

栗子（Leez）板二次开发指南

1. 入手指南

1.1 配件

Leez-RK3399 的标准套装包含以下配件：

- Leez -RK3399 主板一块
- 12V-2A 电源适配器一个

另外可以选购的配件有：

- Leez 串口模块

另外，在使用过程中，你可能需要以下配件：

- 显示设备
 - 带 HDMI 接口的显示器或电视，及 HDMI 连接线
- 网络
 - 100M/1000M 以太网线缆，及有线路由器
 - WiFi 路由器
- 输入设备
 - USB 无线/有线的鼠标/键盘
 - 红外遥控器(需要接上红外接收器)
- 升级固件，调试
 - Type-C 数据线
 - 串口转 USB 适配器.

Leez-RK3399 支持从以下存储设备启动：

- SD 卡
- eMMC(目前还在开发中)

我们需要将系统固件烧写到 SD 卡或 eMMC 里,这样开发板上电后才能正常启动进入操作系统。

1.2 编译 sdk 和烧写固件 (sdcard 方式)

1.2.1 编译环境:

1. 需要准备一台安装有 **ubuntu16.04** 的机器.

2. **ubuntu16.04** 安装好后按下面步骤执行:

1) 以 root 用户登录(sudo -s)

```
sudo apt-get install repo git-core gitk git-gui gcc-arm-linux-gnueabi\
u-boot-tools device-tree-compiler \
gcc-aarch64-linux-gnu mtools parted libudev-dev libusb-1.0-0-dev
python-linaro-image-tools \
linaro-image-tools autoconf autotools-dev libsigsegv2 m4 intltool libdrm-dev
curl sed make \
binutils build-essential gcc g++ bash patch gzip bzip2 perl tar cpio python
unzip rsync file bc wget \
libncurses5 libqt4-dev libglib2.0-dev libgtk2.0-dev libglade2-dev cvs git
mercurial rsync openssh-client \
subversion asciidoc w3m dblatex graphviz python-matplotlib libc6:i386
libssl-dev texinfo \
liblz4-tool genext2fs lib32stdc++6
```

2) 解压 **linuxsdk_rk3399** 源码包

3) 编译 **linuxsdk_rk3399**

- 进入 sdk 根目录 (kernel 同级目录)
- **./build.sh** (编译带 buildroot 文件系统的完整 image,编译结果在当前目录的 rockdev)

```
root@zhangxg:/data/my_work# ls
app      build.sh  DebianS  distro  envsetup.sh  firmware  kernel  mkfirmware.sh  prebuilts  rkbin  rockdev  sdboot.sh  tools
buildroot  copy.sh  device  docs    external     IMAGE     Makefile  out            repo      rkflash.sh  rootfs  sd_rockdev  u-boot
```

./build.sh all_debian(编译带 debian 文件系统的完整 image,编译结果在当前目录的 rockdev 下面) 注意编译 **debian** 首先要编译一次 **buildroot** (即**./build.sh**)

1.2.2 固件烧写

方法 1: **linuxsdk** 源码目录下烧写方式:

- 进入 sdk 根目录(kernel 同级目录)

```
root@zhangxg:/data/my_work# ls
app      build.sh  DebianS  distro  envsetup.sh  firmware  kernel  mkfirmware.sh  prebuilts  rkbin  rockdev  sdboot.sh  tools
buildroot  copy.sh  device  docs    external     IMAGE     Makefile  out            repo      rkflash.sh  rootfs  sd_rockdev  u-boot
```

- 插入带 **sdccard** 的硬件装置




- `fdisk -l` (查看 `sdcard` 是哪个设备节点, 比如 `/dev/sdc`)
- `./sdboot.sh /dev/sdc` (烧写编译好的固件包到 `sdcard`)

方法 2: 现有编译好的 Image 烧写方式:

- 1) 将编译好的固件包解压

```
root@zhangxg:/data# cd my_buildroot/  
root@zhangxg:/data/my_buildroot# ls  
boot_merger  mkimage  rockdev  sdboot.sh  
root@zhangxg:/data/my_buildroot#  
root@zhangxg:/data/my_buildroot#  
root@zhangxg:/data/my_buildroot#  
root@zhangxg:/data/my_buildroot#
```



- 2) `fdisk -l` (查看 `sdcard` 是哪个设备节点, 比如 `/dev/sdc`)
- 3) `./sdboot.sh /dev/sdc`

1.3 串口调试

1.3.1 选购适配器

网店上有许多 USB 转串口的适配器, 按芯片来分, 有以下几种:

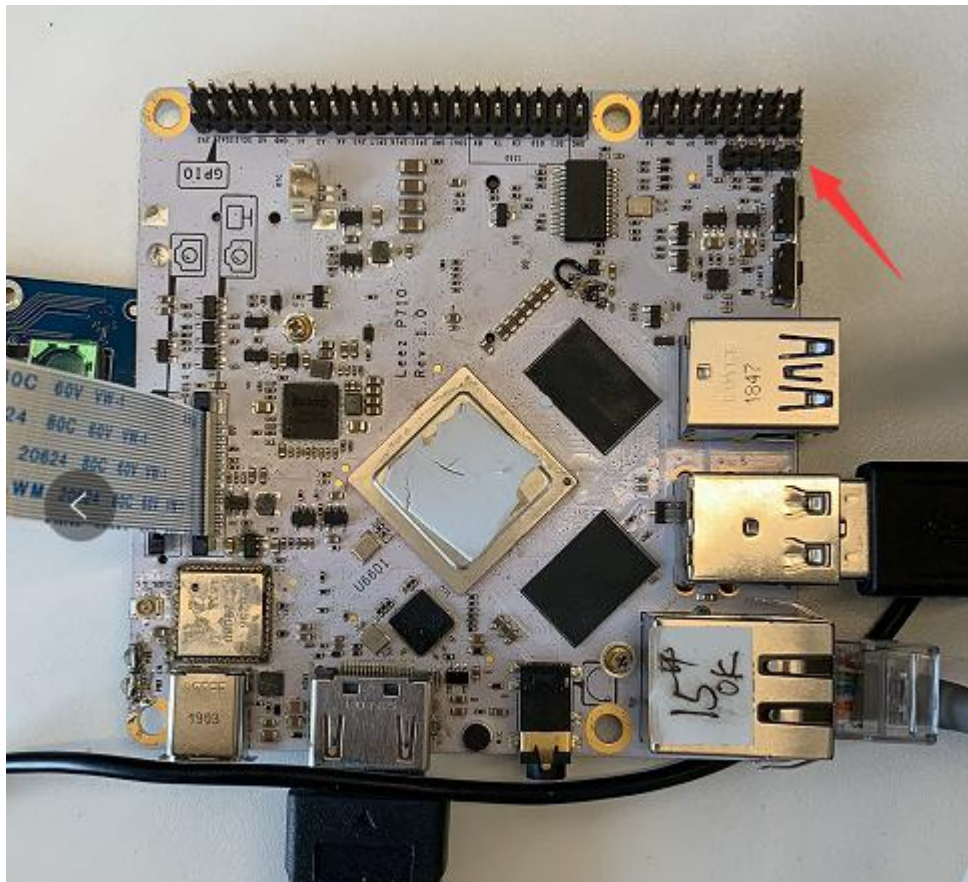
- [CP2104](#)
- PL2303
- CH340
- 注意: RK3399 默认的波特率是 1500000, 有些 USB 转串口芯片波特率无法达到 1500000, 同一芯片的不同系列也可能会有差异, 所以在选购之前一定要确认是否支持。

1.3.2 硬件连接

串口转 USB 适配器, 有四根不同颜色的连接线:

- 红色：3.3V 电源，不需要连接
- 黑色：GND，串口的地线，接开发板串口的 GND 针
- 白色：TXD，串口的输出线，接开发板串口的 TX 针
- 绿色：RXD，串口的输入线，接开发板串口的 RX 针

注：如使用其它串口适配器遇到 TX 和 RX 不能输入和输出的问题，可以尝试对调 TX 和 RX 的连接。



1.3.3 开机

确认主板配件连接无误后，将电源适配器插入带电的插座上，电源线接口插入开发板，开发板第一次加电会自动开机。

开机时，蓝色的电源指示灯会亮起。如果板子接了 HDMI 显示器，可以看到桌面

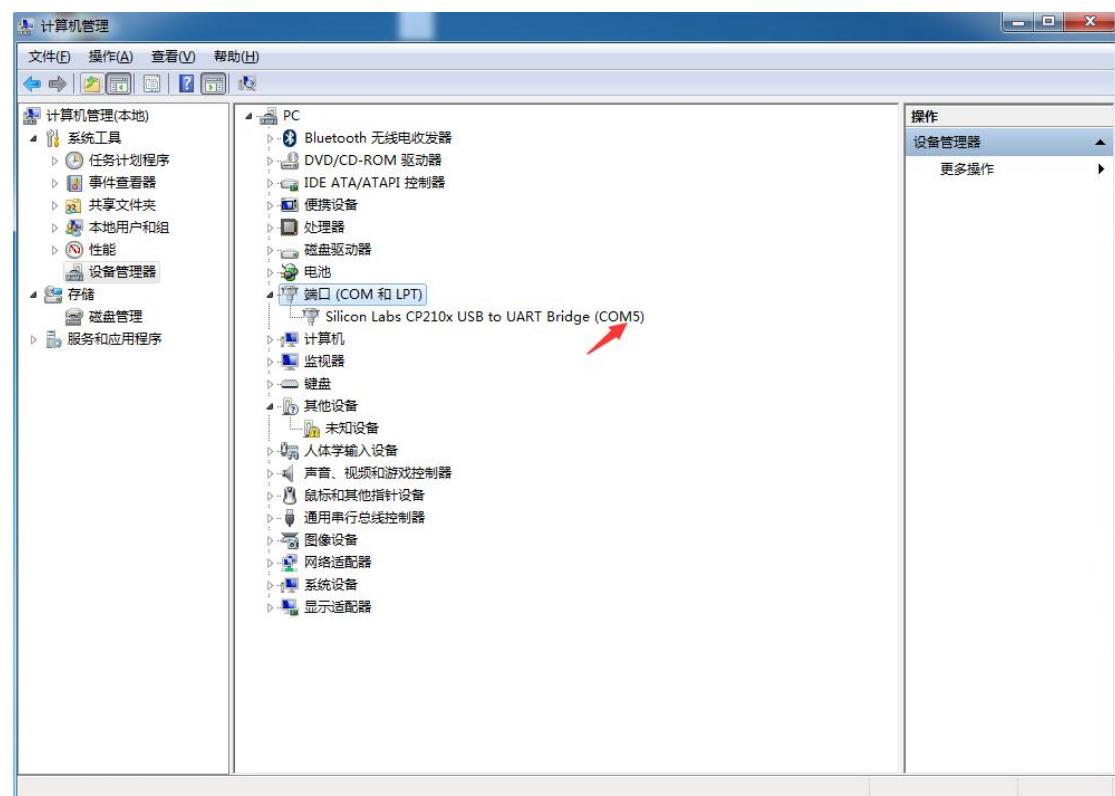
1.3.4 （Windows 端）串口调试

1.3.4.1 安装驱动

下载驱动并安装:

- [CH340\(https://sparks.gogo.co.nz/ch340.html\)](https://sparks.gogo.co.nz/ch340.html)
- [PL2303\(http://www.prolific.com.tw/US/ShowProduct.aspx?pcid=41\)](http://www.prolific.com.tw/US/ShowProduct.aspx?pcid=41)
- [CP210X\(https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers\)](https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers)

插入适配器后,系统会提示发现新硬件,并初始化,之后可以在设备管理器找到对应的 COM 口:

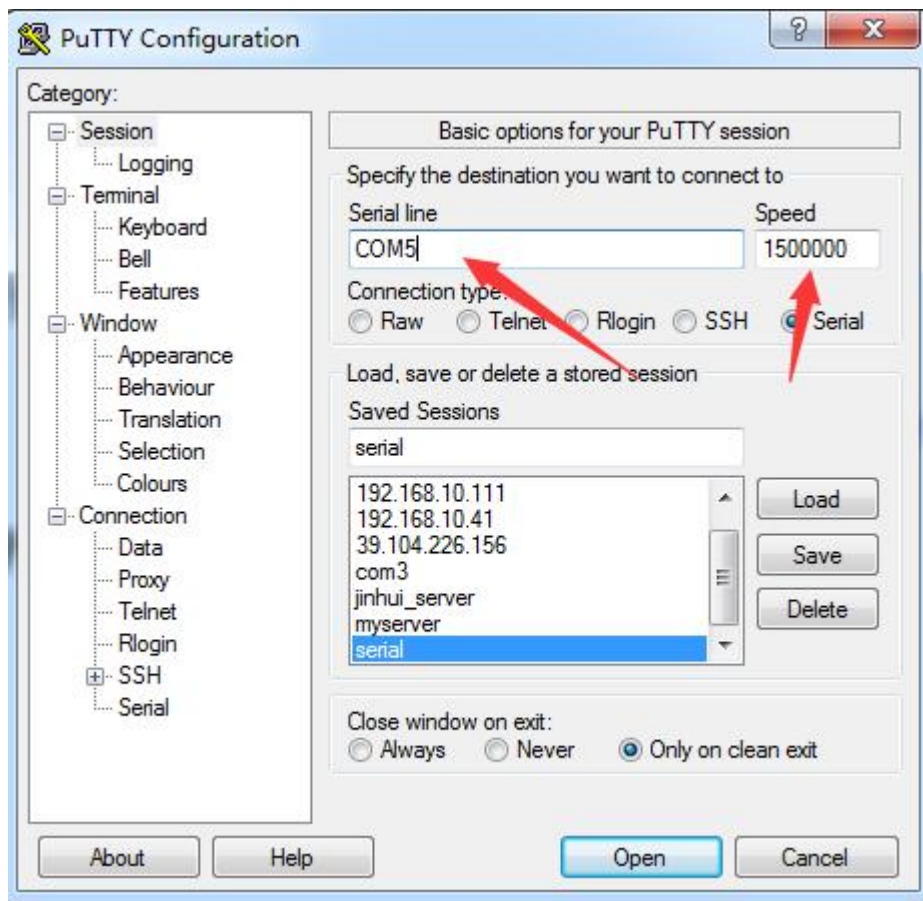


1.3.4.2 putty 串口配置方法

1. 安装 putty(<https://the.earth.li/~sgtatham/putty/0.71/w64/putty-64bit-0.71-installer.msi>)
2. Putty 配置

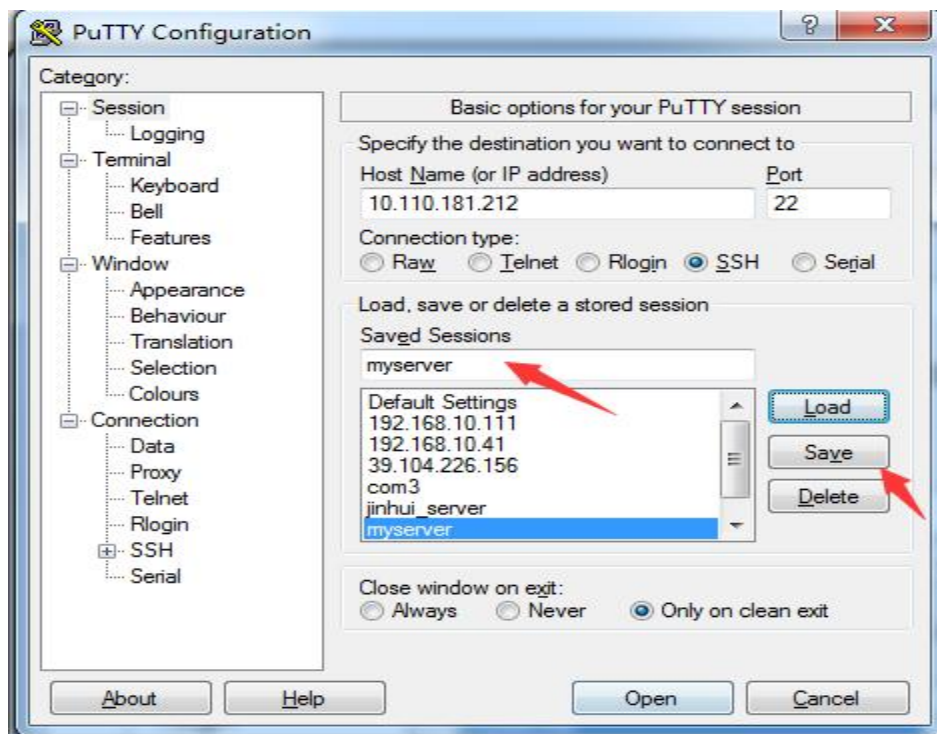
如图 1: SerialLine 箭头这个地方填入的名称要跟设备管理器中（图 2 中的名称一致）

Speed 的地方一定要配成 1500000



3. Putty 配置保存

（为了方便下次使用，再箭头地方输入一个自己喜欢的名称，然后点击 **save** 按钮），然后点击 **open** 就可以开始看串口输出了



2.AI 支持

2.1 Rk3399 TensorFlow 环境安装:

1. `sudo apt-get update`
2. `sudo apt-get install python-dev python-pip`
3. `sudo apt install python3-pip`
4. `sudo apt-get install python3-numpy`
5. `sudo apt-get install python3-dev`
6. `sudo pip3 install --upgrade pip`
7. `mkdir ~/.pip/`
8. `sudo vi ~/.pip/pip.conf`
`[global]`
`index-url = http://mirrors.aliyun.com/pypi/simple/`
`[install]`
`trusted-host=mirrors.aliyun.com`
9. `update-alternatives --list python`
10. `update-alternatives --install /usr/bin/python python /usr/bin/python3.5 2`
11. `update-alternatives --config python`
12. `pip3 install setuptools`
(如果出现问题, 执行下面这两句)
`pip uninstall setuptools`
13. `pip install setuptools==39.1.0`
14. `sudo apt-get install git-core`
`git clone https://github.com/JohnnyChan8/TensorFlow4FireflyRK3399.git`
`pip3 install tensorflow-1.1.0rc1-cp35-cp35m-linux_aarch64.whl`

2.2 测试实例

- 1) `vi hello.py`
- 2) `import tensorflow as tf`
`hello = tf.constant('Hello world!')`
`sess = tf.Session()`
`print(sess.run(hello))`
- 3) `python3 hello.py`

2.3 更复杂的实例:

解压 `ai_example.tar.gz` 后 `python3 mnist.py`