

Smart Radio Configuration Guide

Advanced MIMO Mesh Router for Resilient Private Wireless Networks

Introduction

The purpose of this guide is to aid in the software configuration of the Smart Radio. This guide is organized as follows.

1. [Smart Radio Settings \(For all models\)](#)
2. [Model Specific Configuration](#)
3. [System Configuration using the CLI](#)
4. [Additional Smart Radio Ports](#)
5. [Smart Radio Network Topologies](#)
6. [Centralized Configuration, ACS, and Link Recovery](#)
7. [Backup and Upgrade](#)
8. [Connect with Tech Support](#)
9. [References](#)
10. [Appendix A – Application Notes](#)
11. [Appendix B – Extending the Smart Radio Functionality](#)

Additional Guides

For a complete set of documentation, please refer to the following additional guides available from the [Tech Library](#) on our website. Listed below are some of the more popular guides which can aid in your setup.

1. Hardware Integration Guide (includes Mechanical drawings and CAD files)
2. Evaluation Kit guide
3. Datasheets
4. Regulatory Certification and Compliance Information
5. Trouble-Shooting guide

Smart Radio Settings (For all models)

Default Network Configuration

The Smart Radio runs Doodle Labs Mesh Rider® OS, a customized version of Openwrt with enhancements. These enhancements are useful for applications requiring low-latency command-and-control transmission, HD Video, high-throughput and long range. Fig. 1 shows the default network configuration of Smart Radio.

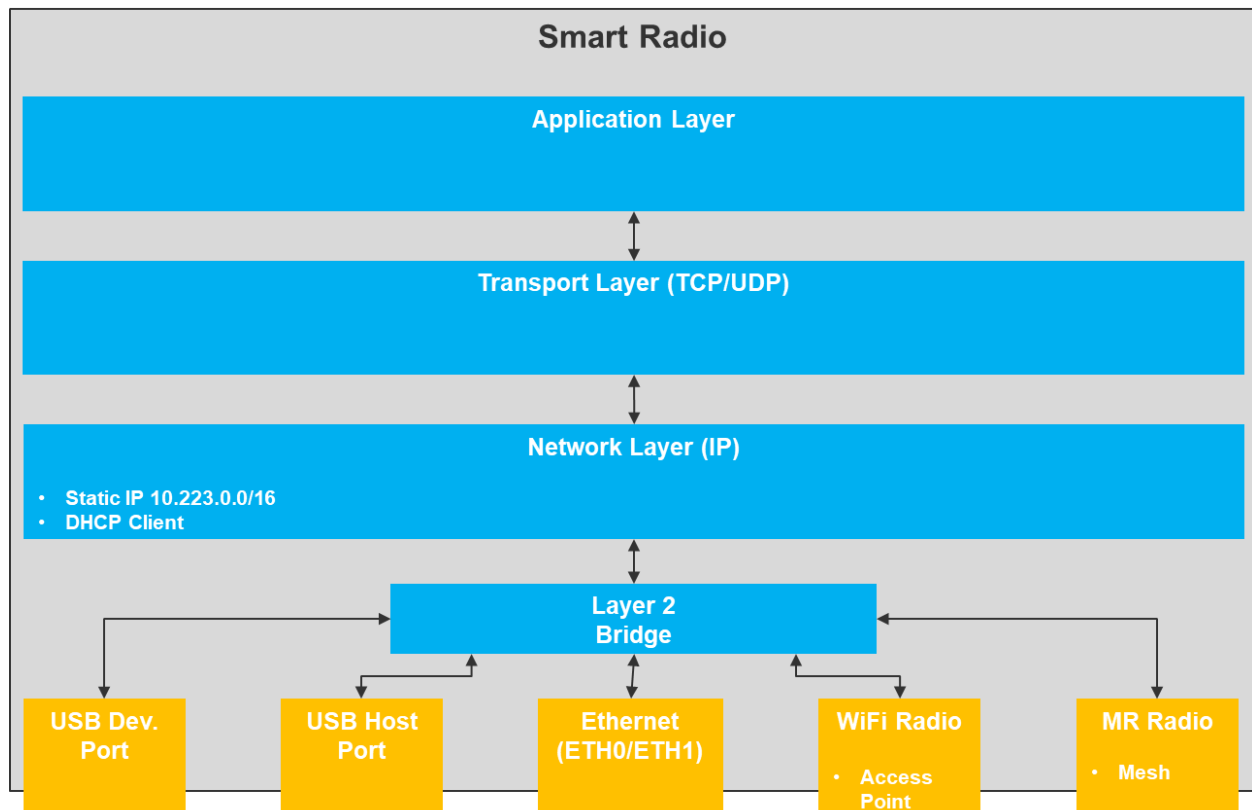


Fig. 1 Default Network Configuration

Smart Radio Interfaces:

Smart Radios are available in five different variants: Embedded, External, Wearable, OEM, and Helix. As each form factor is designed to cater to a particular use case, there are differences in the hardware interfaces between these designs. Table 1 shows the available interfaces for the different variants. ETH1 is configured by default as a configuration interface which is not remotely accessible, but it can be re-configured if necessary. Please note that the -H variant of

the Embedded Smart Radio is a legacy model. We recommend new designs to use the -J models. For the -H it comes with either a UART or a USB Host port.

Table 1: Available Interfaces for Smart Radios

Interface	Embedded	External	Wearable/OEM	Helix
Mesh Rider Radio	Yes	Yes	Yes	Yes
Ethernet (ETH0)	Yes	Yes	No	No
Ethernet (ETH1)	Yes	Yes	Yes	Yes
USB Device Port	No	No	Yes	Yes
USB Host Port	Yes	Yes	Yes	Yes
WiFi Radio	No	No	Yes	No
UART	Yes	Yes	Yes	Yes

Mesh Rider Interface

The Mesh Rider interface is the main wireless interface and is available on all Smart Radio products. Because it uses a proprietary wireless protocol, it is only accessible from other Mesh Rider devices.

ETH0 and ETH1 Interface

ETH0 and ETH1 are standard Ethernet interfaces and are bridged to the Mesh Rider Interface.

USB Device Port

The USB Device port is a Ethernet over USB interface and can be connected to USB host ports like those found on PCs, or USB OTG ports like those on Smart Devices. It is bridged to the Mesh Rider interface.

USB Host Port

The USB Host port is setup as an Ethernet over USB interface and can be connected to USB device ports or USB OTG ports like those found on Smart Devices. It is bridged to the Mesh Rider interface. Its functionality can be extended by installing additional USB drivers on the Smart Radio, like USB HID drivers.

WiFi Radio

The WiFi radio is bridged to the Mesh Rider Interface and provides standard WiFi connectivity to devices such as PCs and Smart Devices. The default SSID is DoodleLabsWiFi and the password is DoodleSmartRadio.

Configuration and Management Access

The Smart Radio can be managed by following protocols.

Configuration Guide

1. Web GUI
2. SSH
3. JSON-RPC API
4. SNMP

In this guide, we will use the SSH protocol which uses network port 22, and the web GUI which uses network port 443. The web GUI is accessible from standard browsers like Mozilla Firefox. An SSH client is built into most desktop Linux distributions, as well as Windows 10. You can use SSH for terminal access to the Smart Radio with the command,

```
user@host-pc:~$ ssh root@<IP ADDRESS>
```

Or you can remotely execute a command in the Smart Radio with

```
user@host-pc:~$ ssh root@<IP ADDRESS> '<COMMAND>'
```

You may need to update you PC's list of known hosts with

```
user@host-pc:~$ ssh-keygen -R root@<IP ADDRESS>>
```

The next section describes the IP configuration.

IP Addressing

There are three IP addresses defined by default.

1. A static IP address in the 10.223.0.0/16 subnet. This IP address differs from device to device and can be used for remote configuration or sending data to or from the Smart Radio. The IP address can be calculated from the MAC address. If the IP address is 10.223.x.y and the MAC address is 00:30:1A:4E:A4:3E, the x is equal to the decimal value of **0xA4**, or **164**, and y is equal to the decimal value of **0x3E**, or **62**. The full IP address is therefore 10.223.164.62/16. On newer devices, this IP address is printed on the label.
2. A DHCP client. This is a dynamic IP address which is automatically assigned when there is a DHCP server on the network.
3. A static IP address at 192.168.153.1/24. This is a fixed IP address which is the same for all radios. This IP address is designed to be used during initial configuration only and should never be used when more than one radio is on the network.

If you are unsure of the IP address on the 10.223.0.0/16 subnet, then the best option is:

1. First change the IP address of your host system to 192.168.153.10/24 (any valid IP address in the 192.168.153.0/24 subnet not equal to 192.168.153.1 will work). The method to do so varies depending on the OS and whether you are using the WiFi

Configuration Guide

interface, or the USB device interface. If you are connecting over WiFi for example, then you probably need to have your devices associated with SmartRadioWiFi before changing the IP address.

2. Next run a command on the Smart Radio over SSH to print out the IP address on the 10.223.0.0/16 subnet.

```
user@host-pc:~$ ssh -q -o stricthostkeychecking=no root@192.168.153.1 `ip  
addr | grep 10.223`  
inet 10.223.170.187/16 brd 10.223.255.255 scope global br-wan
```

Once you have the 10.223.0.0/16 IP address, you can change your host system's network configuration to use a static IP address on this subnet.

IP Advertisement

The radio runs an IP advertiser daemon which listens for UDP broadcast packets on port 11111 with the message "Hello" and replies with a list of the IP addresses on br-wan as well as the interface on which the message was received. This can be used to quickly identify all nodes on the network, as well as which one is local. To test it, run the following command on a Linux system with socat installed

```
user@host-pc:~$ echo "Hello" | socat - udp-  
datagram:10.223.255.255:11111,broadcast,sp=5000  
{ "ingress": ["bat0"], "IPaddr": ["10.223.187.2/16", "192.168.153.1/24", "192.168.0.  
.136/24", "fe80::230:1aff:fe4e:aa02/64"] }
```

The response is shown above.

Gateway Modes

There are two gateway modes defined in the simpleconfig menu, WDS AP with gateway, and Mesh with gateway. In these modes, a DHCP server runs on the Mesh Rider interface at 10.222.1.1/16, and the 192.168.153.1/24 fallback IP address is disabled.

Wireless Settings

The radio's Mesh Rider wireless settings can be found by navigating to `network` → `wireless` in the GUI. This is where you can configure the Mesh Rider wireless interface. The Wearable (and OEM) variants have an additional WiFi radio which is also configured here, and the Helix variant's band switching is also configured here. These are discussed in [the next section](#).

Mesh Rider Radio

To modify the wireless settings of the Mesh Rider radio, click `Edit` next to the relevant radio interface. You should see a page similar to the one below. We recommend the following updates.

Configuration Guide

- For point-to-point networks such as a Control Station controlling a single UAV, or Robot, **Enable** Dynamically Adjust txpower based on neighbor sounding (Transmit Power Control).
- At power up, the Smart Radio will scan the environment and choose the best channel for the environment. A different channel can be chosen manually.
- For 2.4-GHz ISM-band radios, use a channel bandwidth of 15-MHz to avoid normal WiFi interference. This is especially important in urban areas.
- Change the Mesh ID and under Wireless Security, change the password.
- Do NOT change the Mode of operation. This should be pre-selected in the Simple Configuration menu.

The screenshot shows the configuration page for a wireless network named "radio0.network1". The interface is divided into two main sections: Device Configuration and Interface Configuration.

Device Configuration:

- Status:** Shows "SSID: ? | Mode: Unknown" and "0% Wireless is disabled or not associated". There is an "Enable" button.
- Wireless network is disabled:** A toggle switch is currently turned off.
- Transmit Power Control (recommended for P2P connections only):** A checkbox is checked.
- Distance Optimization (114750 meters max):** A text input field contains the value "4000". A note below states: "Distance to farthest network member in meters, or 'auto' for dynamic acknowledgements."
- Operating frequency:** A table with three columns: Mode, Channel, and Bandwidth.

Mode	Channel	Bandwidth
N	51 (2250.000000 MHz)	10 MHz

 A note below states: "There are some channel and bandwidth combinations that are not possible. Please select desired bandwidth of operation, then click 'Save & Apply' button and finally select desired channel of operation."
- Country Code:** A dropdown menu is set to "GV - Government Domain". A note below states: "Use ISO/IEC 3166 alpha2 country codes."

Interface Configuration:

- Wireless Security:** This section is active and shows:
 - Encryption:** WPA2-PSK
 - Cipher:** Force 128 bit CCMP (AES)
 - Key:** A text input field with a green lock icon on the right.

Fig. 2 Wireless Settings

Network Interface Settings

Network interface settings can be modified in the `Network` → `Interfaces` tab. Below are some common changes you *may* want to make.

- Enable a DHCP server on *one* of the radios.

Configuration Guide

- Click **Edit** next to the **WAN** interface. You should see a page similar to the picture below. Change the **Protocol** from **DHCP** to **Static Address**. Add an IPv4 address and netmask of your choosing, then scroll to the bottom of the page, and un-check **Disable DHCP** for this Interface. Click **Save & Apply**.
- You may also wish to enable a DHCP server on the **LAN** interface (**ETH1**). The steps are similar to those above.
- Change the default static IP address of **ETH0**. To do this, edit the **WAN2** interface, and change the static IP address to your liking.

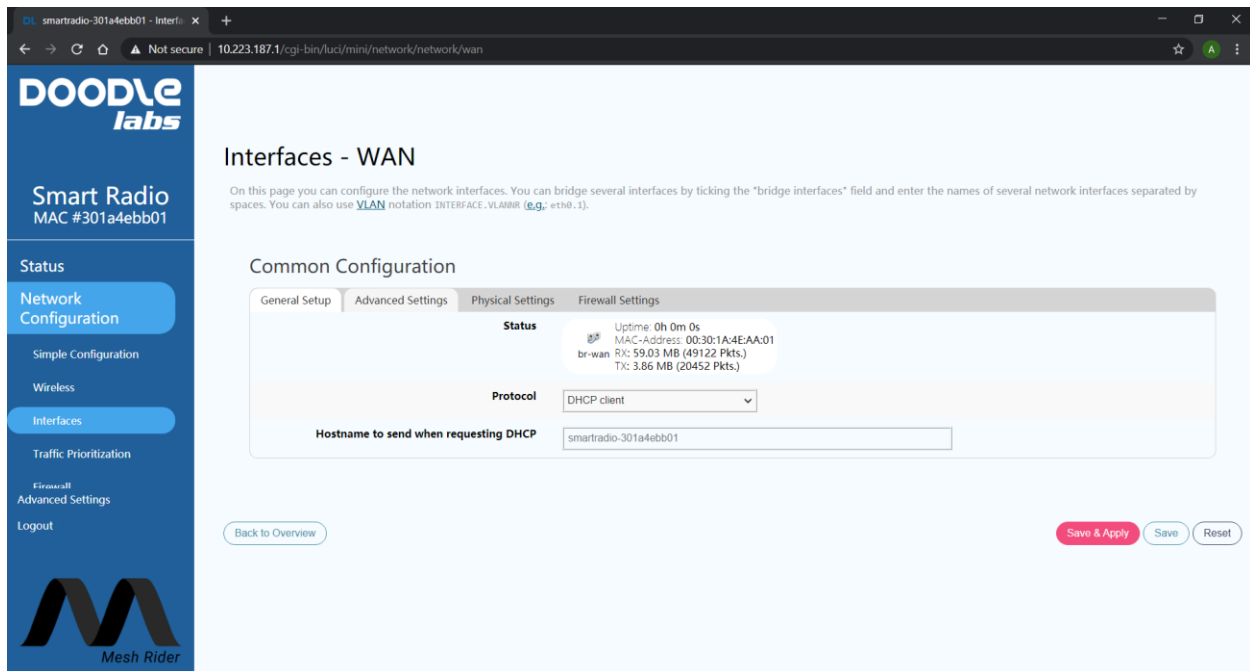


Fig. 3 Network Interface Settings

Mesh Settings

The default radio network configuration is a mesh, and the configuration settings can be accessed at **Network** → **Mesh Configuration**. However, this menu is only accessible in the **Advanced Settings** which is opened by clicking the **Advanced Settings** button in the bottom of the left hand side menu bar. Fig. 4 shows the Mesh Configuration Page. In general, you should not need to change most of the settings except for:

- **OGM Broadcasting Interval** – This number affects the responsiveness of the mesh to routing changes. A smaller number is faster, but results in more network overhead. For large meshes, it is recommended to increase this.

- Bridge Loop Avoidance – This is only necessary if there are potential network loops created in the Ethernet backbone. For example, if two of the wireless mesh nodes are also connected over the wired backbone, a loop is formed, and Bridge Loop Avoidance should be enabled.
- Group-Aware Multicast to Unicast Conversion – This option is necessary if your application uses the IGMP/MLD protocol for multicasting. It eliminates redundant data distribution and converts the multicast transmission to unicast.
- Fast re-routing – This experimental feature makes the mesh algorithm more responsive to changes in the optimal packet routing. This feature should not be used if long range broadcasting is required. i.e. using the broadcast IP address (e.g. 10.223.255.255) for long-range.

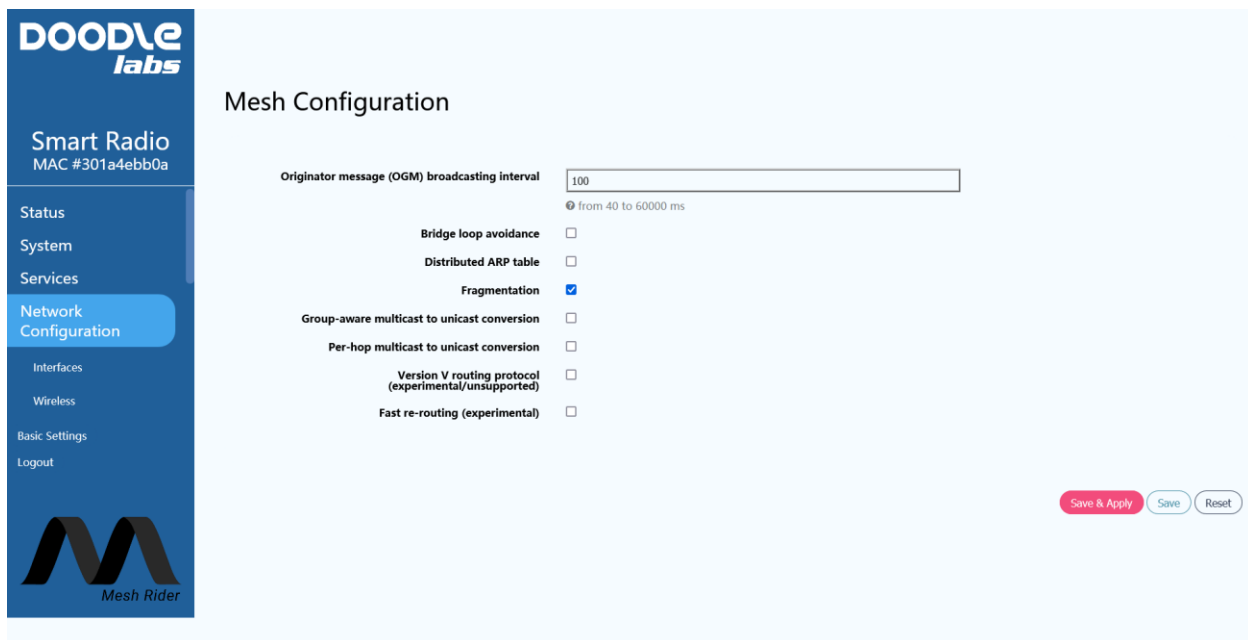


Fig. 4 Mesh Configuration Page

Traffic Prioritization

Different types of traffic can be prioritized in the Traffic Prioritization menu. This is useful when operating in a crowded wireless medium. There are four different queues – Voice, Video, Best Effort, and Background. The Voice queue optimizes latency and may also be used for command and control, the Video queue optimizes throughput, the Best Effort Queue is essentially unoptimized, and the Background queue is for low-priority data.

Configuration Guide

In order to use these QoS features, open up the web GUI and navigate to `network` → `differentiated services`. The Smart Radio includes software to map different network protocols or ports to the various QoS queues. To do so, click `Enable Differentiated Services`, and add a classification rule to suit the application's needs. For example, you can send all UDP traffic to the Video queue which is beneficial for video transmission.

URLLC (Ultra Reliable Low Latency Channel) and Video Optimization

The Smart Radio includes protocol optimizations for URLLC applications as well as video optimizations. URLLC applications typically include command and control (C&C) data, but can be extended to any application requiring a reliable low latency. Assuming that we have a C&C application which uses network port 7000 over UDP. In the screenshot below, first click `Optimize Command & Control for URLLC`. Next click `Add`, and then change the new classification rule to use `Port 7000`, and set the `DSCP` value to `CS6`. The comment section can be filled if desired. Finally click `Save & Apply` and wait for the page to refresh.

Smart Radio
MAC #00301a4ebb02

Network Configuration

- Simple Configuration
- Wireless
- Interfaces
- Traffic Prioritization**
- Firewall
- Services
- Advanced Settings
- Logout

Differentiated Services

General Settings

- Enable Differentiated Services
- Optimize Command & Control and Voice for URLLC
- Optimize Video Streaming
- Video bad link threshold (dBm):
- Video bad link drop (percentage):
- Diversity Rates Only

Classification Rules

Source host	Destination host	Protocol	Port(s)	DSCP	Comment	Sort
all	all	UDP	2000	Voice, Command & Control (CS6)	socat raw	↑ ↓ Delete
all	all	UDP	14550	Voice, Command & Control (CS6)	socat raw	↑ ↓ Delete

Fig. 5 Traffic Prioritization Settings

If `Optimize Video Streaming` is enabled, the radio will

1. Apply radio PHY settings which are optimized for video transmission

Configuration Guide

- If the RSSI to a particular station is below the Video bad link threshold, then the radio will drop Video bad link (percentage) packets. This is an optional feature, and you should adjust the RSSI to a reasonable value for your application.

The RSSI bad-link threshold is an additional protection against network overload when the link quality is bad, but even without setting the RSSI bad-link threshold, the C&C queue is prioritized over the VI queue.

The Diversity Rates Only checkbox forces the radio to send the same data over both antennas redundantly. This can lead to smoother performance in highly dynamically changing conditions.

Firewall Settings

The Firewall configuration is located under Network → Firewall and can be modified over the GUI and CLI at `/etc/config/firewall`.

We shall use the iperf3 application as an example to demonstrate setting a firewall rule to allow access to port 5201 of the router. Navigate to the Firewall page, and click the Traffic Rules tab. In the “Open ports on router” section, enter the name, protocol, and port number as “Allow iperf”, “TCP”, and “5201” respectively so that clients is able to connect to port 5201.

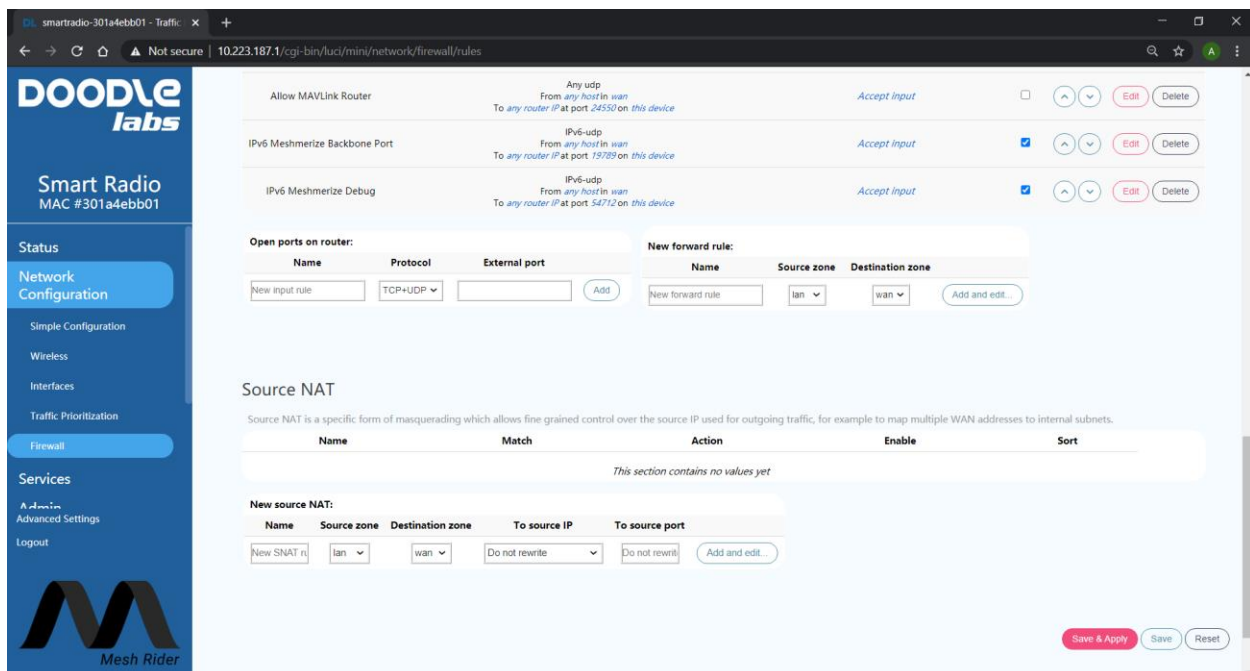


Fig. 6 Firewall Settings

Configuration Guide

Before port 5201 is opened, run iperf in server mode inside the Smart Radio.

```
root@smartradio-301a4e8646:~# iperf3 -s
```

If you try and connect to the iperf server from your local machine, you will get an error, `connect failed: Connection refused`. After opening the firewall at port 5201, you should be able to connect to the iperf3 server.

Extensive information regarding Firewall configuration is available at the [openwrt.org website](https://openwrt.org).

Advanced Settings

By default, the web GUI only presents a small subset of all of the possible configuration options of the Smart Radio. For advanced users, you may wish to reveal all configuration menus by clicking the Advanced Settings button at the bottom left hand corner of the page.

Model Specific Configuration

The Wearable (and OEM) and Helix model variants have addition configuration options compared to the defaults discussed in the section above.

Wearable/OEM Variant (-K)

The Wearable/OEM variant of the Smart Radio includes a WiFi radio along with some additional features to make it ideal as a hand-held or wearable unit.

WiFi Radio (Wearable/OEM only)

Fig. 7 shows the Wireless Configuration page for the Wearable/OEM models. Radio 0 (above) refers to the WiFi radio and Radio 1 (below) refers to the Mesh Rider radio.

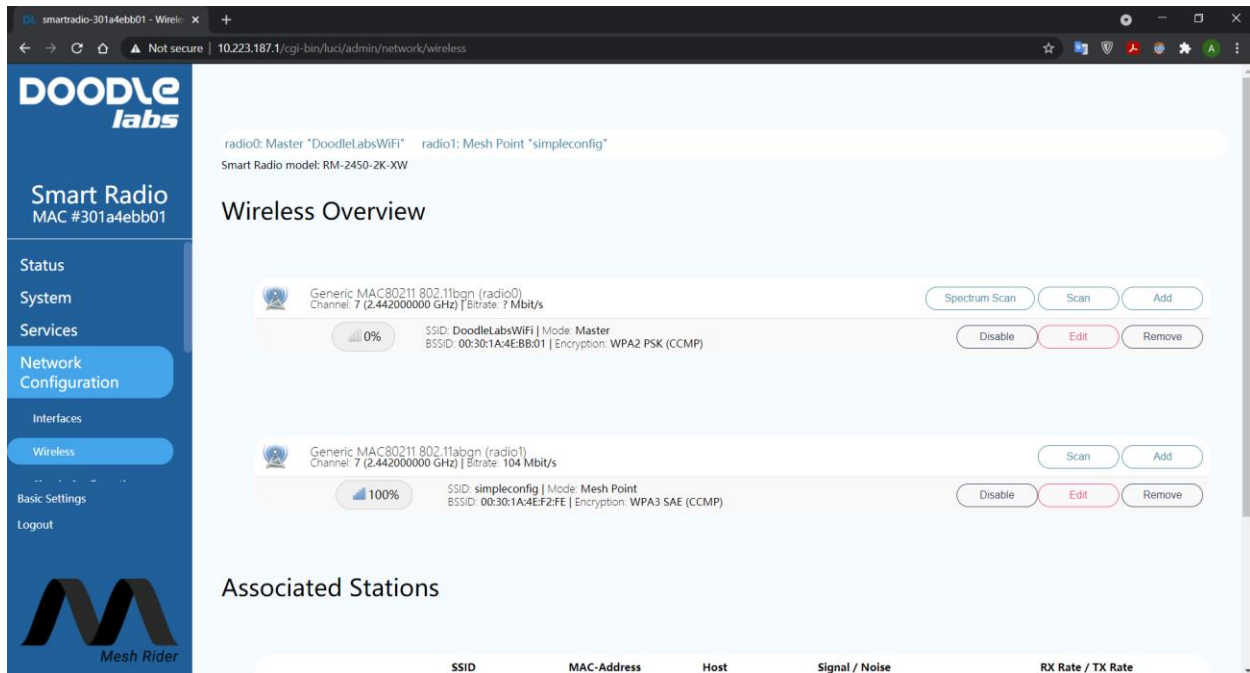


Fig. 7 Wireless Configuration Page

The WiFi radio is set up as an AP by default and is bridged to the Mesh. The WiFi radio can also be configured to connect to a hotspot and in this case, it will do the following

1. Start a DHCP server on the Mesh Rider network.
2. Enable DHCP client mode on the WiFi radio and join the hotspot’s network.
3. Perform network address translation (NAT) between the Mesh Rider network and the hotspot’s network.

Note that because NAT is used between the hotspot’s network and the Mesh Rider network, the IP address range in the Mesh Rider network is not accessible (by default) from the hotspot’s network. This means that while Mesh Rider devices will be able to access nodes on the hotspot network such as the internet, devices in the hotspot network will not be able to talk to applications in the Mesh Rider network unless port forwarding is enabled for those specific applications. This is similar to how an office WiFi network works. In order to configure the WiFi radio as a client device, click the `Scan` button next to Radio 0. The radio will scan for available networks and create a list like the one shown in Fig. 8. Click the button `Hotspot assistant` next to the network you wish to join and follow the on-screen instructions.

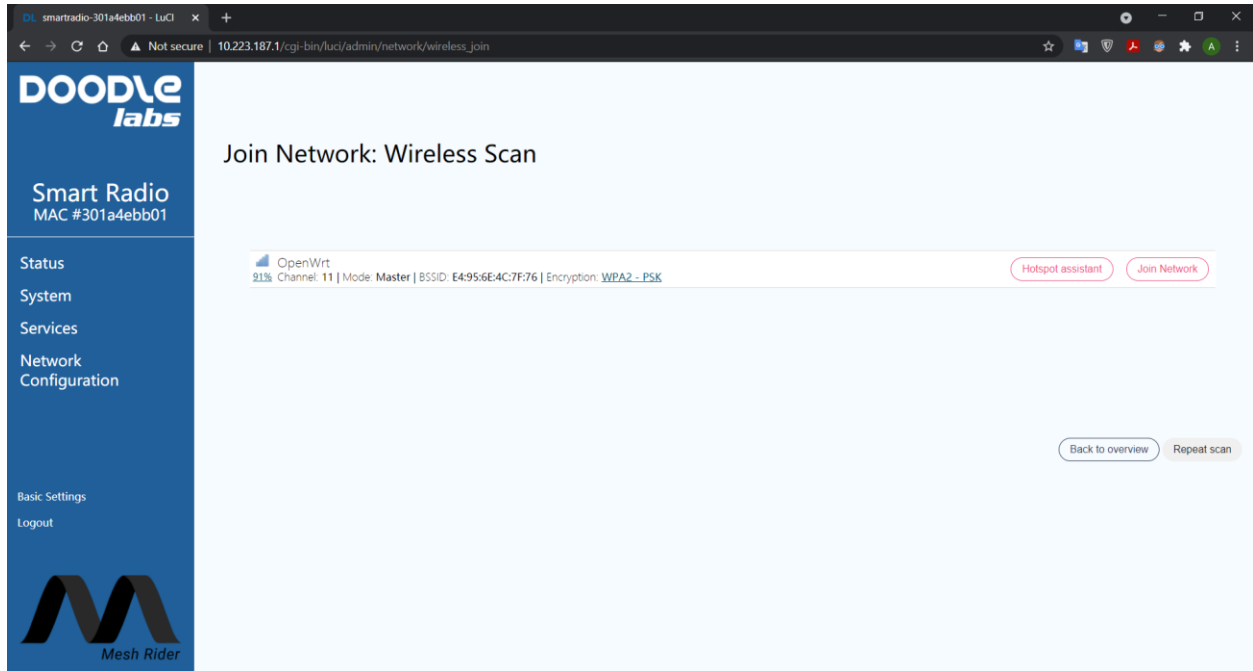


Fig. 8 Scanning for available WiFi networks

Wearable Configuration

The Wearable Smart Radio has an additional configuration page at `System` → `Wearable Smart Radio`. This menu is only visible when the Advanced Settings are enabled. The Wearable Smart Radio includes GPS, altitude, battery voltage and temperature sensors which are reported on this configuration page. The user may choose to enable standby mode, or the temperature control fan, or switch to external GPS and Mesh Rider antennas.

The screenshot shows the 'Wearable Smart Radio' configuration page. On the left is a dark blue sidebar with the 'DOODLE labs' logo at the top, followed by 'Smart Radio' and 'MAC #93001e706f'. Below this are menu items: 'Status', 'System', 'Wearable Smart Radio' (highlighted), 'System', 'Administration', 'Software', 'Startup', 'Scheduled Tasks', 'GPS receiver', 'Basic Settings', and 'Logout'. At the bottom of the sidebar is a 'Mesh Rider' logo. The main content area is light blue and titled 'Wearable Smart Radio'. It is divided into two sections: 'Status' and 'Configuration'. The 'Status' section displays: 'GPS location' (Unknown), 'Battery voltage (Volts)' (8.91), 'Temperature (Celsius)' (28), and 'Altitude (Meters)' (25). The 'Configuration' section includes: 'GPS antenna' (Internal), 'Mesh Rider antennas' (Internal), 'Temperature control fan' (On) with a note 'The fan will turn on when the radio temperature exceeds 45 degrees Celsius.', 'Auto standby' (Enabled) with a note 'Automatically go to standby when network activity is low, and wake up periodically.', 'Wakeup duration (Minutes)' (2), and 'Sleep duration (Minutes)' (10). At the bottom right of the configuration area are three buttons: 'Save & Apply' (red), 'Save' (light blue), and 'Reset' (light blue).

Fig. 9 Wearable Smart Radio Configuration

Helix Variant (-L)

Helix radios have the option to select the operating band. Choose the desired operating band in the drop-down box at the top of the page and click “Change”.

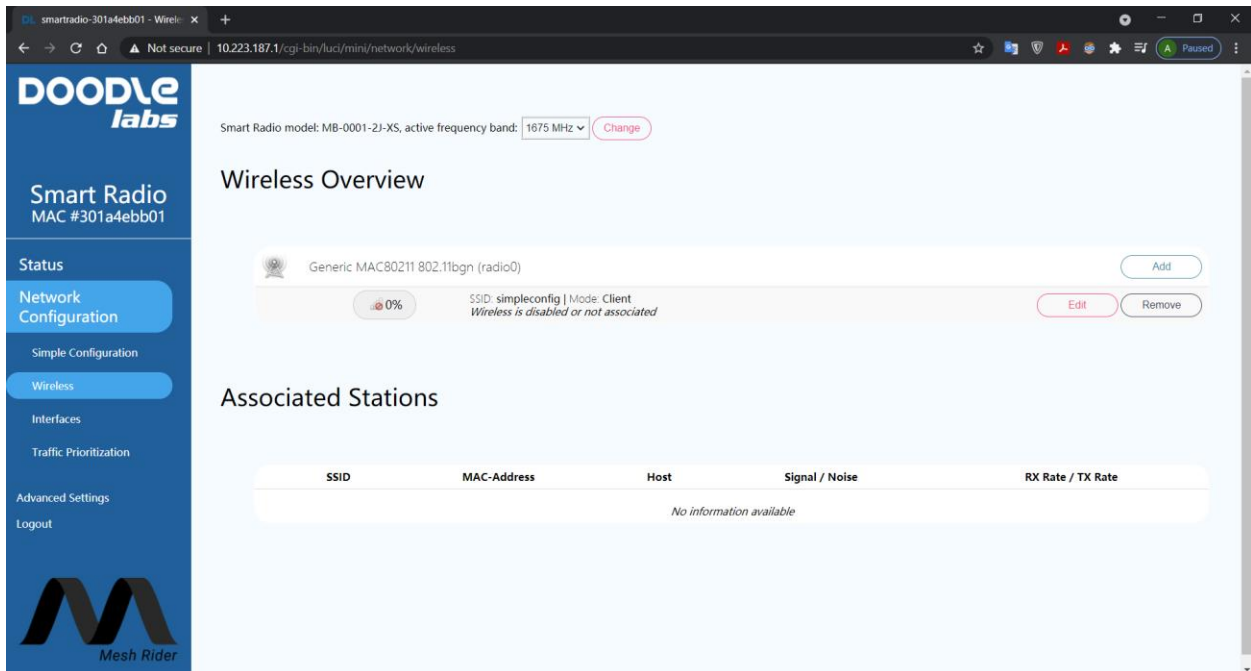


Fig. 10 Wireless band selection for Helix models

System Configuration using the CLI

The Smart Radio runs the Linux kernel, and many Linux utilities such as `ifconfig`, `iw`, and `ip` are pre-installed on the radio. Generally speaking we can divide CLI tasks that a user may wish to perform into persistent configuration changes, and on-the-fly changes or queries.

In this section, we will first discuss the UCI system, which is designed for persistent configuration changes, and afterwards we will discuss how to configure or query the device on the fly. You can SSH into the radio using these commands.

UCI Overview

Openwrt uses the UCI system for device configuration. The user is encouraged to explore the [OpenWrt documentation](#). Configuring the radio using the UCI system is a slow process especially for the network interface. Configuration changes survive a reboot, but the network interfaces may need to be restarted for changes to take effect, and this could take 20-30 seconds depending on the changes being made. Most of the configuration files can be found [here](#),

```
root@smartradio:~# ls /etc/config/
batman-adv      diffserv      firewall      network
openvpn_recipes system        ucitrack     wireless
dhcp           dropbear     luci         openvpn
rpcd           ubootenv     uhttpd
```

To show the current wireless configuration, type

```
root@smartradio:~# uci show wireless
wireless.radio0=wifi-device
wireless.radio0.type='mac80211'
wireless.radio0.path='platform/qca953x_wmac'
wireless.radio0.htmode='HT20'
```

Only the first four results are shown above. You can do this for any of the files listed. To change a setting, e.g. the wireless channel, use `uci set`:

```
root@smartradio:~# uci set wireless.radio0.channel='10'
```

Changes to the configuration can be stored using

```
root@smartradio:~# uci commit
```

This change will be held in memory but will have an effect when the network settings are reloaded using

```
root@smartradio:~# /etc/init.d/network reload
```


Configuration Guide

You can list the services which can be controlled by running

```
root@smartradio:~# ls /etc/init.d/
```

Common control commands are `start`, `stop`, `restart`, `reload`, `enable`, and `disable`.

On the Fly Querying and Configuration

The UCI system is the preferred system for persistent configuration of the Smart Radio. Standard Linux tools can be used to make configuration changes in real-time, or query the system state. We will only cover the basics.

1. Getting help – Most Linux utilities have a help menu which can be access with a `--help` argument. The help menu typically provides list of commands and options. For example,

```
root@smartradio:~# iwinfo --help
Usage:
    iwinfo <device> info
    iwinfo <device> scan
    iwinfo <device> txpowerlist
    iwinfo <device> freqlist
    iwinfo <device> assoclist
    iwinfo <device> countrylist
    iwinfo <device> htmodelist
    iwinfo <backend> phyname <section>
```

2. There are various utilities installed which can be used to filter the output like `sed`, `awk`, `grep`, and `tr`. There is plenty of online resources on each of them. An example would be

```
root@smartradio:~# iw wlan0 station get 00:30:1a:4e:f3:00 | grep "signal: " |
awk '{print $1" "$2" "$5}' | sed 's/signal:/RSSI=/g'
RSSI= -66 dBm
```

3. Below are some commonly used utilities. Try running `--help` on each of them, or try running the example command.
 - a. `ifconfig` – configure/query networking information like IP addresses. Example:
 - i. `ifconfig br-wan`
 - b. `ip` – another tool to configure/query networking information. Example:
 - i. `ip a show dev br-wan`
 - c. `iw` – configure/query wireless interfaces. Examples:
 - i. `iw wlan0 set bitrates ht-mcs-2.4 0 #sets the bitrate to MCS0`
 - ii. `iw wlan0 set bitrates ht-mcs-2.4 0 #resets the bitrate to auto`
 - iii. `iw wlan0 station dump #get connection information`

- iv. `iw wlan0 set txpower fixed 2000 #Fixes the output power to 20 dBm (measured in millibels, not decibels)`
 - v. `iw wlan0 set txpower auto`
 - d. `iwinfo` – another tool for querying/configuring wireless interfaces. Example:
 - i. `iwinfo wlan0 assoclist`
 - e. `dmesg` – see kernel messages. No help menu.
 - f. `batctl` – utility for querying/configuring mesh information. Example:
 - i. `batctl o #see nodes participating in the mesh, last seen time, the best route, and the transmit quality for each route`

Additional Smart Radio Ports

UART

The UART port is a standard TTL level 3-pin serial port (RX, TX, GND). The UART port can be used directly with flight controllers such as the PixHawk 4. It is also possible to setup a network to serial relay, or a direct serial to serial link over the network. Please refer to Doodle Labs' Tech Library for an Application Note on using the serial port for further details.

USB Device Port

The USB Device Port only supports Ethernet over USB protocols. It works with Laptops and Tablets as long as they support Ethernet over USB. The USB Device Port uses the ETH0 interface logically.

USB Host Port

The USB port is a host port and is pre-configured as an Ethernet over USB interface. When plugged into a USB device or OTG port with a compatible Ethernet over USB configuration, a new interface, USB0, will be instantiated and bridged to the WAN interface. It is then possible to access the Smart Radio over the web browser or SSH at the default static IP address defined earlier. It is possible to install other USB packages to include other USB functions for your application.

The Smart Radio has a USB hub inside and USB ports are accessible on the main and secondary connectors. USB0 on the main connector does not have a 5-V supply associated with it.

GPIO

In hardware version J and later, GPIOs are available for programming. The GPIOs are provided by an on-board MCU and interfaced to the Operating System over USB HID. A program "sr-ctrl-usb" is pre-installed in the system and is used to access the GPIOs. The general syntax for accessing the MCU is

```
root@smartradio:~# sr-ctrl-usb <r/w> <param #> <value>
```

r/w is either "0" for read or "1" for write. The parameter numbers for the three GPIOs are tabulated below. The possible values which can be read or written are "0" and "1" and correspond to 0-V and 3.3-V. Table 2 summarizes GPIO parameters.

Table 2: GPIO Parameters

Param #	Function
6	Logic level of GPIO1, 0 is GND and 1 is 3.3-V
7	Logic level of GPIO2, 0 is GND and 1 is 3.3-V
8	Logic level of GPIO3, 0 is GND and 1 is 3.3-V
11	Direction of GPIO1, 0 is Output and 1 is Input
12	Direction of GPIO2, 0 is Output and 1 is Input
13	Direction of GPIO3, 0 is Output and 1 is Input

As an example, if we want to write a level 1 to GPIO2, we first set the direction of GPIO2 to 0 and then we set the logic level to 1.

```
root@smartradio:~# sr-ctrl-usb 1 12 0
root@smartradio:~# sr-ctrl-usb 1 7 1
```

We can read the logic level of GPIO3 with,

```
root@smartradio:~# sr-ctrl-usb 1 13 1
root@smartradio:~# sr-ctrl-usb 0 8
```

Note that the GPIO directions are reset to input when the MCU is reset. When the MCU is reset, a reset flag is raised at param # 13 and it should be reset to zero by the user.

Wake-Up Timer

For very low power monitoring applications, the Smart Radio can be configured to completely shut down and wake up on a timer. In order to conserve power, the CPU itself shuts down and it requires around 30 seconds to boot up again. The command to shutdown the system is,

```
root@smartradio:~# sr-ctrl-usb 1 2 <time in seconds>
```

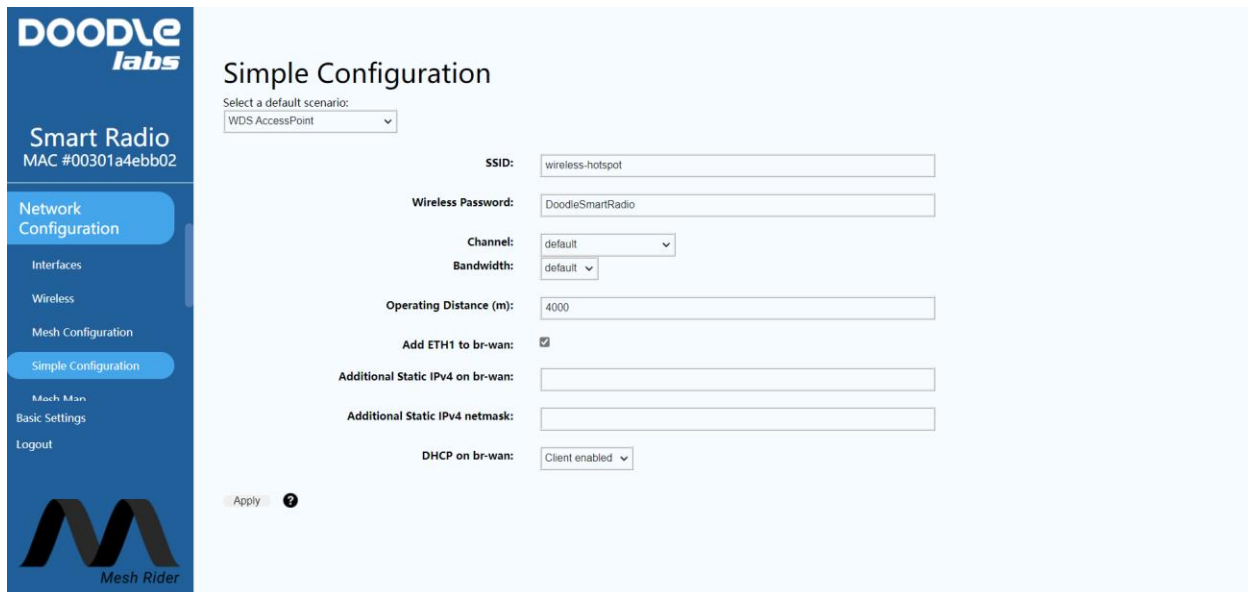
where <time in seconds> is the amount of time that the CPU should shut down for. This feature is only available in -J hardware variants.

Smart Radio Network Topologies

The Smart Radio supports several different network configurations. The Simple Config menu makes it easy to configure the radio. The Simple Config menu is accessed in the web GUI at network → simple config.

Basic Network Configurations

The Simple Configuration menu is shown in Fig. 11. It should be the starting point of any network design. From here, you should first choose the mode which you want to switch to, and a selection of the more common configuration options, and then click `Apply`.



The screenshot displays the 'Simple Configuration' menu in the Doodle Labs web GUI. The left sidebar contains navigation options: Network Configuration, Interfaces, Wireless, Mesh Configuration, Simple Configuration (highlighted), Mesh Man, Basic Settings, and Logout. The main content area is titled 'Simple Configuration' and includes a dropdown for 'Select a default scenario' (set to 'WDS AccessPoint'). The configuration fields are as follows:

- SSID: wireless-hotspot
- Wireless Password: DoodleSmartRadio
- Channel: default
- Bandwidth: default
- Operating Distance (m): 4000
- Add ETH1 to br-wan:
- Additional Static IPv4 on br-wan: [empty field]
- Additional Static IPv4 netmask: [empty field]
- DHCP on br-wan: Client enabled

An 'Apply' button with a help icon is located at the bottom left of the configuration area.

Fig. 11 Simple Configuration Menu

These are the support network configurations.

1. Mesh
2. Multi-hop relay for range extension
3. Muti-hop relay for NLOS conditions
4. WDS AP/Client
5. Dynamic Mesh

The basic network configuration in any of these modes is designed so that the entire Smart Radio network acts like one big Ethernet switch. Any host machines plugged into this “Ethernet switch” will be able to communicate with one another as long as they are on the same IP subnet, regardless of the IP addresses of the Smart Radios. In order to access the Smart Radios

Configuration Guide

themselves, however, it is necessary for the host machines to be on the same IP subnet as the Smart Radios.

Mesh

All Smart Radios are setup in Mesh mode by default. In mesh mode, all nodes are able to communicate directly with each other and data can even hop between nodes to reach a destination. This is illustrated in Fig. 12. Maintaining the mesh requires additional network overhead compared to WDS AP/Client mode. Some benefits of Mesh mode compared to other modes include

- The network is homogenous and there is no need to configure each node differently
- There is no single point of failure
- The network can be extended beyond the line of sight through intelligent packet routing

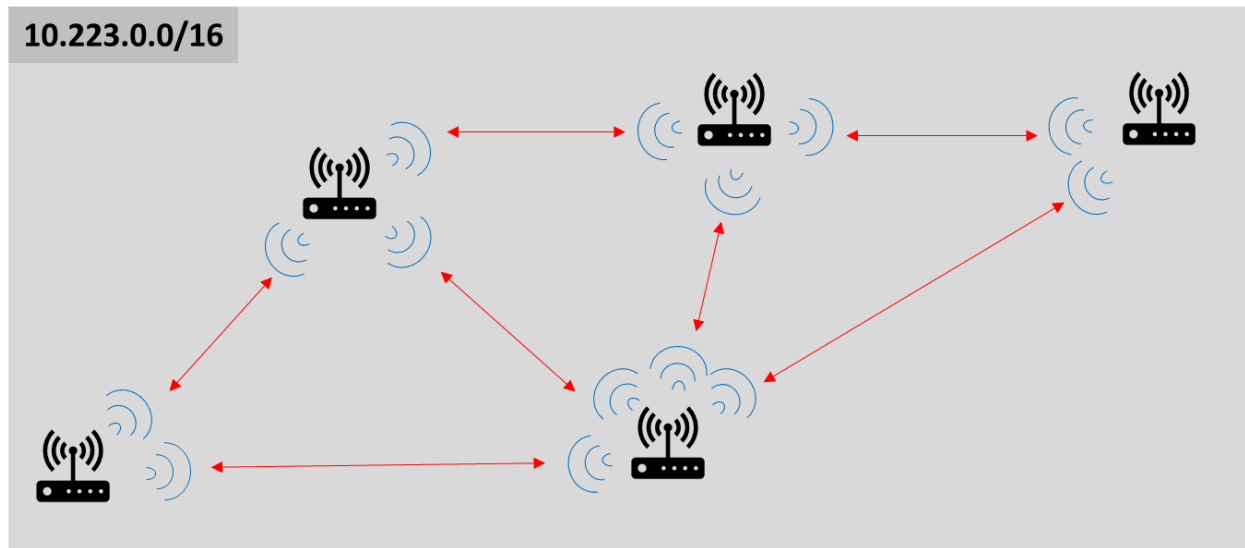


Fig. 12 – Mesh mode

WDS AP/Client

WDS AP/Client mode is a star network configuration. All data must go through the AP. If meshing is not required, this is the preferred network mode as it is more responsive than Mesh mode.

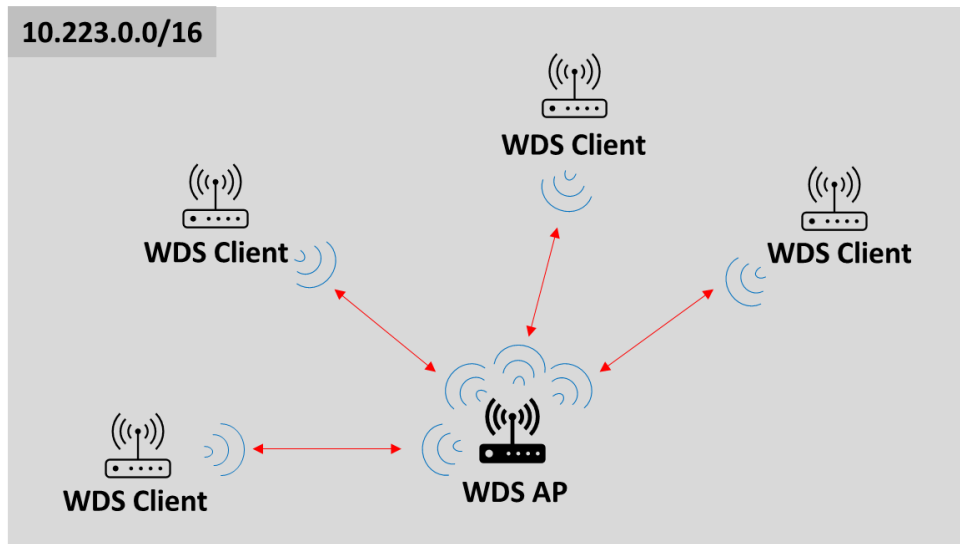


Fig. 13 – WDS AP/Client mode

Some benefits of WDS AP/Client mode compared to the others

- It provides the highest throughput
- It automatically searches for the best channel on start-up
- Client devices automatically scan and reconnect to the AP if the AP switches channels

Dynamic Mesh

Dynamic Mesh is a state-of-the-art network protocol based on establishing multiple redundant paths in a mesh network. Dynamic Mesh has ZERO packet re-routing or failover time. Dynamic Mesh also supports dual radio failover where two back-to-back radios on the same or different frequency bands route packets redundantly between nodes (Fig. 14). This makes Dynamic Mesh best-in-class in terms of latency and resiliency in highly dynamically changing environments.

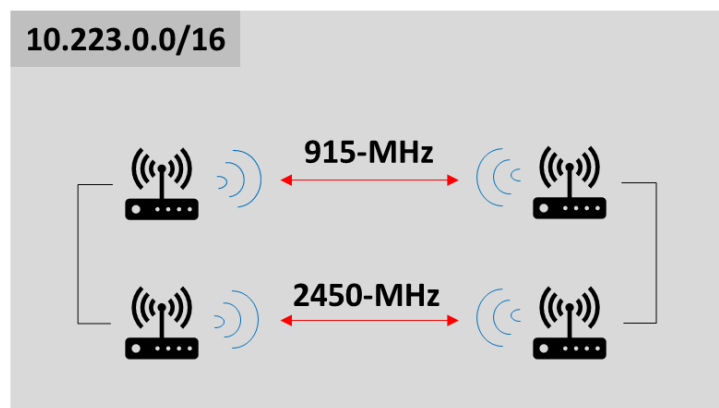


Fig. 14 – Dual Radio Dynamic Mesh

Configuration Guide

Some benefits of Dynamic Mesh mode compared to other modes include

- ZERO failover or packet re-routing time
- No single point of failure
- Can support very large number of nodes in a single network

Extended Modes

Aside from the basic network configurations above, the Smart Radio also supports

1. Gateway mode
2. Multi-Radio Mesh mode
3. Mesh + AP mode
4. Wired Backbone

Gateway Mode

Gateway mode is supported in both WDS AP/Client and Mesh modes. In Gateway mode, one node is elected as a gateway to a wider network (the internet for example). The main data interface is connected to your modem, and performs NAT between the Smart Radio's LAN and the wider network. Additionally, the gateway Smart Radio starts a DHCP server on the LAN on the 10.222.0.0/16 subnet and a gateway IP address of 10.222.1.1. From firmware version 2022-02 onwards, ETH1 runs the DHCP client which connects to your modem, and ETH0 is bridged to the local network. Fig. 15 illustrates gateway mode for a mesh configuration.

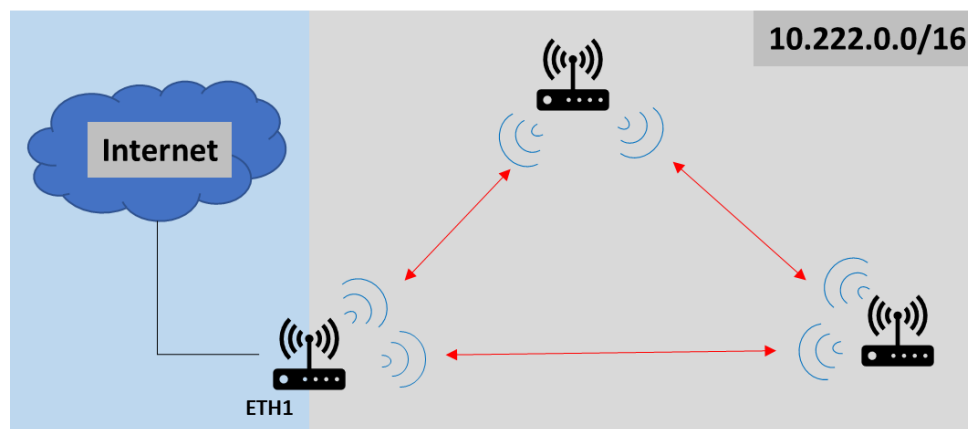


Fig. 15 – Gateway mode

Multi-Radio Mesh mode

Multi-Radio Mesh mode is specifically designed to extend the range of a network without sacrificing throughput. Looking at the diagram below, if both sets of radios operated in the

Configuration Guide

same band, then they would need to share airtime. In Multi-Radio Mesh mode, radios are connected to each other over wire using the ETH1 interface.

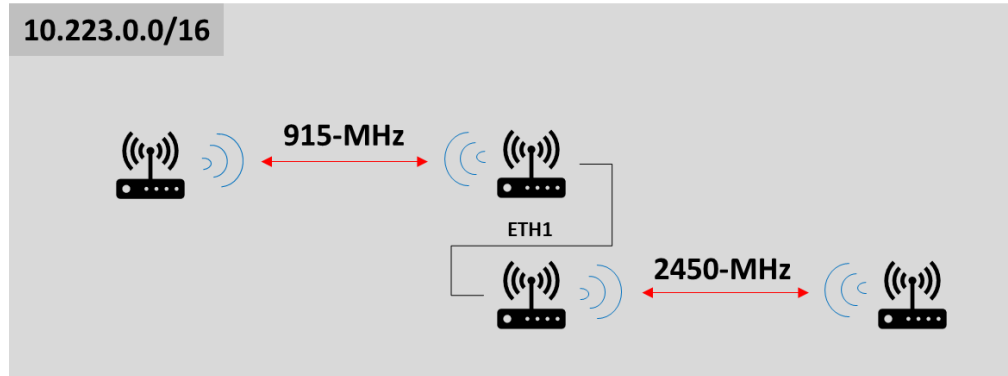


Fig. 16 – Multi-Radio Mesh

Mesh + AP mode

Mesh + AP mode is illustrated in Fig. 17. Two or more units form a mesh backbone and also run an AP which is bridged to the mesh. Of note is that this is standard AP mode and not WDS AP mode. The APs are interoperable with standard WiFi clients for 2.4-GHz models if the channel bandwidth is set to 20-MHz.

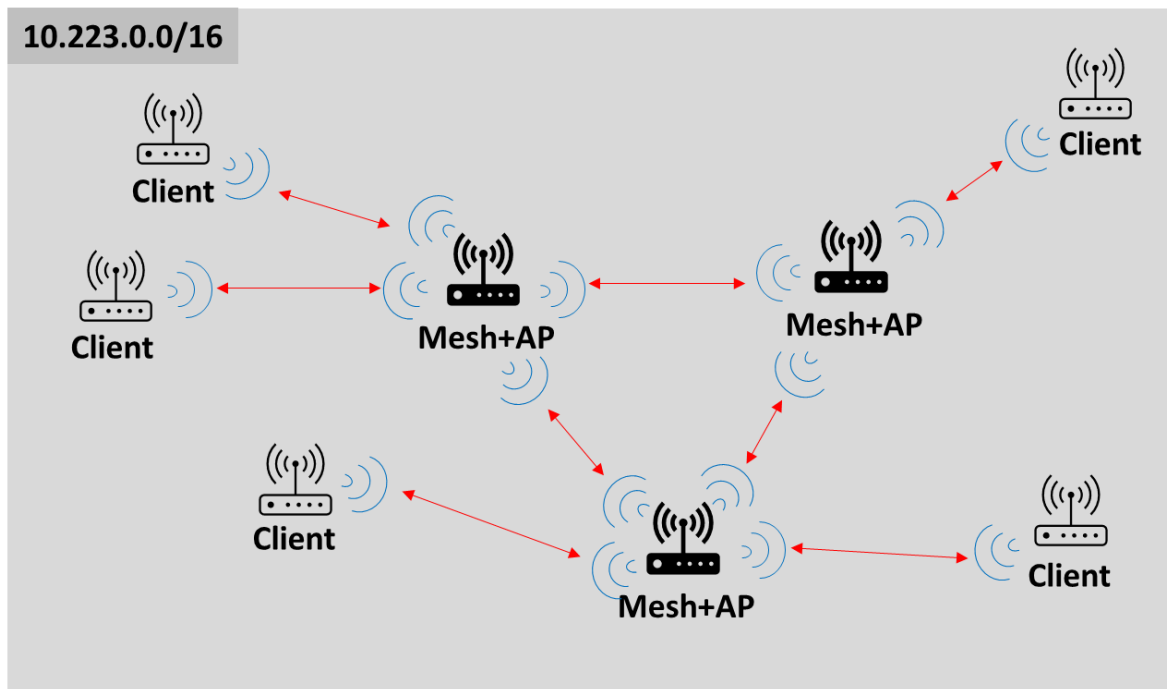


Fig. 17 – Mesh + AP mode

Wired Backbone

Although not a selectable mode, the Smart Radio supports extension of the network with a wired backbone in all three basic modes. In Mesh mode, it is necessary to enable “Bridge Loop Avoidance” when using a wired backbone. In WDS AP/Mode, no additional configuration is required, and clients can roam between WDS APs even with the APs on different operating channels. Typical handover duration is 0.2 seconds for APs in the same channel and 0.6 seconds for APs in different channels. Dynamic Mesh also supports use of a wired backbone with zero failover time.

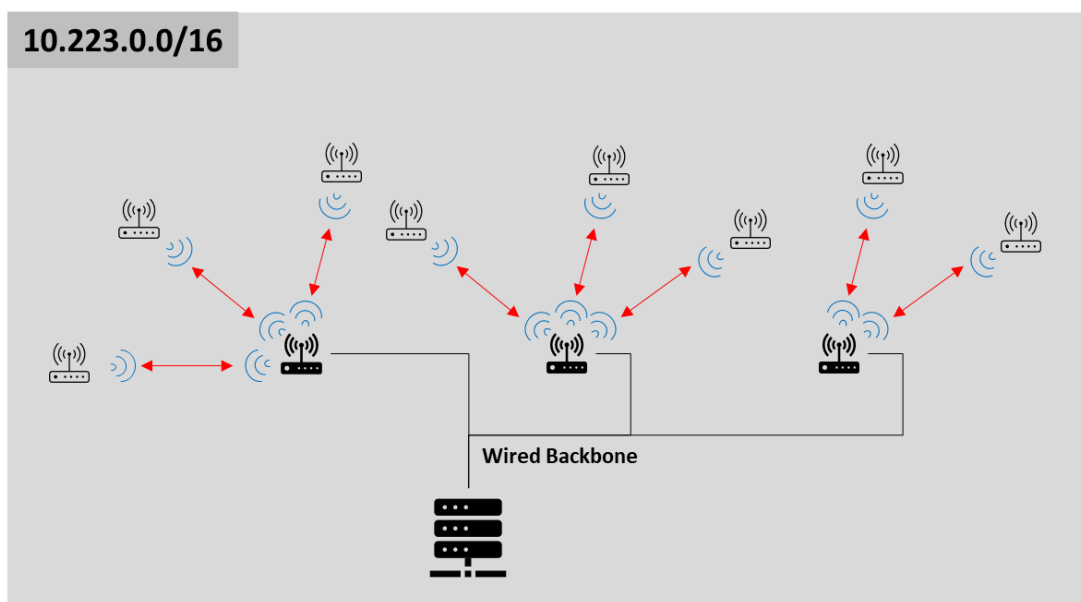


Fig. 18 – Wired Backbone

Use Case Examples

As can be seen, the Smart Radio is very flexible and provides many software configurable operating modes. This allows it to be used in many different use cases. Please refer to the [Technical Library](#) section of the website for application notes on these use cases.

Centralized Configuration, ACS, and Link Recovery

With the release of the 2022-02 update of the Mesh Rider OS comes a new Centralized Configuration utility along with automatic channel selection (ACS), and link recovery. All three features use the Central Configuration utility as a basis for operation, therefore Central Config must be set up properly before ACS and Link Recovery will work.

Central Config

In order to enable Central Configuration, you need to choose one node as the primary node. Navigate to Services → Central Config (Fig. 19).

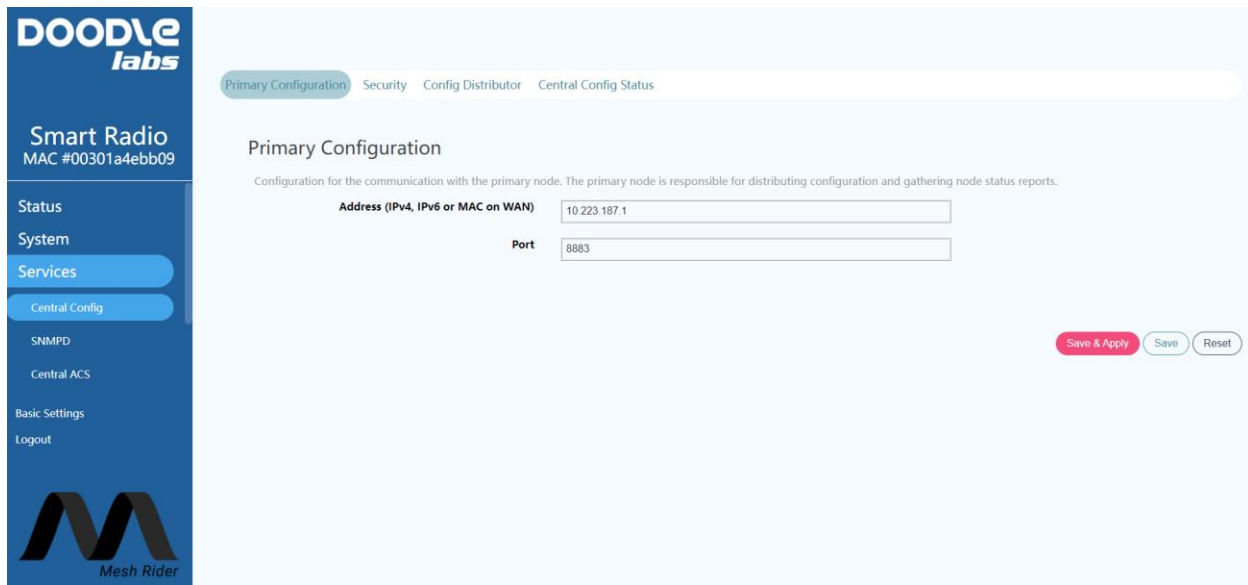


Fig. 19 – Central Config Menu

The primary node is selected by inputting its address (IPv4, IPv6, or MAC) in the Primary Configuration tab and clicking *Save & Apply*. The Central Configuration utility uses TLS for security, and this can be configured in the *Security* tab. From the Config Distributor menu, you can update the operating Channel, Distance setting, or TX Power. Additional features are planned in future. It is also possible to send central configuration commands through the CLI. Information can be found in the updated “Remote Management Guide for Smart Radio” available on the Doodle Labs website [1].

In the Central Config Status tab, we can see the status of the other nodes in the network. In Mesh modes, TQ (transmit quality) as a percentage. This is the overall TQ from the local node to all other nodes. In AP/Client modes, the RSSI is shown in dBm. The last seen time in

minutes:seconds is also shown. This is the time since the last packet from this node was received.

The screenshot shows the DOODLE labs web interface. On the left is a navigation sidebar with the following items: Status, System, Services (highlighted), Central Config (highlighted), SNMPD, Central ACS, Basic Settings, and Logout. The main content area is titled 'Central Config Status' and contains a table with the following data:

MAC-Address	HOSTNAME	TQ	Last Seen
00:30:1A:4E:BB:0A	smartradio-301a4ebb0a	94.51	00:13
00:30:1A:4E:BB:02	smartradio-301a4ebb02	96.86	00:13
00:30:1A:4E:BB:01	smartradio-301a4ebb01	96.08	00:00
00:30:1A:4E:BB:09	smartradio-301a4ebb09	self	00:13

Fig. 20 – Central Config Status

Central ACS

Central ACS (Auto-Channel-Selection) allows the radio to automatically select the best operating channel. Central ACS is enabled by navigating to *Services* → *Central ACS*. Central Config must be enabled first.

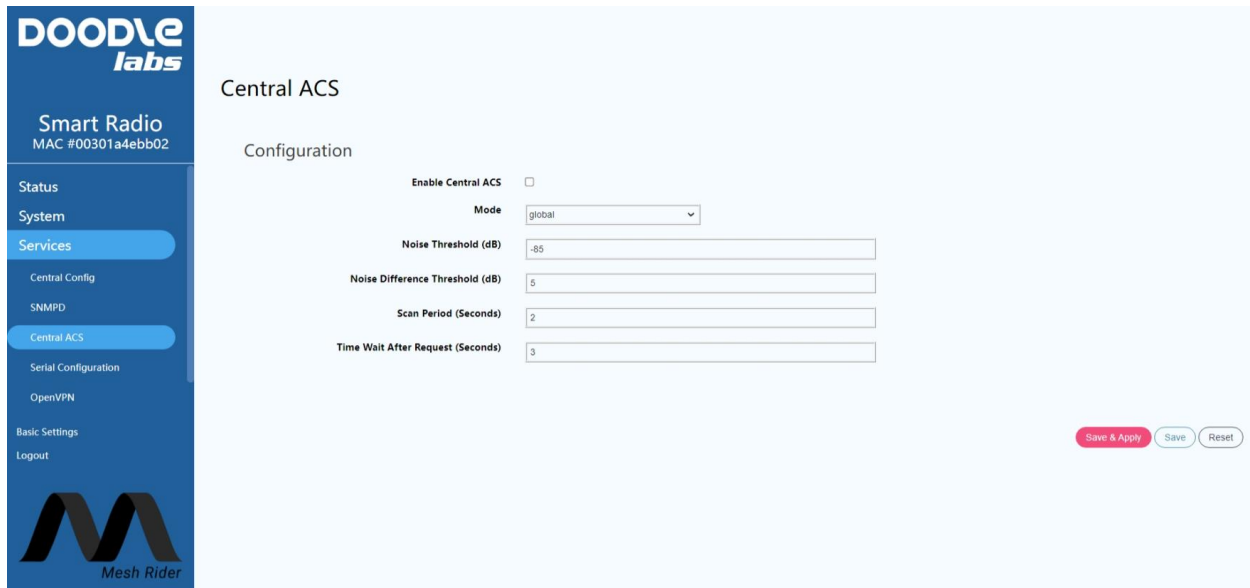


Fig. 21 – Central ACS

Central ACS operates in two modes: `global` and `local`. In both modes, the primary node measures background noise in the operating channel every `Scan Period` seconds and reacts when it exceeds the user-defined `Noise Threshold`.

In `global` mode, the primary node instructs all nodes to take background noise readings, collects the information, and then instructs all other nodes to change channels if there is another channel whose background noise is lower by `Noise Difference Threshold` or more. The timeout when talking to other nodes is `Time Wait After Request` seconds.

In `local` mode, the primary node monitor simply instructs all other nodes to change channels if there is another channel whose background noise is lower by `Noise Difference Threshold` or more.

In general, `local` mode is simpler and more responsive. We recommend using `local` mode with the UAV monitor the background noise.

Link Recovery

Link recovery is a new feature which is enabled automatically as long as Central Config is enabled. Essentially, if the channel is changed while a node is inaccessible (for example, out-of-range, or rebooting), then that node will scan and rejoin the network. There is no configuration for Link Recovery, however, take note that it only works for mis-matched channels and not fields like channel bandwidth, SSID, or password.

Backup and Upgrade

Configuration Backup

You can create a backup of your configuration in the web GUI by first clicking `Advanced Settings` in the bottom left-hand corner of the page, and then navigating to `system → backup/flash firmware` and clicking `Generate Archive`.

Firmware Upgrades

The Mesh Rider OS can be upgraded Over the Air (OTA) or from the Ethernet ports, using either the Web GUI, or the Linux console.

Linux Console Method

1. Copy firmware from PC to the module:

```
scp firmware-sysupgrade.bin root@<IP ADDRESS>:/tmp/
```

2. Login to Smart Radio with `ssh root@<IP ADDRESS>`
3. Use the `sysupgrade` command to update the firmware.

```
root@smartradio:~# sysupgrade -v /tmp/firmware-sysupgrade.bin
```

4. Wait until the update is complete. **DO NOT CYCLE THE POWER SUPPLY UNTIL THE FIRMWARE HAS BEEN UPDATED!**

Web GUI Method

1. Enter the following into the address bar of your browser

```
https://<IP ADDRESS>/cgi-bin/luci/admin/system/flashops
```

2. At the bottom of the page, de-select the “keep settings” button
3. In the “image” field, click browse and select your firmware image
4. Click Flash Image
5. After the system has verified the image, click Proceed

Configuration Copy

Simply restoring a configuration of another radio on to a different one can make the two radios incompatible with each other as there are several settings which are unique to each radio like the IP addressing and security keys. Doodle Labs has created a Configuration Clone tool to

Configuration Guide

quickly distribute a radio configuration to multiple radios. The tool currently only works over the command line. In order to use it,

1. [SSH](#) into the radio
2. To see a list of options, run

```
root@smartradio:~# configclone.sh --help
usage: configclone.sh <ACTION>
ACTIONS:
-b) create_backup
-r) restore_backup
-c) cleanup
-k) do not overwrite conflicting parameters during restore
-o) overwrite conflicting parameters during restore
```

```
output on create_backup: /tmp/backup.tar.gz
```

```
input on restore_backup: /tmp/backup.tar.gz
```

Running `configclone.sh -b` will create a backup (`/tmp/backup.tar.gz`) which can then be copied to other radios (`scp` to `/tmp/backup.tar.gz`) and restored with `configclone.sh -r`. During the backup and restore phases, the radio will ask questions to help you to resolve potential configuration conflicts. You can skip all questions in the restore phase by running it with the `-k` or `-o` options.

Connect with Tech Support

Before connecting with Tech Support, please read the Trouble-Shooting Guide available in our [Technical Library](#) section of the website.

Otherwise, please feel free to contact us (<https://www.doodlelabs.com/about-us/tech-support/tech-support-request-form/>). Please describe your problem in as much detail as possible, and include photos of the setup if possible.

References

- [1] Technical Library, <https://doodlelabs.com/technologies/tech-library/technical-library-application-notes/>, Feb 18 2022

Appendix A – Application Notes

Doodle Labs has developed many application notes for Smart Radio. They are available in the [Technical Library](#) section of the website. Below list shows some of the commonly used application notes.

1. Optimizing the wireless link
2. Concurrent HD Video Streaming and Low Latency C&C Tutorial
3. Integration with the Nvidia Jetson Nano
4. Integration with Pixhawk and QGroundControl
5. Smart Radio Remote Management guide
6. Serial Interface Guide
7. LTE Failover
8. An overview of the Mesh Rider OS

Appendix B – Extending the Smart Radio Functionality

Doodle Labs can supply an Image Builder Software Development Kit (SDK) for interested customers. The SDK allows customers to build custom packages and package them into a custom firmware image. It is also possible to make cosmetic changes to the OS to re-brand the radio. Our Image Builder SDK is available on request. Please contact sales@doodlelabs.com for more information.

Blinking LED

A simple LED blinking script is available in the Smart Radio which will cause one of the GPIOs to toggle ON/OFF. This script only applies to -J and -L hardware. Wearable models (-K) have a built-in LED blinking function which works separately. By default, LED blinking is disabled, but it can be enabled by running

```
root@smartradio:~# uci set led_blink.general.enable='1'  
root@smartradio:~# uci commit led_blink  
root@smartradio:~# /etc/init.d/startup_blink restart
```

For -L models, only GPIO0 blinking can be enabled. On -J models, you can see a full list of configuration parameters by running

```
root@smartradio:~# uci show led_blink  
led_blink.general=led_blink  
led_blink.general.inactive_thres='1000'  
led_blink.general.blink_rate='0.3'  
led_blink.general.gpio_selected='2'  
led_blink.general.enable='0'
```

The `blink_rate` is measured in seconds, and the `inactive_threshold` is the time since hearing from another station, after which the LED will start to blink.