

# 第一课 安装树莓派

## 【硬件准备】

- 树莓派 3B 或 3B+



- 8G 以上 tf 卡



- USB tf 卡读写卡器



- Windows 或 MacOS 工作电脑

## 【操作步骤】

1. 下载树莓派镜像 (<https://www.raspberrypi.org/downloads/raspbian/>), 解压缩
2. 下载安装 etcher (<https://etcher.io/>)
3. 使用 etcher 将镜像写入 tf 卡
4. 配置打开 sshd, 配置树莓派的 wifi
5. 启动树莓派, 确认树莓派的 IP 地址, 并通过 ssh 登录树莓派

## 【参考】

- wpa\_supplicant.conf 文件

```
country=CN
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
    ssid="your_wifi_ssid"
    psk="your_wifi_password"
    key_mgmt=WPA-PSK
    priority=1
}
```

- putty 下载地址

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

注：对于网络不好，或者希望偷懒的用户，可以直接使用 HACHina 的镜像文件，其中已经安装好了 HomeAssistant、NodeRed、JupyterNotebook、Samba 等软件，可以直接使用。

地址：<https://www.hachina.io/8408.html>

## 第二课 树莓派基础配置

### 【操作步骤】

1. 查看系统基础信息
2. 修改 pi 用户密码
3. 使用 raspi-config 修改时区、打开 vnc
4. 修改 APT 与 PIP 的配置
5. 安装一些常用的基础库

### 【参考】

- /etc/apt/sources.list 文件  
`deb http://mirrors.aliyun.com/raspbian/raspbian/ stretch main non-free contrib rpi`  
`deb-src http://mirrors.aliyun.com/raspbian/raspbian/ stretch main non-free contrib rpi`
- /etc/pip.conf 文件  
`[global]`  
`index-url=https://mirrors.aliyun.com/pypi/simple/`  
`extra-index-url=https://www.piwheels.org/simple`
- 安装一些常用的基础库  
`sudo apt-get install libevent-dev libjpeg-dev build-essential libssl-dev libffi-dev python3-dev libavahi-compat-libdnssd-dev libatlas-base-dev autoconf python3-venv`
- VNC Viewer  
<https://www.realvnc.com/en/connect/download/viewer/>

## 第三课 安装 Samba 和 Jupyter Notebook

### 【操作步骤】

1. 安装、配置、使用 Samba
2. 安装、配置、使用 Jupyter Notebook

### 【参考】

- Samba 命令  
安装: `sudo apt-get install samba samba-common`  
增加用户 pi 并设置密码: `sudo smbpasswd -a pi`  
重启 samba 服务: `sudo systemctl restart smbd`
- Samba 配置文件位置  
`/etc/samba/smb.conf`
- 如果打开共享是空的 (未显示共享目录)  
在 Windows 的 DOS 窗口中, 运行命令:  
`net use z: \\raspberrypi password /USER:pi`  
修改其中 password 为你的 password, 如果要保持 z 盘长久映射, 在命令中增加:  
`/PERSISTENT:YES`  
*注: 很多 windows 缺省使用匿名访问共享, 而我们的共享是以 pi 用户共享的, 所以就看不到。*  
*使用这个命令, 指定使用 pi 用户访问, 将共享的目录映射为 windows 下的 z 盘*
- Jupyter Notebook 命令  
安装: `sudo pip3 install jupyter`  
生成配置文件: `jupyter notebook --generate-config`  
设置访问密码: `jupyter notebook password`
- 修改 Jupyter Notebook 的配置 (文件位置 `~/jupyter/jupyter_notebook_config.py`)  
在文件中找到下面的 4 个配置项, 去除前面的注释符号 (#), 并修改等号后的值为下面的值。  
`c.NotebookApp.ip = '0.0.0.0'`  
`c.NotebookApp.open_browser = False`  
`c.NotebookApp.port = 8888`  
`c.NotebookApp.notebook_dir = '/home/pi/'`

## 第四课 安装 HomeAssistant

### 【操作步骤】

1. 安装 HomeAssistant
2. 启动 HomeAssistant
3. 第一次启动，创建用户，登录
4. 查看 HomeAssistant 的错误日志
5. 查看并修改 HomeAssistant 的配置文件

### 【参考】

- HomeAssistant 命令  
安装: `sudo pip3 install homeassistant`  
启动: `hass`  
升级: `sudo pip3 install homeassistant --upgrade`

- yaml 格式  
<https://www.hachina.io/docs/335.html>

- 比特币配置

sensor:

.....

- platform: bitcoin

display\_options:

- exchangerate

- trade\_volume\_btc

## 第五课 HomeAssistant 自启动

### 【操作步骤】

1. 自启动配置文件
2. 设置自启动
3. 查看日志输出
4. 手动控制服务
5. 同样操作，设置 Jupyter-notebook 的自启动

### 【参考】

- /etc/systemd/system/home-assistant@pi.service 文件

```
[Unit]
Description=Home Assistant
After=network.target
[Service]
Type=simple
User=%i
ExecStart=/usr/local/bin/hass
[Install]
WantedBy=multi-user.target
```
- /etc/systemd/system/jupyter-notebook@pi.service 文件

```
[Unit]
Description=Jupyter Notebook
[Service]
Type=simple
ExecStart=/usr/local/bin/jupyter-notebook
User=%i
[Install]
WantedBy=multi-user.target
```
- 自启动服务相关命令 (以 home-assistant@pi 服务为例)
  - 重载服务配置: `sudo systemctl --system daemon-reload`
  - 将服务加入自启动: `sudo systemctl enable home-assistant@pi`
  - 将服务移除自启动: `sudo systemctl disable home-assistant@pi`
  - 手工启动服务: `sudo systemctl start home-assistant@pi`
  - 手工停止服务: `sudo systemctl stop home-assistant@pi`
  - 手工重启服务: `sudo systemctl restart home-assistant@pi`
  - 查看服务输出: `sudo journalctl -f -u home-assistant@pi`