

华中科技大学计算机科学与技术学院



TinyOS 实验

入门指导



石柯

2016/7/25



目录

1	开发环境.....	2
1.1	综述	2
1.2	硬件平台	2
1.3	软件	4
2	安装和配置指南.....	5
2.1	Linux 下的安装与配置.....	5
2.2	虚拟机环境下的安装与配置	11
2.3	IDE 的安装与使用	13
3	开发与调试.....	15
4	参考文献及网站.....	16



1 开发环境

1.1 综述

实验开发环境由安装 Linux 操作系统的计算机和 DEVISER TelosB 节点构成，两者之间通过 USB 连接。

1.2 硬件平台

DEVISER TelosB 节点，如图 1.1 所示，包括防水盒、锂电池和 TelosB 节点构成。



图 1.1 DEVISER TelosB 平台



TelosB 节点 (TPR2420) 最初由 Berkeley 大学研发、发布, 具有低功耗和快速苏醒功能, 可以保证更长的电池寿命, 与开源 TinyOS 兼容, 如图 2.1 所示。TelosB 节点把所有要素都集中在一个独立的平台上: USB 编程能力, IEEE802.15.4 射频器和天线, 低功耗扩大内存的 MCU 和传感器套件, 具有以下特性:

- IEEE802.15.4 RF 收发器
- 2.4—2.4835GHz 全球兼容 ISM 波段
- 250kbps 传输速率
- 集成板载天线
- TI MSP430 10kB RAM 微处理器
- 低电流消耗
- 1MB 用于数据存储的外部闪存
- 通过 USB 接口获取数据和编程
- 包含光, 温度和湿度的传感器套件
- 运行 TinyOS1.1.11 或更高版本。

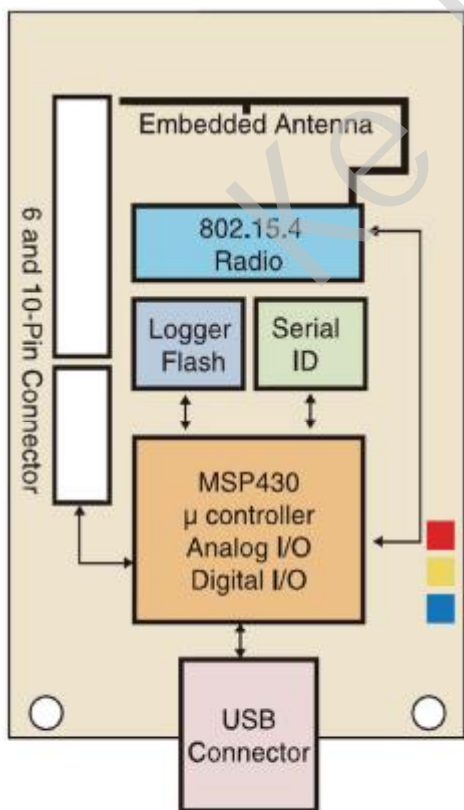


图 1.2 TelosB 结构



1.3 软件

软件由运行在 Linux 操作系统下的交叉编译工具链和 TinyOS 操作系统构成，具体用到的软件见安装与配置部分。

Ke HUST



2 安装和配置指南

2.1 Linux 下的安装与配置

(1) Linux 操作系统安装

下载和安装 Ubuntu 14.04.4 Desktop (64-bit) (ubuntu-14.04.4-desktop-amd64.iso), 下载地址为 <http://www.ubuntu.com/download/alternative-downloads>, 若要用双系统, 请先安装 Windows 再安装 Ubuntu 14.04.4, 安装教程网络上很多, 请自行 Google 或 baidu, 注意安装 linux 的主分区建议留 40G 空间以上, 交换分区(swap partition)留 8G(物理内存的 2 倍以上)。

(2) 安装开发工具

具体的开发工具包括:

nesc: 1.3.6

tinyos-tools: 1.4.2

tinyos-tools-14: 1.4.2

tinyos-tools-devel: 1.5.0

mispdebug: 0.22

avarice-tinyos: 2.11-20110905

avr-binutils-tinyos: 2.17-20110905

avr-gcc-tinyos: 4.1.2-20110905

avr-libc-tinyos: 1.6.7-20110905

avr-tinyos: 20110905

avrdude-tinyos: 5.10-20110905

avr-binutils-tinyos-legacy: 2.17-20110905

avr-gcc-tinyos-legacy: 4.1.2-20110905

avr-libc-tinyos-legacy: 1.4.7-20110905

avr-tinyos-legacy: 20110905



avrdude-tinyos-legacy: 5.4-20110904

uisp-tinyos: 20050519tinyos-20111126

misp430-binutils-tinyos-legacy: 2.17-20110905

misp430-gcc-tinyos-legacy: 3.2.3-20110905

misp430-libc-tinyos-legacy: 20060801cvs-20110905

misp430-tinyos-legacy: 20110905

arm-all-tinyos: 2011.03-42-20110912 (i386 only)

建议采用自动安装方式，基于 Debian 的.deb 包自动从 repository 安装，自动解析软件包之间的依赖关系，防止产生冲突，采用的 repository 位于 <http://www.tinyprod.net/>，需要计算机联网。

安装前，请先检查系统，删除所有已安装的相关包（第一次安装不需要）：

```
$ sudo dpkg -P `dpkg -I nesc '*tinyos*' | grep ^ii | awk '{ print $2 }' | xargs`
```

```
$ sudo apt-get clean
```

连接 tinyprod 服务器之前进行必要的认证：

```
$ sudo wget -O - http://tinyprod.net/repos/debian/tinyprod.key | sudo apt-key add -
```

或

```
$ gpg --keyserver keyserver.ubuntu.com --recv-keys A9B913B9
```

```
$ gpg -a --export A9B913B9 | sudo apt-key add -
```

一般前一种即可。

新建或修改/etc/apt/sources.list.d/tinyprod-debian.list 文件，让系统知道从哪里找到相关的软件包。

```
$ sudo -s
```

```
$ cd /etc/apt/sources.list.d
```

```
$ echo "deb http://tinyprod.net/repos/debian wheezy main" >> tinyprod-debian.list
```

```
$ echo "deb http://tinyprod.net/repos/debian misp430-46 main" >> tinyprod-debian.list
```



最终的/etc/apt/sources.list.d/tinyprod-debian.list 包含如下两行:

```
deb http://tinyprod.net/repos/debian wheezy main
```

```
deb http://tinyprod.net/repos/debian msp430-46 main
```

安装 nesc、tinyos-tools 和 msp430-46 (用于 telosb 的交叉编译工具):

```
$ sudo apt-get update
```

```
$ sudo apt-get install nesc tinyos-tools msp430-46
```

在安装过程中, ubuntu 会自动安装和配置 JDK (openjdk) 环境。

修改/etc/apt/sources.list.d/tinyprod-debian.list 文件:

将 deb http://tinyprod.net/repos/debian wheezy main 改为 deb

```
http://tinyprod.net/repos/debian squeeze main
```

安装 avr-tinyos (用于 micaz、iris 等的交叉编译工具):

```
$ sudo apt-get update
```

```
$ sudo apt-get install avr-tinyos
```

如果想要删除所安装的软件, 可以使用如下命令:

```
$ sudo apt-get autoremove --purge nesc tinyos-tools msp430-46 avr-tinyos
```

(3) 安装和配置 TinyOS-2.1.2

首先获取源代码, 在用户主目录即/home/[username](例如用户名为 tinyos, 则目录名为 /home/tinyos)进入终端:

```
wget http://github.com/tinyos/tinyos-release/archive/tinyos-2_1_2.tar.gz
```

```
tar xf tinyos-2_1_2.tar.gz
```

```
mv tinyos-release-tinyos-2_1_2 tinyos-main
```

第一个命令是从 github 网上代码仓库获取源代码文件, 第二个命令是解压缩源代码压缩包文件, 第三个命令是重命名解压后的目录。

然后设置环境变量:

```
export TOSROOT="$HOME/tinyos-main"
```

```
export TOSDIR="$TOSROOT/tos"
```

```
export CLASSPATH=$CLASSPATH:$TOSROOT/support/sdk/java/tinyos.jar:.
```




```
export MAKERULES="$TOSROOT/support/make/Makerules"
```

```
export PYTHONPATH=$PYTHONPATH:$TOSROOT/support/sdk/python
```

```
echo "setting up TinyOS on source path $TOSROOT"
```

上述命令会把环境变量写入当前用户目录下的.bashrc 文件，设置在当前目录下有效。
若想设置对全部用户有效，请将前 5 行的内容单独编辑为独立的 tinyos.env 文件，将此文件拷贝到/etc/profile.d/

(4) 检查开发环境

设置完成后，在终端键入

```
tos-check-env
```

检查环境是否安装和设置好，结果应该类似：

Path:

```
/usr/local/sbin  
/usr/local/bin  
/usr/sbin  
/usr/bin  
/sbin  
/bin  
/usr/games  
/usr/local/games
```

Classpath:

```
/home/tinyos/tinyos-main/support/sdk/java/tinyos.jar
```

.

rpms:

nesc:

```
/usr/bin/nesc  
Version: nesc: 1.3.6
```

perl:

```
/usr/bin/perl  
Version: v5.18.2) built for x86_64-linux-gnu-thread-multi
```

flex:



bison:

java:

/usr/bin/java

--> WARNING: The JAVA version found first by tos-check-env may not be version 1.4 or version 1.5 one of which is required by TOS. Please ensure that the located Java version is 1.4 or 1.5

graphviz:

/usr/bin/dot

dot - graphviz version 2.36.0 (20140111.2315)

--> WARNING: The graphviz (dot) version found by tos-check-env is not 1.10. Please update your graphviz version if you'd like to use the nescdoc documentation generator.

tos-check-env completed with errors:

--> WARNING: The JAVA version found first by tos-check-env may not be version 1.4 or version 1.5 one of which is required by TOS. Please ensure that the located Java version is 1.4 or 1.5

--> WARNING: The graphviz (dot) version found by tos-check-env is not 1.10. Please update your graphviz version if you'd like to use the nescdoc documentation generator.

关于 java 和 graphviz 的警告是由于 Java 版本较新造成的，可以不必理会。

```
printenv MAKERULES
```

显示

/home/tinyos/tinyos-main/support/make/Makerules

(5) 允许读写串口

烧写节点需要访问串口的权限，可键入下面的命令：

```
sudo gpasswd -a <your-user> dialout
```

用用户名替代<your-user>。

(6) 第一个 TinyOS 程序的编译和烧写



将节点电源关闭，用 USB 电缆连接计算机和节点，键入 **motelist** 将显示下面的信息：

```
tinyos@tinyos-VirtualBox:~$ motelist
```

Reference	Device	Description

XBT4P3EA	/dev/ttyUSB0	XBOW Crossbow Telos Rev.B

其中 `/dev/ttyUSB0` 为该节点在 linux 中对应的设备文件。

命令 **motelist** 仅支持 telosB 节点，其他节点请用 **dmesg** 命令，输出类似：

```
[ 461.770353] usb 2-2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 461.770354] usb 2-2: Product: Crossbow Telos Rev.B
[ 461.770356] usb 2-2: Manufacturer: XBOW
[ 461.770357] usb 2-2: SerialNumber: XBT4P3EA
[ 461.793897] usbcore: registered new interface driver usbserial
[ 461.794030] usbcore: registered new interface driver usbserial_generic
[ 461.794175] usbserial: USB Serial support registered for generic
[ 461.800271] usbcore: registered new interface driver ftdi_sio
[ 461.800541] usbserial: USB Serial support registered for FTDI USB Serial Device
[ 461.800601] ftdi_sio 2-2:1.0: FTDI USB Serial Device converter detected
[ 461.800646] usb 2-2: Detected FT232BM
[ 461.808918] usb 2-2: FTDI USB Serial Device converter now attached to ttyUSB0
```

然后可以编译和烧写程序，进入 LED 闪烁程序目录：

```
cd $TOSROOT/apps/Blink
```

```
make telosb install /dev/ttyUSB0
```

已有程序，请将 **install** 换成 **reinstall**

显示如图 2.1：

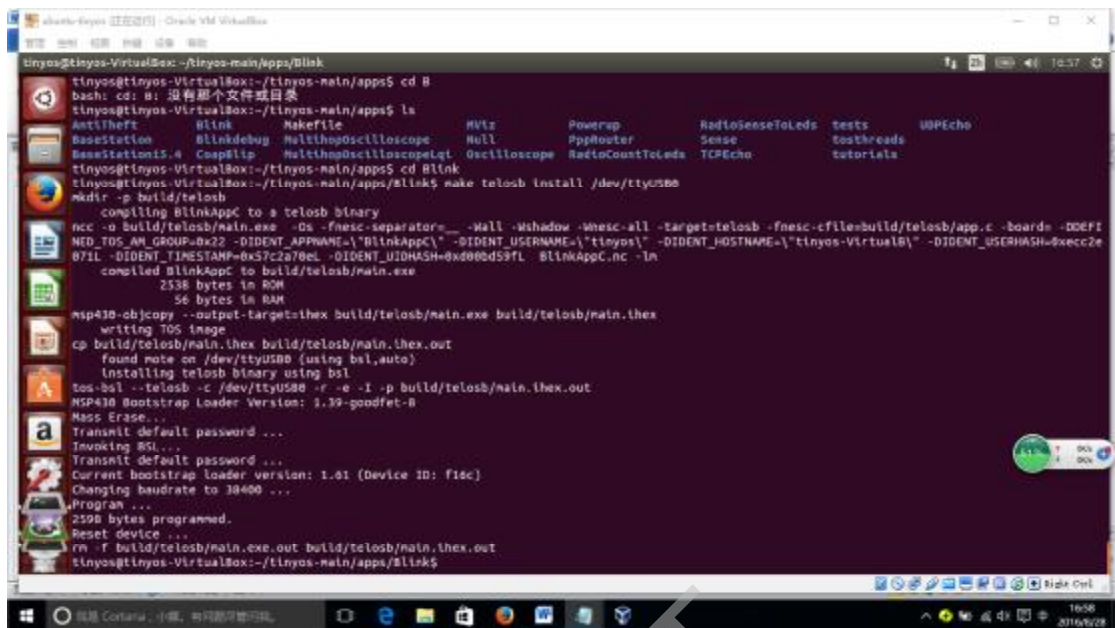


图 2.1 烧写程序至 telosb 节点

然后可以看到节点的三个 LED 灯闪烁。

(7) 使用 TOSSIM 进行仿真 (可选)

目前 tossim 仅支持 micaz 节点, 进入程序目录输入命令:

```
make micaz sim
```

会编译完成相应的仿真库文件, 使用 tossim 即可仿真, 请参考 tinyos 文档。若未显示

```
*** Successfully built micaz TOSSIM library.
```

一般是因为未安装所需的开发包, 请执行:

```
sudo apt-get install build-essential
```

```
sudo apt-get install python-dev
```

2.2 虚拟机环境下的安装与配置

到 <https://www.virtualbox.org/> 下载虚拟机软件:

VirtualBox-5.1.0-108711-Win.exe

Oracle_VM_VirtualBox_Extension_Pack-5.1.0-108711.vbox-extpack

按提示安装即可, 包括主程序和扩展组件。为保证能够在虚拟机中访问到节点, 请在 Windows 系统中安装 USB 转串口驱动, CDM v2.12.18 WHQL Certified.zip(www.ftdichip.com)。



可以按上一节的步骤新建虚拟机安装 **ubuntu** 系统进行配置, 也可以使用已有的虚拟机文件, 用户名和密码均为 **tinyos**, 此时注意计算机的硬件配置中要将虚拟化支持选项打开(缺省一般是打开的)。

首次系统启动后要运行扩展组件, 如图 2.2 所示。

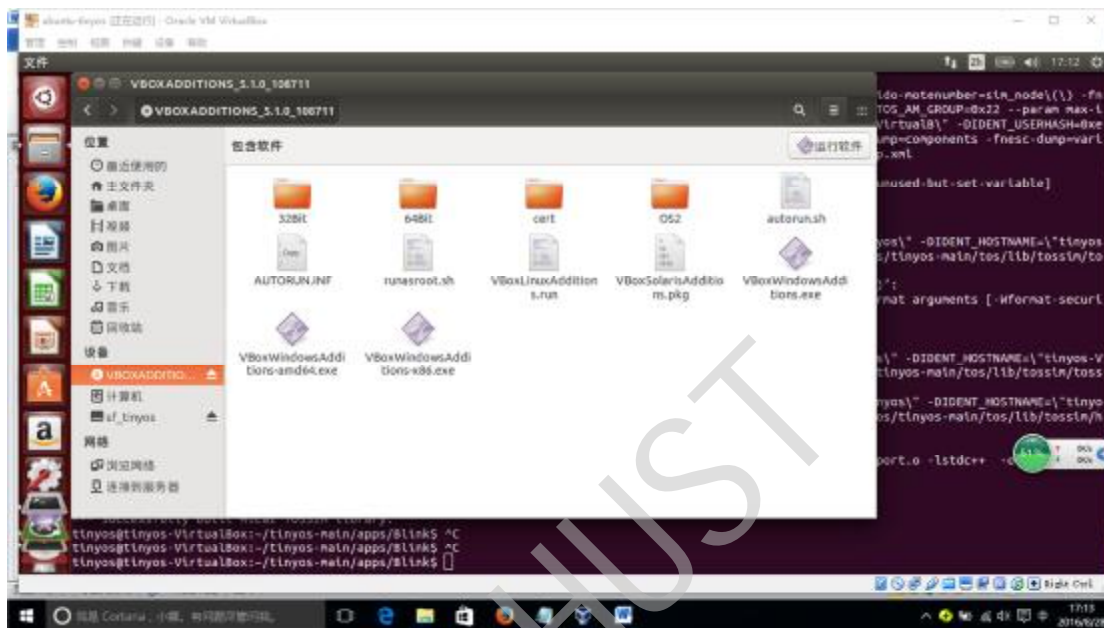


图 2.2 加载扩展组件

连接节点后, 左键点击右下角的 **USB** 设备标志, 选择节点, 如图 2.3 所示。

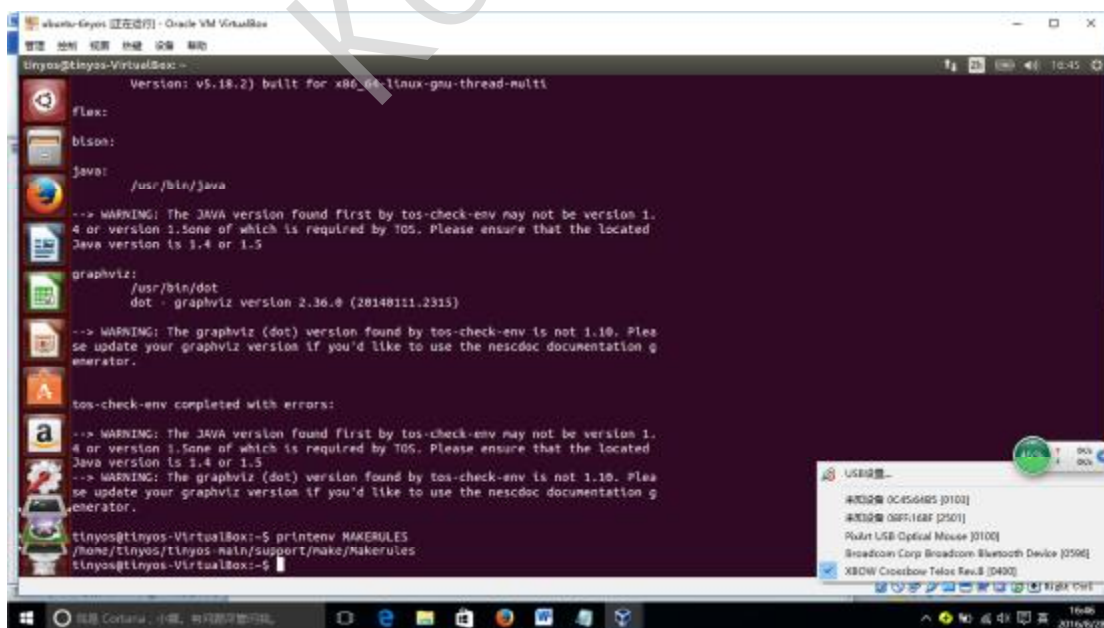


图 2.3 选择 USB 设备



2.3 IDE 的安装与使用

系统可以安装 eclipse 和相应的插件 Yet2 (<http://tos-ide.ethz.ch/wiki/index.php>), 但该系统已不再更新, 下面是具体的过程指导 (来自网站 <https://github.com/pillforge/tinyos/wiki/TinyOS-for-Eclipse-on-Ubuntu>)。

1. Follow the instructions for [installing TinyOS command line tools on Ubuntu](#)
2. Install Eclipse, the C Development Tools and the Graphical Editing Framework and all dependencies:

```
sudo apt-get update
```

```
sudo apt-get install eclipse eclipse-cdt eclipse-gef
```

3. Start Eclipse, select "Help > Software Updates...".
4. You need to add a new update site, so click "Add...".
5. Enter the update site <http://tos-ide.ethz.ch/update/site.xml> (name: "Yeti2 Update Site") into the dialog.
6. You are now able to select the plugins to install. You should select the following entries:
 - Yet2 - Core Plugin
 - Yet2 - TinyOS NesC 1.3.x Parser
 - Yet2 - TinyOS Base Environment
 - Yet2 - TinyOS 2.x on Unix
 - Yet2 - Formatter
 - Yet2 - Refactoring
 - Yet2 - TinyOS Debugger
 - Yet2 - TinyOS Search
7. Restart Eclipse and select the "TinyOS" perspective (Window menu).
8. Select "Window/Preferences" and setup the environment at the "TinyOS/Environments/TinyOS 2.x unix-environment" node. If you followed [this tutorial](#) the defaults should be OK, and no architecture specific paths should be set, otherwise set the directory paths properly.
9. Run "TinyOS/Check installation..." for verifying your setup
10. Start a new example project: "File/New/Example..." and select "TinyOS Example"
 - Name: MyBlink
 - Environment: TinyOS Unix Wrapper 2
 - Example: Blink
- Target: telosb
11. Select "Run/Run Configurations..." and select "TinyOS Build":
 - Name: MyBlinkConfig
 - Project: MyBlink
 - Extras: **select debug**
- Click "Apply"
12. Click "Run/Run". The project should compile with no errors.



13. Select "Run/External Tools/External Tools Configurations..." and select "Program"

Name: mspdebug

Location: /usr/bin/mspdebug (or your custom location)

Working Directory: select "Workspace..." and select "MyBlink/build/telosb"

Arguments: rf2500

"prog main.exe"

Click "Apply"

14. Click "Run/External Tools/mspdebug" for programming the mote

15. Select "Run/Debug Configurations..." and select "TinyOS Debugger"

Name: MyBlinkDebug

Project: MyBlink

TinyOS Binary: build/telosb/main.exe

GDB Proxy: User defined command

GDB Proxy Command: mspdebug rf2500 gdb

GDB Debugger: msp430-gdb

Select "Verbose console mode"

Port: 2000

Click "Apply"

16. Select "Project/Properties" and select "TinyOS Source". Click on "Link Source..."

Linked folder location: /opt/tinyos

Folder name: tinyos

17. Source code level debugging should work now by using "Run/Debug".

注意：调试功能需要相应的硬件支持，我们的节点没有这样的硬件。



3 开发与调试

Tinyos 的调试比较困难，使用 GDB 调试需要专用的硬件，在没有硬件的情况下建议采用以下的调试方式：

(1) 使用 LED 灯显示；

(2) 使用串口回传数据至计算机，tinyos 提供了 `printf` 函数，可以将信息通过串口回传到计算机上。

在计算机端键入

```
java net.tinyos.tools.PrintfClient -comm serial@/dev/ttyUSBXXX:telosb
```

即可显示程序中 `printf` 的内容，图图 3.1 所示。

`printf` 的使用见：

http://tinyos.stanford.edu/tinyos-wiki/index.php/The_TinyOS_printf_Library

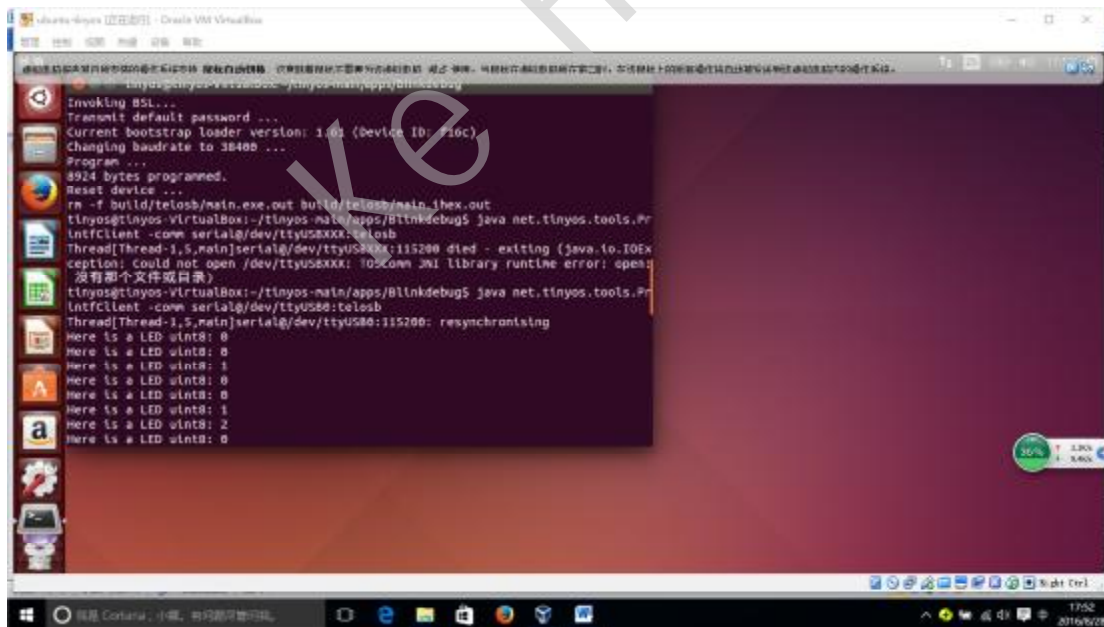


图 3.1 tinyos 的 printf 显示



4 参考文献及网站

TinyOS 相关:

<http://tinyos.net/>

<http://www.tinyprod.net/>

<https://github.com/tinyprod>

<https://github.com/tinyos>

虚拟机相关:

<https://www.virtualbox.org>

Linux 相关:

<http://www.ubuntu.com/global>

Ke HUST