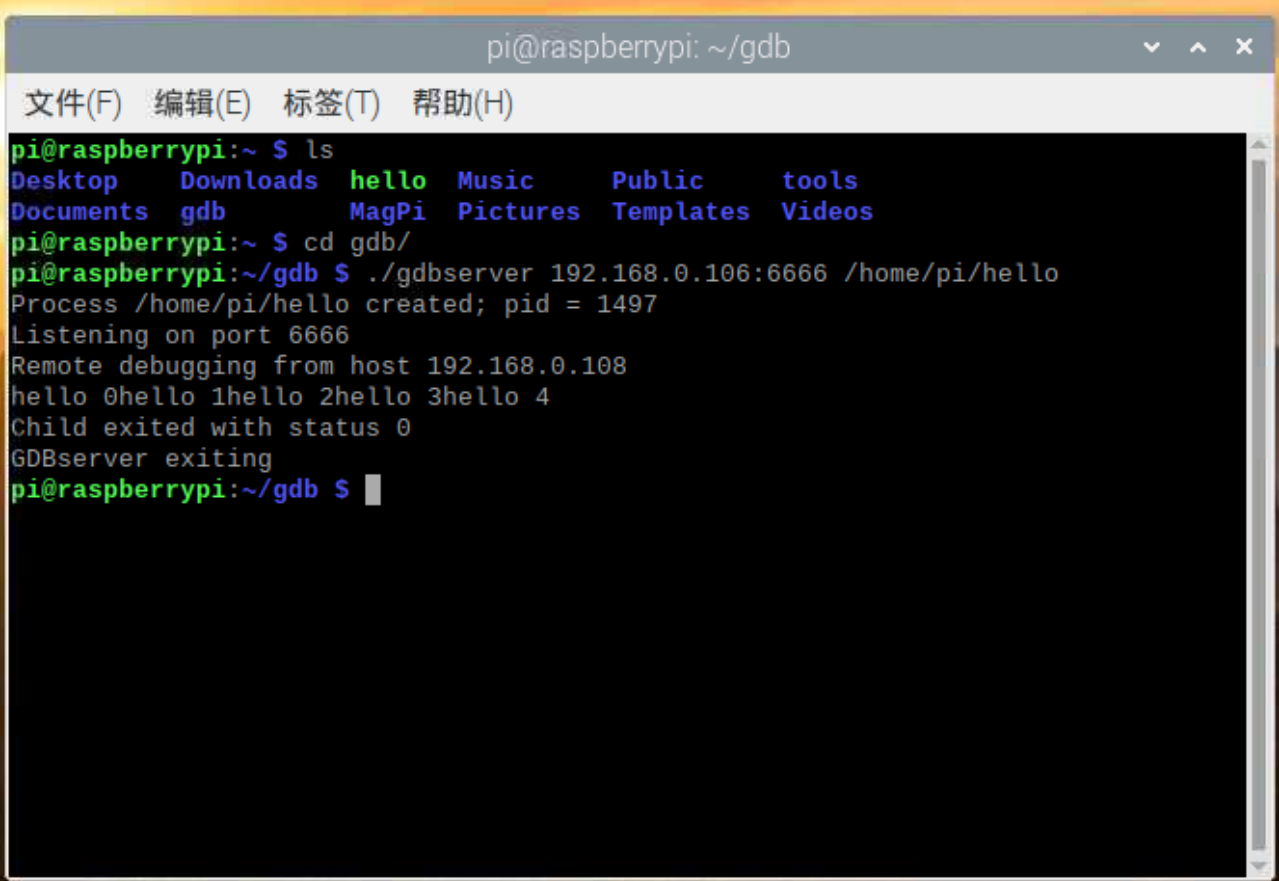
Breakpoint 1, main () at hello.c:6
6          printf("hello %d",i);
(gdb) c
Continuing.

Breakpoint 1, main () at hello.c:6
6          printf("hello %d",i);
(gdb) p i
$2 = 3
(gdb) c
Continuing.

cBreakpoint 1, main () at hello.c:6
6          printf("hello %d",i);
(gdb) c
Continuing.
[Inferior 1 (process 1497) exited normally]
(gdb) c
The program is not being run.
(gdb) c

```

- 在目标端树莓派gdbserver观察到程序执行成功



```
pi@raspberrypi: ~/gdb
文件(F) 编辑(E) 标签(T) 帮助(H)
pi@raspberrypi:~ $ ls
Desktop  Downloads  hello  Music  Public  tools
Documents  gdb  MagPi  Pictures  Templates  Videos
pi@raspberrypi:~ $ cd gdb/
pi@raspberrypi:~/gdb $ ./gdbserver 192.168.0.106:6666 /home/pi/hello
Process /home/pi/hello created; pid = 1497
Listening on port 6666
Remote debugging from host 192.168.0.108
hello 0hello 1hello 2hello 3hello 4
Child exited with status 0
GDBserver exiting
pi@raspberrypi:~/gdb $
```

- 结论

使用gdb-gdbserver进行远程调试可以非常方便的在主机端调试目标端程序，并可加入断点，观察变量等，极大的方便了程序的开发。

需要注意的是在交叉编译时需要加入-g关键字使程序支持gdb调试，否则在gdb中进行调试将无法被识别。

四、实验总结

本次实验我掌握了在树莓派上安装了Linux系统，并实现了主机端与目标端的交叉编译、交叉调试环境，熟练了Linux下的编程。