

2.1 网络与蓝牙配置

本章节主要介绍开发板有线、无线网络配置的修改方法。

有线网络{#config_ethnet}

PROF

开发板有线网络默认采用静态IP配置，初始IP地址为192.168.1.10。用户可通过如下方法实现静态、DHCP模式的切换。

修改静态IP配置

开发板静态网络配置保存在/etc/network/interfaces文件中，通过修改address、netmask、gateway等字段，可完成对静态IP配置的修改，metric是网络优先级配置，设置为700是为了让有线网络的优先级更低，当有线和无线网络同时使能时优先会使用无线网络，例如：

```
sudo vim /etc/network/interfaces
```

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
```

```
source-directory /etc/network/interfaces.d
auto eth0
iface eth0 inet static
    address 192.168.1.10
    netmask 255.255.255.0
    gateway 192.168.1.1
    metric 700
```

修改完成后，命令行输入`sudo restart_network`命令让配置生效。

修改DHCP配置

DHCP(Dynamic Host Configuration Protocol，动态主机配置协议)通常被应用在局域网络环境中，主要作用是集中的管理、分配IP地址，使网络环境中的主机动态的获得IP地址、Gateway地址、DNS服务器地址等信息，并能够提升地址的使用率。

开发板的DHCP网络配置保存在`/etc/network/interfaces`文件，通过修改`eth0`相关配置，可完成对DHCP模式的修改，例如：

```
sudo vim /etc/network/interfaces
```

```
source-directory /etc/network/interfaces.d
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
    metric 700
```

修改完成后，命令行输入`sudo restart_network`命令让配置生效。

修改MAC地址配置

如需修改开发板默认MAC地址，可通过在`/etc/network/interfaces`文件中增加`pre-up`配置信息，指定用户需要的MAC地址，例如：

```
sudo vim /etc/network/interfaces
```

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d
auto eth0
iface eth0 inet static
    address 192.168.1.10
```

```
netmask 255.255.255.0
gateway 192.168.1.1
pre-up ifconfig eth0 hw ether 00:11:22:9f:51:27
```

修改完成后，**reboot**重启让配置生效。

无线网络

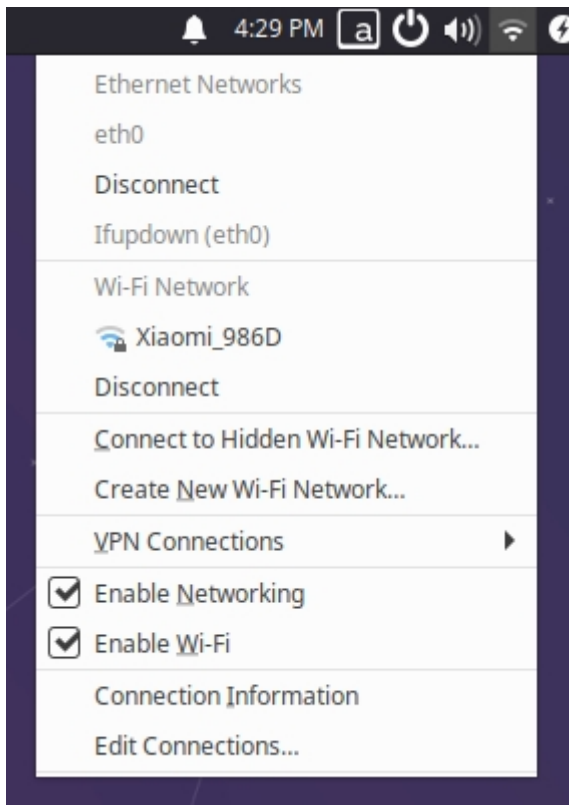
PROF

开发板集成了2.4GHz无线WiFi模块，支持Soft AP和Station两种模式，默认运行在Station模式下。下面介绍两种模式的使用方法。

Station模式

Station模式下，开发板作为客户端，接入路由器无线热点进行联网。

- 对于使用Ubuntu Desktop版本系统的用户，可点击桌面右上角Wi-Fi图标，选择对应热点并输入密码以完成网络配置，如下图：



- 对于使用Ubuntu Server版本系统的用户，可通过命令行完成无线网络配置，步骤如下：
1. 使用`sudo nmcli device wifi rescan`命令扫描热点。如返回如下信息，说明扫描过于频繁，需要稍后再试

```
root@ubuntu:~# sudo nmcli device wifi rescan
Error: Scanning not allowed immediately following previous scan.
```

2. 使用`sudo nmcli device wifi list`命令列出扫描到的热点
3. 使用`sudo wifi_connect "SSID" "PASSWD"`命令连接热点，返回如下信息，说明网络连接成功

```
root@ubuntu:~# sudo wifi_connect "WiFi-Test" "12345678"
Device 'wlan0' successfully activated with 'd7468833-4195-45aa-aa33-3d43da86e1a7'.
```

如果连接热点后，返回如下信息，说明热点没有找到，可以执行`sudo nmcli device wifi rescan`命令重新扫描后再次连接

```
root@ubuntu:~# sudo wifi_connect "WiFi-Test" "12345678"
Error: No network with SSID 'WiFi-Test' found.
```

Soft AP模式

开发板无线网络默认运行在Station模式下，如需使用Soft AP模式，请按照以下步骤进行配置。

1. 安装hostapd和isc-dhcp-server

```
sudo apt update
sudo apt install hostapd
sudo apt install isc-dhcp-server
```

2. 运行sudo vim /etc/hostapd.conf命令来配置hostapd.conf，主要关注下面几个字段:

```
interface=wlan0 #作为AP热点的网卡
ssid=Sunrise #WiFi名字
wpa=2 #0为WPA 2为WPA2 一般为2
wpa_key_mgmt=WPA-PSK #加密算法 一般为WPA-PSK
wpa_passphrase=12345678 #密码
wpa_pairwise=CCMP #加密协议，一般为CCMP
```

- 无密码的热点配置，请在hostapd.conf文件添加以下内容：

```
interface=wlan0
driver=nl80211
ctrl_interface=/var/run/hostapd
ssid=Sunrise
channel=6
ieee80211n=1
hw_mode=g
ignore_broadcast_ssid=0
```

- 有密码的热点配置，请在hostapd.conf文件添加以下内容：

```
interface=wlan0
driver=nl80211
ctrl_interface=/var/run/hostapd
ssid=Sunrise
channel=6
ieee80211n=1
hw_mode=g
ignore_broadcast_ssid=0
wpa=2
wpa_key_mgmt=WPA-PSK
rsn_pairwise=CCMP
wpa_passphrase=12345678
```

3. 配置isc-dhcp-server文件，步骤如下：

- 执行 `sudo vim /etc/default/isc-dhcp-server` 修改 `isc-dhcp-server` 文件，添加如下定义的网络接口：

```
INTERFACESv4="wlan0"
```

- 执行 `sudo vim /etc/dhcp/dhcpd.conf` 修改 `dhcpd.conf` 文件，取消以下字段的注释：

```
authoritative;
```

- 然后在 `/etc/dhcp/dhcpd.conf` 文件末尾增加以下配置：

```
subnet 10.5.5.0 netmask 255.255.255.0 { #网段和子网掩码
    range 10.5.5.100 10.5.5.254; #可获取的IP范围
    option subnet-mask 255.255.255.0; #子网掩码
    option routers 10.5.5.1; #默认网关
    option broadcast-address 10.5.5.31; #广播地址
    default-lease-time 600; #默认租约期限，单位秒
    max-lease-time 7200; #最长租约期限，单位秒
}
```

4. 停止 `wpa_supplicant` 服务，并重启 `wlan0`

```
systemctl stop wpa_supplicant

ip addr flush dev wlan0
sleep 0.5
ifconfig wlan0 down
sleep 1
ifconfig wlan0 up
```

5. 按如下步骤启动 `hostapd` 服务

- 执行 `sudo hostapd -B /etc/hostapd.conf` 命令

```
root@ubuntu:~# sudo hostapd -B /etc/hostapd.conf

Configuration file: /etc/hostapd.conf
Using interface wlan0 with hwaddr 08:e9:f6:af:18:26 and ssid
"sunrise"
wlan0: interface state UNINITIALIZED->ENABLED
wlan0: AP-ENABLED
```

- 通过`ifconfig`命令，配置无线接口`wlan0`的IP和网段，注意要跟第三步的配置保持一致

```
sudo ifconfig wlan0 10.5.5.1 netmask 255.255.255.0
```

- 最后开启`dhcp`服务器，连上热点会从`10.5.5.100`到`10.5.5.255`之间分配一个ip地址给客户端

```
sudo ifconfig wlan0 10.5.5.1 netmask 255.255.255.0
sudo systemctl start isc-dhcp-server
sudo systemctl enable isc-dhcp-server
```

6. 连接开发板热点，例如 `sunrise`



7. 如需切换回`Station`模式，可按如下方式进行：

```
# 停止 hostapd
killall -9 hostapd

# 清除 wlan0 的地址
ip addr flush dev wlan0
sleep 0.5
ifconfig wlan0 down
sleep 1
ifconfig wlan0 up

# 重启 wpa_supplicant
systemctl restart wpa_supplicant

# 连接热点，具体操作可以查看上一章节“无线网络”
wifi_connect "WiFi-Test" "12345678"
```

DNS服务

DNS(Domain Name Server)是进行域名(domain name)和与之相对应的IP地址转换的服务器。

开发板DNS配置通过/etc/systemd/resolved.conf文件管理，用户可通过修改该文件完成DNS相关配置，步骤如下：

1. 修改resolved.conf文件，添加DNS服务器地址，例如：

```
DNS=8.8.8.8 114.114.114.114
```

2. 通过如下命令，使能DNS配置：

```
sudo systemctl restart systemd-resolved
sudo systemctl enable systemd-resolved
sudo mv /etc/resolv.conf /etc/resolv.conf.bak
sudo ln -s /run/systemd/resolve/resolv.conf /etc/
```

系统更新

出于系统安全、稳定性的考虑，推荐用户安装完系统后，通过apt命令对系统进行更新。

在`/etc/apt/source.list`文件中，保存了`apt`命令的软件源列表，在安装软件前，需要先通过`apt`命令更新`package`列表。

首先打开终端命令行，输入如下命令：

```
sudo apt update
```

其次，升级所有已安装的软件包到最新版本，命令如下：

```
sudo apt full-upgrade
```

推荐使用`full-upgrade`而不是`upgrade`选项，这样当相关依赖发生变动时，也会同步更新依赖包。

当运行`sudo apt full-upgrade`命令时，系统会提示数据下载和磁盘占用大小，但是`apt`不会检查磁盘空间是否充足，建议用户通过`df -h`命令手动检查。此外，升级过程中下载的deb文件会保存在`/var/cache/apt/archives`目录中，用户可以通过`sudo apt clean`命令删除缓存文件以释放磁盘空间。

执行`apt full-upgrade`命令后，可能会重新安装驱动、内核文件和部分系统软件，建议用户手动重启设备使更新生效，命令如下：

```
sudo reboot
```

蓝牙配置

初始化

开发板蓝牙功能默认没有开启，需要执行 `/usr/bin/startbt6212.sh`脚本进行初始化，该脚本完成以下工作：

- 复位蓝牙
- 创建 `messagebus` 用户和用户组，`dbus-daemon` 程序运行时需要使用该用户
- 运行 `brcm_patchram_plus` 完成蓝牙的驱动加载和固件加载
- 循环检查 `/sys/class/bluetooth/hci0` 目录是否存在，确认蓝牙驱动已经正常运行
- 出现 **Done setting line discipline** 表示蓝牙启用成功
- 执行 `hciconfig hci0 up` 完成蓝牙的Link Up
- 执行 `hciconfig hci0 piscan` 进行蓝牙扫描，本步骤可以根据情况去掉

脚本执行成功后的log如下：

```

root@ubuntu:~# sudo startbt6212.sh
Waiting for bluetooth initialize.....Done setting line discipline
.Done
Check Bluetooth State...unblocked
Set Bluetooth Up...
Set Bluetooth piscan...
hci0: Type: Primary Bus: UART
BD Address: 08:E9:F6:AE:F8:8B ACL MTU: 1021:8 SCO MTU: 64:1
UP RUNNING PSCAN ISCAN
RX bytes:759 acl:0 sco:0 events:48 errors:0
TX bytes:2522 acl:0 sco:0 commands:48 errors:0

root@ubuntu:~#

```

此外，用户可以使用命令查询蓝牙进程是否正常，命令如下：

```

ps ax | grep "/usr/bin/dbus-daemon\|/usr/lib/bluetooth/bluetoothd"
/usr/bin/dbus-daemon

/usr/lib/bluetooth/bluetoothd

```

配网连接

执行 `sudo bluetoothctl` 进入交互模式下的蓝牙配置界面，出现了类似下图的设备信息表示蓝牙被识别到了，然后用 `show` 来查看蓝牙信息，留意蓝牙的 `powered` 和 `discoverable` 状态。

```

root@ubuntu:~# bluetoothctl
Agent registered
[CHG] Controller 08:E9:F6:AF:18:27 Pairable: yes
[bluetooth]# show
Controller 08:E9:F6:AF:18:27 (public)
    Name: ubuntu
    Alias: ubuntu
    Class: 0x00000000
    Powered: yes
    Discoverable: yes
    DiscoverableTimeout: 0x000000b4
    Pairable: yes
    UUID: Generic Attribute Profile (00001801-0000-1000-8000-00805f9b34fb)
    UUID: Generic Access Profile (00001800-0000-1000-8000-00805f9b34fb)
    UUID: PnP Information (00001200-0000-1000-8000-00805f9b34fb)
    UUID: A/V Remote Control Target (0000110c-0000-1000-8000-00805f9b34fb)
    UUID: A/V Remote Control (0000110e-0000-1000-8000-00805f9b34fb)
    Modalias: usb:v1D6Bp0246d0535
    Discovering: no
Advertising Features:
    ActiveInstances: 0x00
    SupportedInstances: 0x05
    SupportedIncludes: tx-power
    SupportedIncludes: appearance
    SupportedIncludes: local-name

```

执行 `power on` 使能蓝牙，如下图所示：

```

[bluetooth]# power on
Changing power on succeeded
[bluetooth]#

```

为了能够使蓝牙被附近的设备发现，需要执行 `discoverable on` 使能蓝牙并打开蓝牙可发现属性，如下图所示：

```
[bluetooth]# discoverable on
Changing discoverable on succeeded
[CHG] Controller 08:E9:F6:AF:18:27 Discoverable: yes
[bluetooth]#
```

此时使用手机或者电脑扫描蓝牙就可以发现 **ubuntu** 这个名称的蓝牙设备：



接下来测试蓝牙的主动扫描功能，在 **bluetoothctl** 的交互界面输入 **scan on** 即可打开主动扫描，它会周期性地打印附近的设备，可以看到已经发现了我的手机设备，**scan off** 关闭扫描功能并汇总打印扫描到的蓝牙设备：

```
[bluetooth]# scan on
Discovery started
[CHG] Controller 08:E9:F6:AF:18:27 Discovering: yes
[NEW] Device 55:C1:B8:F0:C1:5D 55-C1-B8-F0-C1-5D
[NEW] Device CA:2B:14:31:34:94 CA-2B-14-31-34-94
[NEW] Device C8:28:32:D2:41:7F C8-28-32-D2-41-7F
[NEW] Device 54:48:E6:CB:30:DD 54-48-E6-CB-30-DD
[CHG] Device C8:28:32:D2:41:7F RSSI: -69
[CHG] Device C8:28:32:D2:41:7F ServiceData Key: 0000fdaa-0000-1000-8000-00805f9b34fb
[CHG] Device C8:28:32:D2:41:7F ServiceData Value:
  30 35 31 21 26 22 a7 01 08 02 06 01 2d 6a 34 14 051!&".....-j4.
  67 83 e4 b8 bb e5 8d a7 e7 ff g.....
[NEW] Device F0:5E:CD:B3:30:06 BTM1020262
[NEW] Device C8:28:32:5A:56:72 C8-28-32-5A-56-72
[bluetooth]#
```

```
[bluetooth]# scan off
[CHG] Device F0:5E:CD:B3:30:06 RSSI is nil
[DEL] Device F0:5E:CD:B3:30:06 BTM1020262
[CHG] Device CC:2D:B7:E5:DD:ED RSSI is nil
[CHG] Device 5B:CE:4A:AD:5C:E9 TxPower is nil
[CHG] Device 5B:CE:4A:AD:5C:E9 RSSI is nil
[CHG] Device B4:60:ED:AB:23:29 RSSI is nil
[CHG] Device 56:FF:E4:72:2F:96 TxPower is nil
[CHG] Device 56:FF:E4:72:2F:96 RSSI is nil
[CHG] Device C8:28:32:5A:56:72 RSSI is nil
[CHG] Device 54:48:E6:CB:30:DD RSSI is nil
[CHG] Device C8:28:32:D2:41:7F RSSI is nil
[CHG] Device CA:2B:14:31:34:94 RSSI is nil
[CHG] Device 55:C1:B8:F0:C1:5D TxPower is nil
[CHG] Device 55:C1:B8:F0:C1:5D RSSI is nil
[CHG] Controller 08:E9:F6:AF:18:27 Discovering: no
Discovery stopped
[bluetooth]#
```

然后就是和其他蓝牙的配对：

- 配对命令：`pair [targetMAC]`，输入该命令后，根据提示输入`yes`，对端蓝牙设备选择配对选项完成配对。
- 配对成功后可以使用`trust [targetMAC]`来让下次自动连接

```
[bluetooth]# pair CC:2D:B7:E5:DD:ED
Attempting to pair with CC:2D:B7:E5:DD:ED
[CHG] Device CC:2D:B7:E5:DD:ED Connected: yes
Request confirmation
[agent] Confirm passkey 125453 (yes/no): yes
[CHG] Device CC:2D:B7:E5:DD:ED Modalias: bluetooth:v004Cp7109d0E70
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 00000000-deca-fade-deca-deafdecacafe
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 00001000-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 0000110a-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 00001116-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 0000111f-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 0000112f-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 00001132-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 00001200-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 00001801-0000-1000-8000-00805f9b34fb
[CHG] Device CC:2D:B7:E5:DD:ED UUIDs: 02030302-1d19-415f-86f2-22a2106a0a77
[CHG] Device CC:2D:B7:E5:DD:ED ServicesResolved: yes
[CHG] Device CC:2D:B7:E5:DD:ED Paired: yes
Pairing successful
[CHG] Device CC:2D:B7:E5:DD:ED ServicesResolved: no
[CHG] Device CC:2D:B7:E5:DD:ED Connected: no
[bluetooth]# trust CC:2D:B7:E5:DD:ED
[CHG] Device CC:2D:B7:E5:DD:ED Trusted: yes
Changing CC:2D:B7:E5:DD:ED trust succeeded
[bluetooth]#
```

经过以上操作后，蓝牙的扫描、配对的基本功能就完成了，如需使用更多功能，可查阅 [BlueZ](#) 的官方帮助说明。