# 智能猫眼门锁系统

# 硬件终端环境搭建

## 硬件材料

Nanopi M3开发板、USB摄像头、3.5mm耳机、USB电源线以及电控灵性锁

## Nanopi M3环境搭建

### 制作SD卡启动系统

1）从官网下载debian系统固件（也可从百度云盘下载<https://pan.baidu.com/s/1V0bVQyHNUslPZkPsND0vzQ> 密码: 323b）和window下的系统烧写工具win32diskimager，另外准备一张8G闪迪SD卡。

2）解压固件，得到.img为后缀的系统文件。使用win32diskimager将固件烧写到sd卡，主界面如图4.1所示。

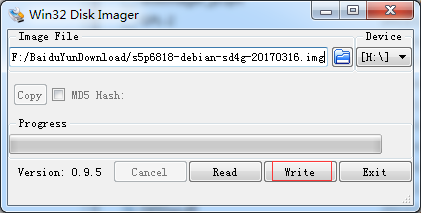


图1.1 win32diskimager工具主界面

1. 系统固件烧写完毕，即SD卡启动系统制作成功。

### 串口登录开发板

1）将sd卡插入开发板，接上电源，启动debian系统。

2）使用USB转串口转化器，连接开发板上的调试串口和PC端，具体接线方式为：板子上的UART0\_TX要接USB转串口转换器上的UART0\_RX，UART0\_RX要接USB转串口转换器上的UART0\_TX，两边的DGND对应接上。连接状况如图1.2。

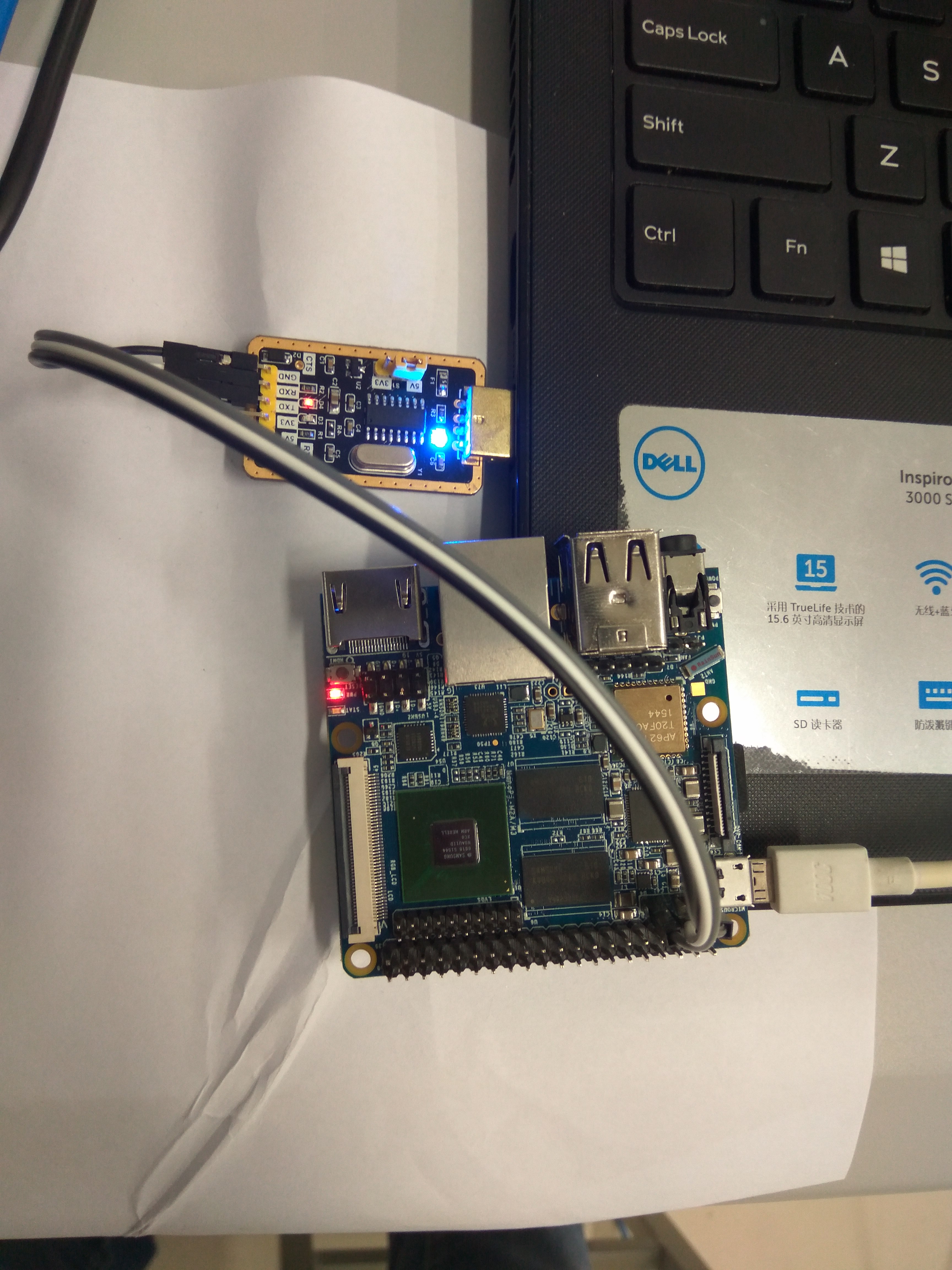


图1.2 USB转串口转换器接线图

1. 使用串口登录工具工具通过串口登录开发板终端。

## 1.3相关工具安装与配置

### 1.3.1 frp客户端安装

Frp工具在本项目中用于支持内网穿透，通过在设备端部署frp客户端，在公网服务器上部署frp服务端，实现智能门锁系统客户端与设备端的直接通信（实际上是通过服务器转发数据），相关理论基础可自行了解。

1）从官网<https://github.com/fatedier/frp/releases>下载frp客户端，如图1.3所示。

图1.3 frp客户端和服务端发行版

1. 配置客户端。将客户端软件包下载到开发板并解压缩。给frpc增加执行权限：

sudo chmod +x ./frpc

### 1.3.2 ffmpeg安装

ffmpeg的作用是完成本地视频流的推送任务，即将设备端采集到的视频信息推送到我们自己搭建的流媒体服务器上。流媒体服务器的搭建过程后面会提及。

1）添加软件源：sudo sh -c 'echo "deb http://www.deb-multimedia.org jessie main" >> /etc/apt/sources.list'

2）更新软件源：sudo apt-get update

3）安装ffmpeg：sudo apt-get install ffmpeg

4）安装完成后，在终端输入ffmpeg命令，会出现命令选项提示，如图1.4所示。

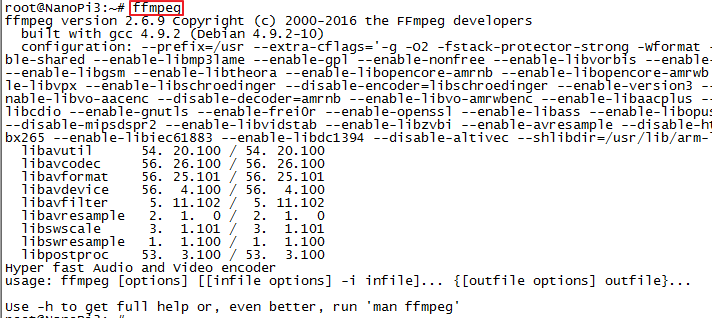


图1.4 ffmpeg命令提示

### 1.3.3 SOX安装

Sox的在本项目中的作用是在硬件终端上播放和录制音频文件。

1)安装alsa包

sudo apt-get install libasound2 alsa-base alsa-utils

2)安装sox

Sox是一种声音文件格式转换工具，可以对多种音频进行处理。本项目使用sox来播放远程语音文件。sox的安装也极其简单，安装过程如下

sudo apt-get install sox

sudo apt-get install sox libsox-fmt-all

### 1.3.4 安装gpio库

Gpio库基于python编写，用于操作Nanopi M3板子上的GPIO口。具体来说，我们可以使用gpio库提供的接口对GPIO口的高低电平直接进行读写。Gpio库的安装过程如下：

1. 先安装pip(基于python2):sudo apt-get install python-pip
2. 再安装gpio库：pip install gpio

若出现以下错误：: symbol SSLv2\_client\_method, version OPENSSL\_1.0.0 not defined in file libssl.so.1.0.0 with link time reference。

说明OpenSSL没有安装，使用如下命令安装OpenSSL.

apt-get install openssl libssl-dev

### 1.3.5 mjpg-streamer安装

Mjpg-streamer是一个轻量级的视频服务器软件。本项目中使用其通过USB摄像头获取视频信息，进而提供http视频流。再利用ffmpeg将本地http流转化成rtmp流并推送至流媒体服务器。适用于debian系统的mjpg-streamer已经编译成功，直接使用即可。位于源码中的door/目录下。

# 客户端环境搭建

1. 开发环境：windows

2. IDE：Qt Creator(下载地址 <http://download.qt.io/archive/qt/> 本项目使用qt5.8)

3. windows qt creator使用mysql请参考 <http://lbbit.com/?p=19>

# 三、云平台环境搭建

云平台环境的部署方式有很多，可以部署在单机，部署到专用云服务器，还可以部署Doker容器中。云平台借助SpringBoot框架内部集成的Tomcat，所以在部署环境中不需要搭建Tomcat环境,必须要搭建的有JDK环境和Mysql数据库环境，下面详细说明。

## 3.1 JDK环境安装

JDK环境包括了Java运行所需的环境和Java虚拟机。云平台选用JDK1.8版本、SpringBoot1.5.9版本，配合Maven3.3.9项目构建工具。

JDK可以到官方网站下载,Window系统需下载可执行文件。顺利安装成功后进行window环境变量的设置，以便系统能够识别Java命令。设置完环境变量后对Java进行简单的测试，window控制台中输入java -version。出现结果如图3.1所示。



图3.1 JDK版本图

## 3.2 Mysql环境配置

Mysql作为开源的关系型数据库具有易用性，稳定性。安装过程中需要初始化设置用户名和密码，这里都置为root,数据库端口为3306，字符集统一设置为utf-8。

云平台中的Mysql数据源配置如图3.2所示。

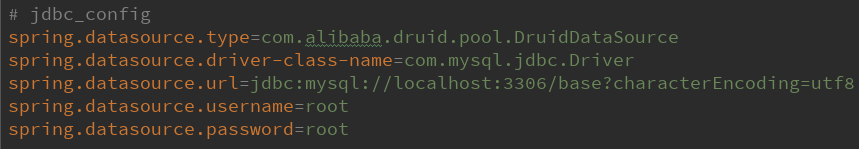


图3.2 Mysql数据源配置图

## 3.3 系统打包部署

借助Maven项目管理工具来对云平台进行打包部署。具体有两种方式，一是在命令行中打开到maven项目目录下，输入命令jar：mvn clean install -Dmaven.test.skip=true。还可以在IDE中利用Maven插件，首先清理,再点击打包,出现如图3.3所示就代表云平台项目打包成功。

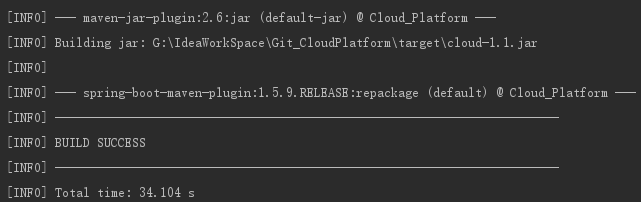


图3.3 Maven打包项目示意图

SpringBoot的运行不需要外部容器只需要在命令行中打开jar包所在目录，输入命令java -jar java -jar cloud.jar十分便捷。成功运行项目如图3.4所示。

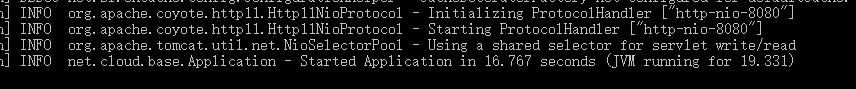


图3.4 项目成功部署图

## 3.4 rtmp流媒体服务器搭建

该流媒体服务器搭建在外网服务器，用于接收并分发设备端推送的rtmp视频流。搭建过程较为简单，可参考<http://lbbit.com/?p=197>

## 3.5 frp服务端搭建

1）从官网<https://github.com/fatedier/frp/releases>下载frp服务端，如图3.5所示。

图3.5 frp客户端和服务端发行版

2)配置服务端。将服务端软件包上传到云服务器并解压缩。给frps增加执行权限，再执行frps，启动frp服务端：

sudo chmod +x ./frps

./frps -c ./frps.ini

1. frp工具使用方法：

环境搭建demo：<https://blog.csdn.net/qq_25351621/article/details/78947477>

一个比较全面的工具使用介绍： [https://studygolang.com/p/frp]( https:/studygolang.com/p/frp)