

SSD20X AP 模式使用方法

1、内核开启 WiFi 配置

make infinity2m_ssc011a_s01a_minigui_defconfig (根据自己实际配置使用 defconfig)

make menuconfig

进入 Networking support -> Wireless, 将下面选项由模块编译改为 build-in

```
--- Wireless
<+> cfg80211 - wireless configuration API
[ ] nl80211 testmode command
[ ] enable developer warnings
[ ] cfg80211 certification onus
[*] enable powersave by default
[ ] cfg80211 DebugFS entries
[ ] use statically compiled regulatory rules database
[*] cfg80211 wireless extensions compatibility
< > Generic IEEE 802.11 Networking Stack (mac80211)
```

进入 Device Drivers->Generic Driver options, 将下面选项编译方式改为 build-in

```
[*] Support for uevent helper
(/sbin/mdev) path to uevent helper
[*] Maintain a devtmpfs filesystem to mount at /dev
[*] Automount devtmpfs at /dev, after the kernel mounted the rootfs
[ ] Select only drivers that don't need compile-time external firmware
[ ] Prevent firmware from being built
-* Userspace firmware loading support
[*] Include in-kernel firmware blobs in kernel binary
() External firmware blobs to build into the kernel binary
[ ] Fallback user-helper invocation for firmware loading
[ ] Allow device coredump
[ ] Driver Core verbose debug messages
[ ] Managed device resources verbose debug messages
[ ] Test driver remove calls during probe (UNSTABLE)
[*] DMA Contiguous Memory Allocator
*** Default contiguous memory area size: ***
(2) Size in Mega Bytes
Selected region size (Use mega bytes value only) --->
(4) Maximum PAGE_SIZE order of alignment for contiguous buffers
```

进入 Networking support → Networking options 将 802.1d Ethernet Bridging 这个配置文件配置成 module, 编译出 llc.ko stp.ko bridge.ko

```
< > IP: IPsec transport mode
< > IP: IPsec tunnel mode
< > IP: IPsec BEET mode
< > INET: socket monitoring interface
[*] TCP: advanced congestion control ---->
[ ] TCP: MD5 Signature Option support (RFC2385)
< > The IPv6 protocol ----
[ ] Security Marking
[ ] Timestamping in PHY devices
[ ] Network packet filtering framework (Netfilter) ----
< > The DCCP Protocol ----
< > The SCTP Protocol ----
< > The RDS Protocol
< > The TIPC Protocol ----
< > Asynchronous Transfer Mode (ATM)
< > Layer Two Tunneling Protocol (L2TP) ----
<M> 802.1d Ethernet Bridging
[*] IGMP/MLD snooping (NEW)
< > Distributed Switch Architecture
< > 802.1Q/802.1ad VLAN Support
+ (+)
<Select> < Exit > < Help > < Save > < L
```

编译内核: make clean -j16;make -j16

2、修改 **demo.sh**，将最后一行（**./zkgui &**）删除(注释)，然后在控制台输入一下命令：

```
sync
reboot
```

3、将 **/kernel/modules** 下面的 **llc.ko stp.ko bridge.ko** 文件拷到 U 盘中，复制到板子上的 **/home** 目录，然后输入以下命令

```
insmod llc.ko
insmod stp.ko
insmod bridge.ko
```

4、将 **ssw101b_wifi_HT40_usb.ko** 拷到 U 盘中，复制到板子上 **/config/wifi** 目录，替换原来的文件，然后输入以下命令

```
insmod ssw101b_wifi_HT40_usb.ko
```

5、建立桥接

```
ifconfig p2p0 up
ifconfig eth0 up
ifconfig p2p0 0.0.0.0(一定要配置成 0)
ifconfig eth0 0.0.0.0(一定要配置成 0)
brctl addbr br0
brctl addif br0 p2p0
brctl addif br0 eth0
ifconfig br0 up
ifconfig br0 172.19.24.186 netmask 255.255.255.0 (ip 要根据实际情况设置)
route add default gw 172.19.24.254 (要根据实际情况设置)
查看一下是否有 /var/run 目录，没有的话就新建一个：mkdir /var/run
cd /config/wifi
export LD_LIBRARY_PATH=/lib:$PWD:$LD_LIBRARY_PATH
./hostapd -B ./hostapd.conf &
```

```

/config/wifi # export LD_LIBRARY_PATH=/lib:$PWD:$LD_LIBRARY_PATH
/config/wifi # ./hostapd -B ./hostapd.conf &
/config/wifi # Configuration file: ./hostapd.conf
rfkill: Cannot open RFKILL control device
[Sstar_log]:br0 netdev open,br_netdev[br0]
br0: port 1(p2p0) entered blocking state
br0: port 1(p2p0) entered forwarding state
[Sstar_log]:br0 netdev open()-l116: dev_get_by name(br0) failed2![Sstar_log]: __Sstar_tx[1]:deatuhen[ff:ff:ff:ff:ff:ff]
Using interface p2p0 with hwaddr de:29:19:06:b7:be and ssid "ssw101bap"
[Sstar_log]:Sstar_upload_beacon: change ds_params channel 4
[Sstar_log]:Sstar_set_priv_queue_cap:[1],queue_cap[224]
p2p0: interface state UNINITIALIZED->ENABLED
p2p0: AP-ENABLED
```

出现 **AP-ENABLED** 说明配置成功。