

# *openSPOT•2*

Standalone digital radio IP gateway / hotspot



## **USER MANUAL**

[sharkrf.com/openspot2](http://sharkrf.com/openspot2)

[info@sharkrf.com](mailto:info@sharkrf.com)

# Table of Contents

openSPOT2 user manual	1.1
First steps	1.2
Connecting to a Wi-Fi network	1.2.1
Quick Setup	1.2.2
Setting up your transceiver	1.3
DMR radio setup	1.3.1
Example configurations	1.3.1.1
DMR usage notes	1.3.1.2
D-STAR radio setup	1.3.2
D-STAR APRS	1.3.2.1
C4FM/Fusion radio setup	1.3.3
Using the Wires-X button	1.3.3.1
C4FM/Fusion APRS	1.3.3.2
NXDN radio setup	1.3.4
P25 radio setup	1.3.5
POCSAG pager setup	1.3.6
LED states	1.4
Button usage	1.5

## Web interface

Opening the web interface	2.1
About the web interface	2.2
Status page	2.3
Connectors page	2.4
Modem page	2.5
Settings page	2.6
Configuration profile	2.6.1
Firmware upgrade	2.6.2
SharkRF Network	2.6.3
Voice announcements	2.6.4
Location settings	2.6.5
DMR settings	2.6.6
D-STAR settings	2.6.7
C4FM/Fusion settings	2.6.8
NXDN settings	2.6.9

P25 settings	2.6.10
POCSAG settings	2.6.11
Lock settings	2.6.12
Miscellaneous settings	2.6.13
Network page	2.7
Network traffic	2.7.1
Network settings	2.7.2
NTP settings	2.7.3
Wireless settings	2.7.4
IP settings	2.7.5
MAC settings	2.7.6
Side panels / Tools page	2.8
Quick call	2.8.1
BrandMeister Manager	2.8.2
DMR SMS chat	2.8.3
POCSAG/DAPNET	2.8.4
APRS chat	2.8.5
ID database lookup	2.8.6
Upgrade	2.8.7
Login page	2.9

## Connectors

Null	3.1
Homebrew/MMDVM	3.2
DMRplus dongle	3.3
DCS/XLX	3.4
REF/XRF	3.5
FCS	3.6
YSFReflector	3.7
NXDNReflector	3.8
P25Reflector	3.9
DAPNET	3.10
APRS	3.11
SharkRF IP Connector Client	3.12
SharkRF IP Connector Server	3.13
AutoCal	3.14

## Advanced usage

Powersaving	4.1
Control calls	4.2
Cross modem modes	4.3
USB serial console	4.4
API	4.5
Factory reset	4.6

## Miscellaneous

Troubleshooting	5.1
Specifications	5.2
Acknowledgement	5.3
Open source licences	5.4

# openSPOT2 user manual

This user manual is for the **stable** firmware version **v87**. It's updated with new information when a new stable firmware is released.

The PDF version is available [here](#).

## Support information

The openSPOT2 is designed to be very simple to use, but if you have questions, feel free to ask them in [email](#). You can also seek help in our [community forum](#).

Please note that we don't have official support on other platforms like Facebook.

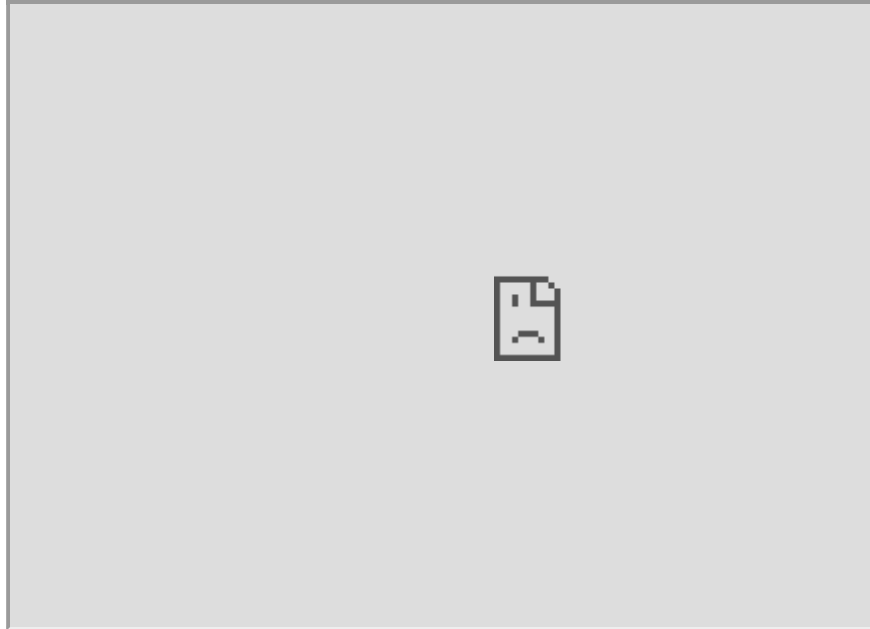
## Package contents

The openSPOT2 package contains an **AC/USB power supply** and a **USB cable**. These are **all you need** to operate the openSPOT2, however, you will need a web browser on a computer, tablet or mobile phone for setting up the openSPOT2 through it's web interface, but after that, the device can work **standalone**.

Please only use the supplied AC/USB power supply unless you are plugging the openSPOT2 to a power bank, a computer or your vehicle's built-in USB power adapter. The usage of inappropriate power supplies voids the warranty.

## First steps

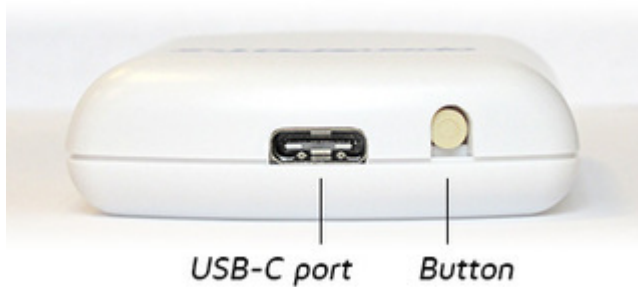
The following video shows how to set up a brand new openSPOT2 and how to connect it to the BrandMeister network using a C4FM/Fusion radio:



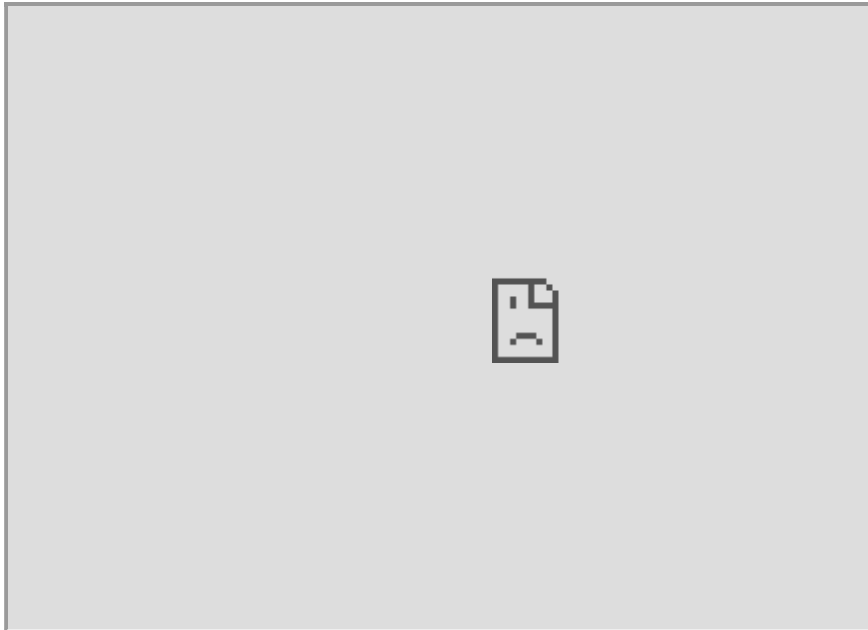
Here are the steps described in detail:

### Powering up

Power up the openSPOT2 by plugging the supplied USB-C cable into the USB-C port.



The openSPOT2 is designed to boot very quickly, as you can see on this video:



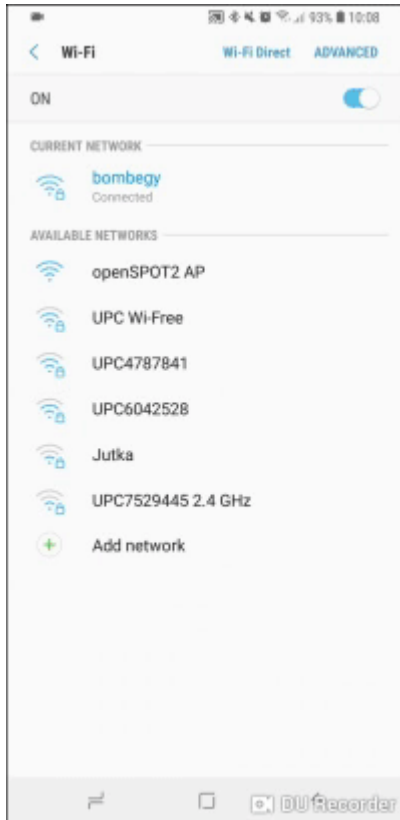
If the device is running with the factory default configuration, then there's an initialization running for about 15-20 seconds and the LED will be blinking white. After this, the LED will be slowly pulsating white indicating that the device has entered access point (AP) mode and it's broadcasting its own Wi-Fi network called **openSPOT2 AP**.



The AP mode is the openSPOT2's default mode of operation if no Wi-Fi networks are configured. AP mode lets you configure a Wi-Fi network which the openSPOT2 will connect in normal mode. You can switch between AP and normal mode anytime by pressing the [button](#) for 3 seconds.

## Connecting the openSPOT2 to a Wi-Fi network

This animation shows you the whole initialization setup process using a phone:



- Connect your web browser device (phone/tablet/computer) to the Wi-Fi network called **openSPOT2 AP**.

If there is no Wi-Fi network called **openSPOT2 AP**, then make sure the openSPOT2 is in AP mode ([white LED is pulsating](#)). You can switch between AP and normal mode anytime by pressing the [button](#) for 3 seconds.

- The initialization setup opens automatically.

If the initialization setup won't open automatically, then you can open it by entering <http://openspot2.local/> or <http://192.168.99.1/> into the web browser.

- Select the country of operation, then click on the **Next** button.
- Select which Wi-Fi network the openSPOT2 will use (usually your own Wi-Fi network) by clicking on the **Connect** button at the desired Wi-Fi network. Enter the Wi-Fi network password in the Network key dialog (make sure you enter the password case sensitive), then click on the dialog's **Connect** button.
- The openSPOT2 stops broadcasting its own Wi-Fi network, and automatically tries to connect to the selected Wi-Fi network.
- The web browser device (ex. your phone) detects that the **openSPOT2 AP** is no longer available, and automatically connects back to the previously used Wi-Fi network. If it does not, connect it to the same network you've selected during the initialization setup (usually your own Wi-Fi network).

A successful connection to the Wi-Fi network is indicated by a pulsating green/yellow LED on the openSPOT2 which means the Null connector is active and the device is connected to the Wi-Fi network.

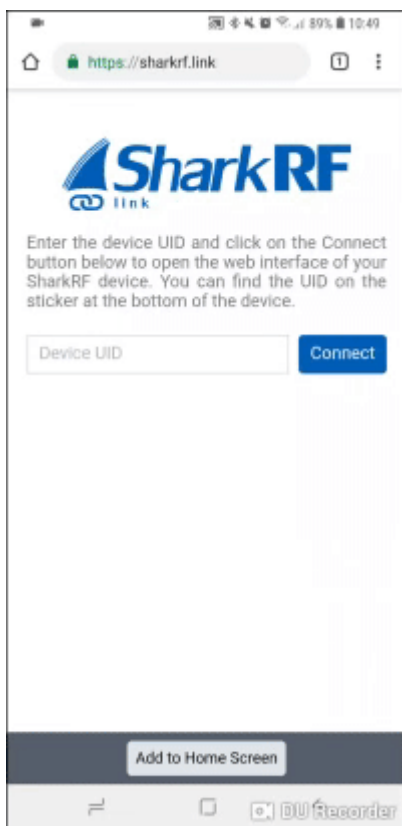




## Quick Setup initialization

Open [SharkRF Link](https://sharkrf.link) in your web browser. Enter the openSPOT2's device UID (you can find this ID on the sticker at the bottom of the openSPOT2), then click on the **Connect** button.

The openSPOT2's web interface will open in your web browser. Fill in your callsign and IDs in the Quick setup, then click on the **Save** button:

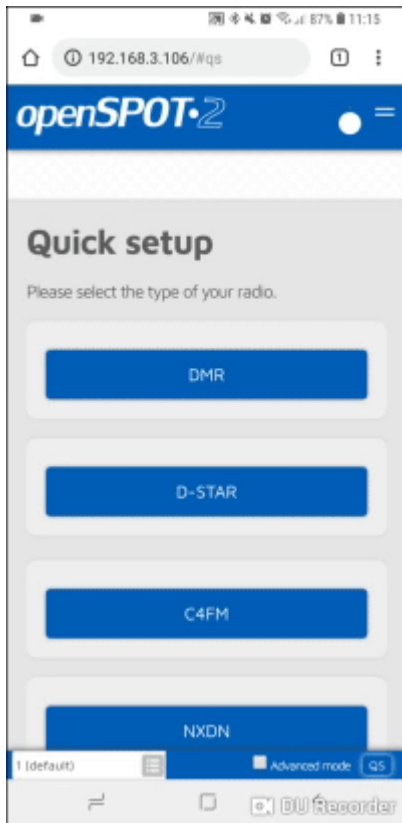


## Quick Setup

Quick Setup helps you to configure the openSPOT2 with your transceiver and connect to the preferred network with the following steps:

- Select the type of your radio
- Set some basic settings like the frequency, and select the network you want to connect (this will set which connector will be used)
- Select the server to connect, set the needed network parameters (like your callsign, IDs, etc.), and click on the **Connect** button
- After the openSPOT2 is connected to the selected server it is **ready to use** with your transceiver

The following animation shows the Quick Setup for connecting to the BrandMeister network using a DMR transceiver:



## Setting up your transceiver

Here are some tips on setting up different types of transceivers.

### DMR radio setup

- Set up a simplex channel (with the same RX and TX frequencies). You can set the timeslot to any value, as it is not used in simplex mode.
- Set the channel's color code to the openSPOT2's color code (by default it's 1, it can be changed on the [Settings page, DMR settings](#) section).
- Set the TX contact for the current channel to the talkgroup you want to talk on. If you want to use a reflector, or [Homebrew call rerouting](#), set it to ID 9 with a group call (which means TG9).
- Create an RX group list and add TG9 to it. Assign this RX group list to the previously created simplex channel. This way you'll be able to hear the openSPOT2's voice announcements, which will be sent to TG9.
- Create a contact with private call to ID 9999, which is the openSPOT2's built-in [local echo service](#). Make a call to this ID to test the connection between the transceiver and the openSPOT2.
- Run AutoCal in case you have voice quality issues. See the [AutoCal's usage description](#) in the Connectors section below.

Take a look at the [example channel configurations](#).

### D-STAR radio setup

- Set the radio to single frequency mode.
- If you want to use a reflector, set the URCALL on your radio to CQCQCQ, and the local module on the openSPOT2 to D.
- If you want to use a gateway, set the URCALL on your radio to the gateway's callsign and remote module. The remote module should be the last character of the URCALL. Make sure you have registered your local module on the D-STAR gateway registration system. If you have a registration, you have probably registered the "space" local module. In this case set your openSPOT2's local module to a space (empty) character.
- You can check your D-STAR registration [here](#).
- You don't have to care about RPT1 and RPT2 settings.
- You can make a test call to the openSPOT2's built-in [local echo service](#) to test the connection between the transceiver and the openSPOT2 by setting the URCALL to `.....E` (7 spaces and an E, replace dots with spaces).

Take a look at the [D-STAR APRS](#) page if you want to forward your transceiver's GPS position to the APRS network.

If you use a Kenwood TH-D74A transceiver, then make sure the *Direct Reply* feature is turned off, as it basically causes automatic callsign routing, which could cause one's transmission to not be heard on the used reflector.

## C4FM/Fusion radio setup

- Set the radio to single frequency mode.
- Use digital narrow (DN) mode. Voice wide (VW) mode does not support late entry.
- If your radio has half deviation mode turned on, make sure you use the *C4FM Half Deviation* modem mode on the openSPOT2.
- You can make a test call to the openSPOT2's built-in [local echo service](#) to test the connection between the transceiver and the openSPOT2 by setting the DGID to 99 in your radio. Don't forget to set the DGID back to AUTO (or ALL or 00, depending what's available on your radio) after you've done with the local echo test.
- Run AutoCal in case you use the C4FM Half Deviation mode and there are voice quality issues. See the [AutoCal's usage description](#) in the Connectors section below.

Take a look at the [Wires-X button usage description](#) and the [C4FM/Fusion APRS](#) page if you want to forward your transceiver's GPS position to the APRS network.

## NXDN radio setup

- Set up a simplex channel (with the same RX and TX frequencies).
- Set the channel's Radio Access Number (RAN) to the openSPOT2's RAN (by default it's 0, it can be changed on the [Settings page, NXDN settings](#) section).
- Create a contact with private call to ID 9999, which is the openSPOT2's built-in [local echo service](#). Make a call to this ID to test the connection between the transceiver and the openSPOT2.
- Run AutoCal in case you have voice quality issues. See the [AutoCal's usage description](#) in the Connectors section below.

## P25 radio setup

- Set up a simplex channel (with the same RX and TX frequencies).
- Set the channel's Network Access Code (NAC) to the openSPOT2's NAC (by default it's 0, it can be changed on the [Settings page, P25 settings](#) section). Note that **the openSPOT2 requires the NAC in decimal format** (example: if you have the NAC 0x293 in your radio then you have to enter 659 in the openSPOT2).
- Create a contact with private call to ID 9999, which is the openSPOT2's built-in [local echo service](#). Make a call to this ID to test the connection between the transceiver and the openSPOT2.
- Run AutoCal in case you have voice quality issues. See the [AutoCal's usage description](#) in the Connectors section below.

## POCSAG pager setup

- Make sure you set the your pager's receive frequency to the openSPOT2's POCSAG frequency, and that the baud rates match. Polarity should be

normal. The openSPOT2's POCSAG frequency and other settings can be changed on the [Settings page](#), [POCSAG settings](#) section.

- Make sure you set the RIC(s) in your pager's configuration to the address(es) you want to receive messages on.
- See the [DAPNET Wiki](#) for more information about POCSAG and DAPNET.

# Example DMR radio channel configurations

## Hytera

Channel Alias OSW TG216 (The actual display may change, See the Help for details)

Color Code 1

Slot Operation Slot 1

Pseudo Trunk Designated TX None

Scan List/Room List None

Auto Start Scan

Quick GPS

Talk Around

Rx Only

IP Multi-site Connect

Auto Start Roam

VOX

Option Board

Priority Interrupt Encode

Priority Interrupt Decode

Reliable Priority Interrupt Transmit

Enhanced Channel Access

Over the Air Encrypt Type None

Only Receive Encrypt Air

Over the Air Encrypt Key None

Rx	Offset (MHz)	Tx
Receive Frequency [MHz] 433.900000	0.000000	Transmit Frequency [MHz] 433.900000
Rx Group List RX group list	Copy	Tx Contact Name TG216
Emergency Alarm Indication <input type="checkbox"/>		GPS Revert Channel None
Emergency Alarm Ack <input type="checkbox"/>		RRS Revert Channel None
Emergency Call Indication <input type="checkbox"/>		Emergency System DigitalSys 1
Encrypt		Phone System Phone System 1
Encrypt <input type="checkbox"/>		Power Level Low
Encrypt Type Basic		Tx Admit Color Code Free
Encrypt Key None		In Call TX Admit Follow TX Admit
MultiKey Decrypt <input type="checkbox"/>		Tx Time-out Time [s] 240
Random Key Encrypt <input type="checkbox"/>		TOT Pre-Alert Time [s] 0
		TOT Re-key Time [s] 0
		TOT Reset Time [s] 0
		Private Call Confirmed <input type="checkbox"/>
		RSSI Threshold -113

## Motorola

Voice Announcement File

Dual Capacity Direct Mode

Timing Leader Preference

Scan/Room List

Auto Scan

Color Code

Repeater/Time Slot

Phone System

ARS

Enhanced GNSS

Window Size

Privacy

Privacy Alias

RAS Alias

Option Board

Option Board Trunking

Lone Worker

Allow Talkaround

IP Site Connect

Per-Site RSSI Threshold (dBm)

Messaging Delay (ms)

Compressed UDP Data Header

Text Message Type

Channel Inhibit

Fist Microphone Disable

Telemetry VIO 1 GPO Level

Telemetry VIO 2 GPO Level

Telemetry VIO 3 GPO Level

Telemetry VIO 4 GPO Level

Telemetry VIO 5 GPO Level

RX Only

**RX**

Frequency (MHz)

Offset (MHz)

Ref Frequency

Group List

Emergency Alarm Indication

Emergency Alarm Ack

Emergency Call Indication

Emergency Call Decode Tone

**TX**

Frequency (MHz)

Ref Frequency

Contact Name

Emergency System

VOX

Power Level

TOT (sec)

TOT Retry Delay (sec)

Allow Interruption

TX Interruptible Frequencies

Admit Criteria

In Call Criteria

RSSI Threshold (dBm)

GNSS Revert

Private Call Confirmed

Data Call Confirmed

Location Data Delivery Mode

Enhanced Channel Access

CSBK Data

Tyt

Digital/Analog Data		Digital Data	
Channel Mode	Digital	Channel Name	OSW
Band Width	12.5kHz	RX Frequency(MHz)	433.90000
Scan List	None	TX Frequency(MHz)	433.90000
Squelch	Normal	Admit Criteria	Channel Free
RX Ref Frequency	Low	Auto Scan	<input type="checkbox"/>
TX Ref Frequency	Low	Rx Only	<input type="checkbox"/>
TOT[s]	555	Lone Worker	<input type="checkbox"/>
TOT Rekey Delay[s]	0	VOX	<input type="checkbox"/>
Power	Low	Allow Talkaround	<input type="checkbox"/>
CTCSS/DCS Dec: None QT Reverse: 180 Rx Signaling System: Qff <input type="checkbox"/> Display PTT ID		Private Call Confirmed <input type="checkbox"/> Emergency Alarm Ack <input type="checkbox"/> Data Call Confirmed <input type="checkbox"/> Compressed UDP Data Header <input type="checkbox"/> Emergency System: None Contact Name: TG216 Group List: GroupList1 Color Code: 1 Repeater Slot: 1 Privacy: None Privacy No.: 1	
CTCSS/DCS Enc: None Tx Signaling System: Qff <input checked="" type="checkbox"/> Reverse Burst/Turn-off Code		Decode 1 <input type="checkbox"/> Decode 2 <input type="checkbox"/> Decode 3 <input type="checkbox"/> Decode 4 <input type="checkbox"/> Decode 5 <input type="checkbox"/> Decode 6 <input type="checkbox"/> Decode 7 <input type="checkbox"/> Decode 8 <input type="checkbox"/>	



## DMR usage notes

There are two basic types of DMR calls your transceiver can transmit or receive: **private** and **group**. The openSPOT2 forwards what it receives from your radio to the network, or from the network to your radio. The network may forward your calls to its other connected stations (hotspots/repeaters/gateways) or calls from other connected stations to you based on linked **talkgroups** and **reflectors**.

Talkgroups and reflectors are network features, **they are completely handled by the network and not the openSPOT2**, and their behavior may be different on each network.

### Talkgroups

DMR networks usually have **static** or **dynamic** linking available for a talkgroup. Statically linked TGs **remain linked all the time**, dynamically linked TGs stay linked for you **only for a limited amount of time after the last call** (usually 15 minutes on BrandMeister, depending on the currently used server's configuration).

**Having a talkgroup linked either dynamically or statically means that stations which have the same linked talkgroup will receive all (usually group) calls sent to that talkgroup.**

### Dynamic talkgroups

You can link a TG dynamically by simply starting a (group) call to the talkgroup ID. The network will automatically link your openSPOT2 to the called TG ID.

To **unlink** all dynamically linked talkgroups, call TG4000 (start a short group call to DMR ID 4000).

**Note:** calling TG4000 will also unlink linked reflectors, not just all dynamic talkgroups.

### Dynamic talkgroups on the BrandMeister Network

The BrandMeister Network lets you call any talkgroup ID and will create that talkgroup dynamically for you.

There's BrandMeister Network feature for hotspots called **auto static talkgroups**: the first talkgroup ID you call **after calling TG4000** will be linked auto static which means **it won't be unlinked after a timeout**. You can unlink this auto static TG by calling TG4000.

### Static talkgroups on the BrandMeister Network

Statically linked TGs can be configured on the [BrandMeister Network's dashboard](#) after logging in, or by using the openSPOT2's [BrandMeister Manager](#).

## BrandMeister manager

**API key** Get key Check key Save  
API key:

**Actions** Drop current call


**Static talkgroups** Add  
Add static talkgroup ID:   
216  
Currently linked static TGs: 216 ✕, 3100 ✕

**Dynamic talkgroups** Drop all Quick call  
Quick call talkgroup ID:   
21667  
Currently linked dynamic TGs: -

**Reflector** Save  
Active ID (4000 - not linked):   
4000

### Static talkgroups on DMRplus/IPSC/other networks

You can define what TGs to link statically at the DMRplus options section at the bottom of the [Homebrew/MMDVM connector](#) settings page.

MMDVM options for DMRplus	
	<input checked="" type="checkbox"/> Use DMRplus options
Start reflector:	4001 Germany 
Start reflector ID:	4001
Relink time (min):	
	<input type="checkbox"/> User link allowed
Static TG TS1 #1:	1
Static TG TS1 #2:	
Static TG TS1 #3:	
Static TG TS1 #4:	
Static TG TS1 #5:	
Static TG TS2 #1:	
Static TG TS2 #2:	
Static TG TS2 #3:	
Static TG TS2 #4:	
Static TG TS2 #5:	
Raw options:	StartRef=4001;UserLink=0;TS1_1=1;

## Reflectors

Reflector IDs are between DMR ID 4000 and 5000. To link a reflector, start a short **private** call to the reflector ID. To unlink the reflector, start a short **private** call to DMR ID 4000.

Only 1 reflector can be linked at a time. To talk on the linked reflector, send your calls to DMR ID 9 with a group call (TG9). You'll receive all calls from the reflector also to TG9.

### Reflectors on the BrandMeister Network

Reflector usage is deprecated (and nowadays mostly disabled) on BrandMeister servers. **Use talkgroups instead.**

## Advices for hotspot usage

### BrandMeister Network

Avoid statically linking talkgroups and only **use one auto static talkgroup at a time.**

### DMRplus/IPSC/other networks

Define what TGs to link statically at the DMRplus options section at the bottom of the [Homebrew/MMDVM connector](#) settings page.

### Changing modules on XLX servers

To unlink the currently linked module, call DMR ID 4000. To connect to module A, call DMR ID 4001, for module B call DMR ID 4002 and so on.

## **Example**

If you have TG969 linked either dynamically or statically, then the server will forward group calls (coming from other stations (repeaters, hotspots, gateways)) going to DMR ID 969 to your openSPOT2. Your openSPOT2 will transmit it on RF and your radio will receive it. Your radio will open the squelch if:

- The TX contact of the currently active channel is set to TG969, or
- You have an RX group list with TG969 in it and this RX group list is assigned to the currently active channel.

## D-STAR APRS

You can set up the openSPOT2 to forward your D-STAR transceiver's location information to the APRS network, so it will be visible on APRS maps like [aprs.fi](http://aprs.fi)

### Setting up the APRS connection

First you have to set up the APRS special connector using the openSPOT2's Quick Setup.

**Quick setup**    « Back    Cancel

**APRS settings**

This special background connector can be used for setting up a connection to the APRS network. This enables device location broadcasting, forwarding GPS position data from transceivers, and messaging (APRS chat).

Enable in background

Server: rotate.aprs2.net

Server address: rotate.aprs2.net

Callsign: HA2NON

**Device location**

Allow uploading device location

QTH locator ([map](#)): JN97MM14fi

Latitude (decimal degrees): +47.518142

Longitude (decimal degrees): +19.01024

Height (ASL, meters): 155

Location symbol on map ([select](#)): &

Location comment: SharkRF openSPOT2

Save

Make sure you check the **Enable in background** checkbox. You don't need to check the **Allow uploading device location** checkbox for D-STAR location forwarding.

You can verify the APRS connection status on the Status page.

### Setting up your D-STAR transceiver

The following setup is described using the Icom ID51's menu structure. The menu structure of your transceiver may be different, but the names of the settings should be the same or similar.

### Enabling the GPS receiver

Go to **Menu > GPS > GPS Set > GPS Select** and select **Internal GPS**.

### Selecting the position data format

Go to **Menu > GPS > GPS TX Mode** and select **D-PRS (DV-A)**. This is the easiest method. You can enable the **Course/Speed** extension.

Now your radio will transmit location information during your calls if it has a GPS fix and you hold the PTT for at least 5-10 seconds. The APRS location comment will be your D-STAR radio's TX message (**Menu > My Station > TX Message**).

### Advanced settings

If you want to select your own APRS symbol, SSID and comment for your transceiver then enable *Advanced mode*, go to the Connectors page, open the APRS connector settings, and enable forced SSID/symbol/comment.

You can also set the APRS symbol and SSID in your transceiver without using forced settings on the web interface: select the **NMEA(DV-G)** GPS TX mode. Enable the **RMC** and **GGA** GPS sentences. Other sentences are ignored by the openSPOT2 so turn them off.

You can set the APRS symbol for your transceiver by entering it as the first 2 characters of the **GPS Message**. You can use the symbol selector on the openSPOT2's web interface at the APRS connector settings to find out the character code of a symbol.

You can set an SSID for your transceiver by entering it after the APRS symbol in the GPS Message.

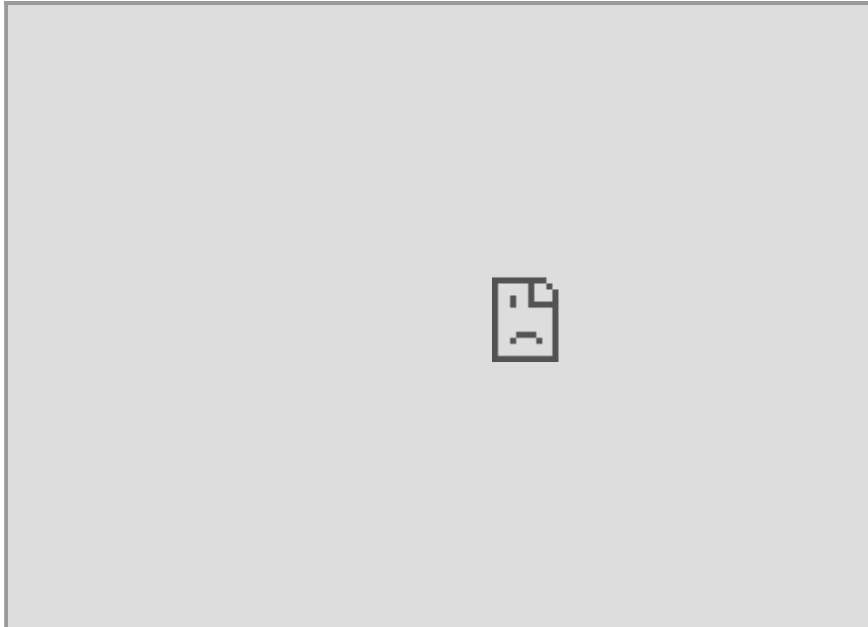
Setting a symbol and an SSID is optional. You can enter 2 space characters for the APRS symbol, or nothing for the SSID, or you can leave the GPS message completely empty if you want.

GPS message examples:

- House symbol with no SSID: `/y`
- Car symbol with SSID 9: `/>9`
- Default symbol with SSID 12: `..12` (replace the 2 dots with 2 spaces)

## Using the Wires-X button

Here's a video about the openSPOT2's Wires-X button support:



If the openSPOT2's modem is in C4FM (or C4FM Half Deviation) modem mode, then the device will automatically broadcast Wires-X information which will be displayed on radios in **Wires-X mode**. You can switch your C4FM/Fusion radio into Wires-X mode by **pressing its Wires-X button** (sometimes labelled as **DX button**).



The openSPOT2 displays the currently active configuration profile number at the bottom right corner of the FT2D's display. Displayed information may look different on other type of C4FM/Fusion radios. The examples here are demonstrated using an FT2D.

Using your openSPOT2 with a C4FM/Fusion radio in Wires-X mode is very convenient as you'll always **see the active server's name**, and you can **easily switch between servers** using the radio's user interface.

## Listing servers

You can get a server listing by selecting the **Search & Direct** option and then selecting the **All** button.



The first 5 entries will be your openSPOT2's configuration profiles. The list will contain the servers of the currently active connector. In case the currently active connector does not have a server list (ex. the Null connector), YSFReflector servers will be listed. To switch to an entry, select it using the dial and press PTT for a short time (or just tap on it in case you have a touch display).

## Searching for servers

You can search in the server list by selecting the **Search & Direct** option and then selecting the **Search & Direct** button.

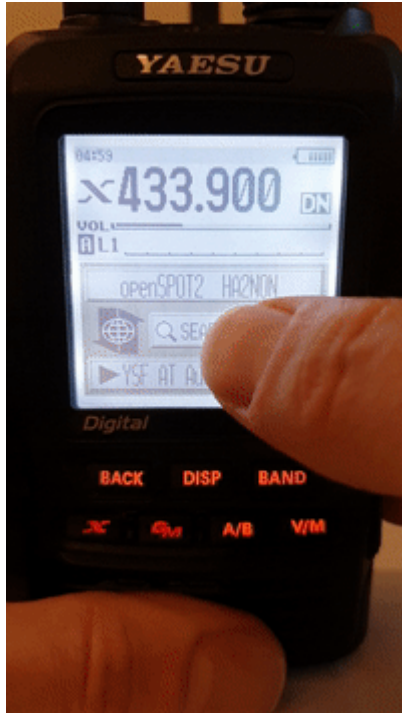




You can enter a part of the server name. All matching servers will be listed. To switch to a server, select it using the dial and press PTT for a short time (or just tap on it in case you have a touch display).

## Direct ID entry

You can connect to YSFReflector servers directly if you know the desired server's ID. Select the **Search & Direct** option, then select the **Search & Direct** button, then select the **ID** button. Enter the server ID and then select the **ENT** button. Server IDs are 5 digit long, make sure you've entered all digits otherwise the radio won't let you select the **ENT** button. Also make sure the ID starts with a **# character** (not all transceivers enter the # before the ID automatically).

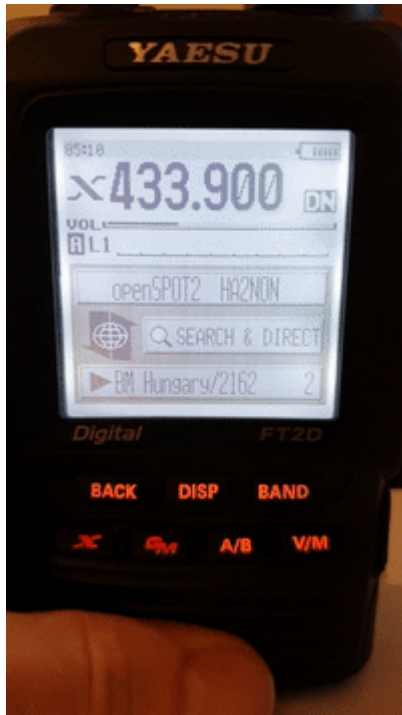


## Categories

You can save your favorite servers by adding them to categories. The openSPOT2 will automatically switch to the saved server's connector in case it's needed.

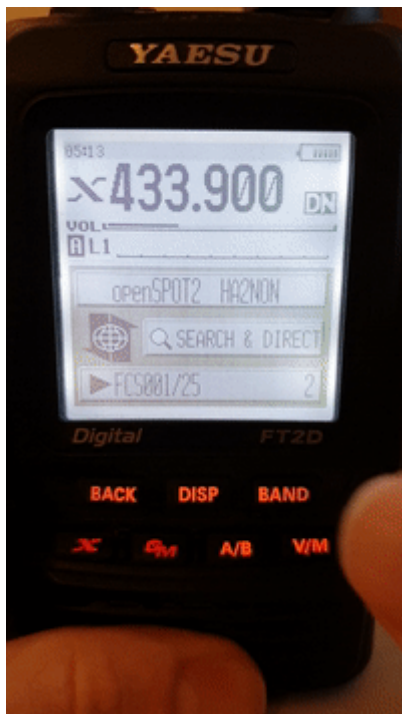
### Adding a server

Select the server name on the main Wires-X screen (the option at the bottom). This will load the station information screen. Select the **Add** button, then select a category.



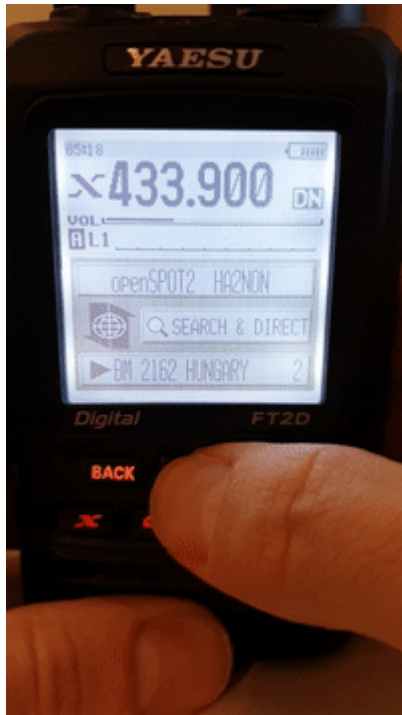
### Recalling a server

Select the **Search & Direct** option, then select a category to list its saved servers. To switch to a server, select it using the dial and press PTT for a short time (or just tap on it in case you have a touch display).



### Deleting/renaming

You can delete a server from a category in your radio's **main settings menu**. Select the **Wires-X** option and then the **Remove room/node** option.



You can rename a category also in this **Wires-X** menu.

## C4FM/Fusion APRS

You can set up the openSPOT2 to forward your C4FM/Fusion transceiver's location information to the APRS network, so it will be visible on APRS maps like [aprs.fi](http://aprs.fi)

### Setting up the APRS connection

First you have to set up the APRS special connector using the openSPOT2's Quick Setup.

**Quick setup**
« Back
Cancel

**APRS settings**

This special background connector can be used for setting up a connection to the APRS network. This enables device location broadcasting, forwarding GPS position data from transceivers, and messaging (APRS chat).

Enable in background

Server:

Server address:

Callsign:

**Device location**

Allow uploading device location

QTH locator (map):

Latitude (decimal degrees):

Longitude (decimal degrees):

Height (ASL, meters):

Location symbol on map (select):

Location comment:

Save

Make sure you check the **Enable in background** checkbox. You don't need to check the **Allow uploading device location** checkbox for C4FM/Fusion location forwarding.

You can verify the APRS connection status on the Status page.

### Setting up your C4FM/Fusion transceiver

The following setup is described using the Yaesu FT2D's menu structure. The menu structure of your transceiver may be different, but the names of the settings should be the same or similar.

### **Enabling the GPS receiver**

Go to **DISP > APRS > GPS POWER** and select **ON**.

### **Enabling GPS data sending during voice calls**

Go to **DISP > TX/RX > DIGITAL > LOCATION SERVICE** and select **ON**.

Now your radio will transmit location information during your calls if it has a GPS fix and you hold the PTT for at least 2-3 seconds. The openSPOT2 will automatically set an SSID and symbol for location reports based on the transceiver type in the received location data.

### **Advanced settings**

If you want to select your own APRS symbol, SSID and comment for your transceiver then enable *Advanced mode*, go to the Connectors page, open the APRS connector settings, and enable forced SSID/symbol/comment.

## LED states

- Factory default configuration, access point (AP) mode initializing:



This LED state is shown when the openSPOT2 is running with the factory default configuration. The initialization takes about 15-20 seconds.

- Access point (AP) mode active, no clients connected:



AP mode is the openSPOT2's default mode of operation if no Wi-Fi networks have been configured.

The LED will show this state if the openSPOT2 is broadcasting its own Wi-Fi network called **openSPOT2 AP**. See the [Connecting the openSPOT2 to a Wi-Fi network](#) section for information on how to connect to this access point and set up the openSPOT2.

If you've previously set up the openSPOT2 and want to switch it back to normal mode, press the [button](#) for 3 seconds.

- Access point (AP) mode active, client connected:



The LED will show this state if you've connected your browser device (phone/tablet/computer) to the openSPOT2's own Wi-Fi network.

- openSPOT2 is trying to connect to a Wi-Fi network:



Connecting to a Wi-Fi network should only take a few moments. If the LED is showing this state for too long, then please make sure you are in the range of a previously set Wi-Fi network, or turn on AP mode by pressing the [button](#) for 3 seconds, and set up a new Wi-Fi connection again. See the [Connecting the openSPOT2 to a Wi-Fi network](#) information on how to set up a Wi-Fi network connection in AP mode.

- openSPOT2 is connected to the Wi-Fi network and trying to connect to the selected server:



Connecting to a server should only take a few moments. If the LED is showing this state for too long, then please see more information in the [Troubleshooting](#) section.

- openSPOT2 is connected to the Wi-Fi network and a special connector is active (ex. [Null connector](#) or [AutoCal](#)):



- openSPOT2 is connected to the selected server and is standing by:



- openSPOT2 is connected to the selected server and is in [powersave mode](#):



- openSPOT2 is connected to the Wi-Fi network, but there's a problem with the internet connection:





The LED will show this state if the openSPOT2 is connected to a Wi-Fi network, trying to connect to the selected server, but it can't because there's a problem with the internet connection. This error can be caused if the router, network, or the internet service provider is using a proxy or some kind of traffic filtering which blocks the openSPOT2 from connecting.

- Upgrading Wi-Fi module firmware:



The LED may show this state for 1-2 minutes after a successful firmware upgrade, if there's a new Wi-Fi module firmware is available and being flashed to the openSPOT2's Wi-Fi module. **Do NOT unplug the openSPOT2 during the upgrade!**

- Modem transmitting:



A brief yellow LED during transmission signals an invalid received packet sequence number (which means network packet loss).

- Modem trying to transmit, but an active [BCLO](#) is preventing it:



- Modem receiving:



- Hardware error:



- Bootloader running, standing by:



The LED will show this state only when the bootloader has been started manually.

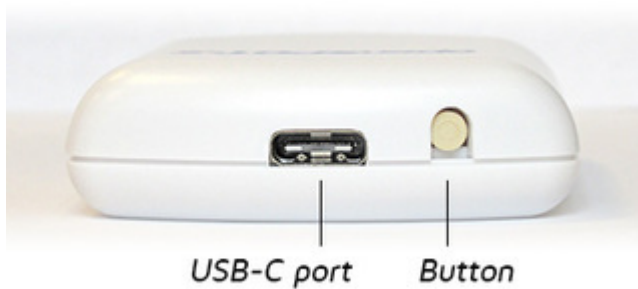
- Bootloader working, upgrading firmware:



The bootloader will automatically reboot and start the openSPOT2 after an automatic firmware upgrade.

**Do NOT unplug the openSPOT2 during the upgrade!**

## Button usage



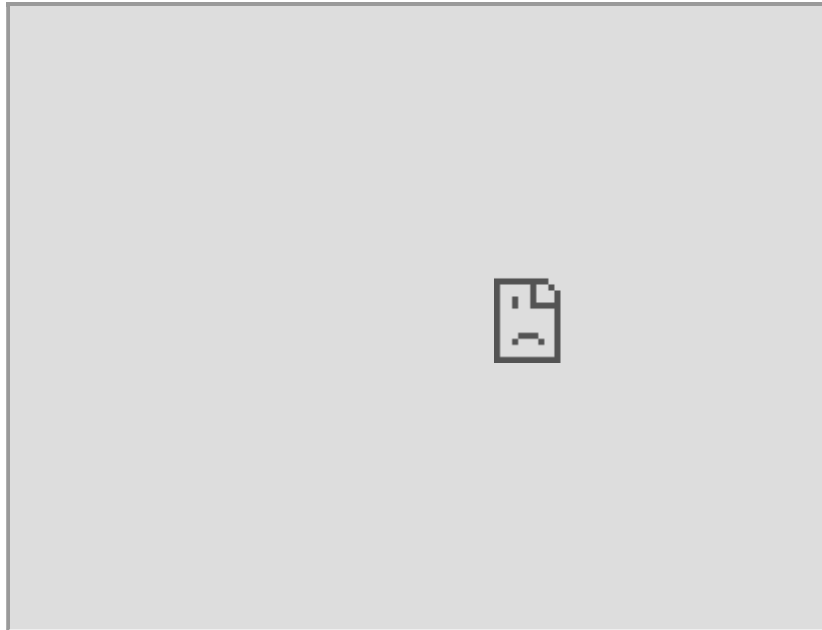
**Please only use a ballpoint pen to press the button, don't use a sharp item like a needle!**

The LED will blink white while the button is held. The blink interval indicates how much time is left until the button press is confirmed. Confirmation is indicated by two short white LED blinks.



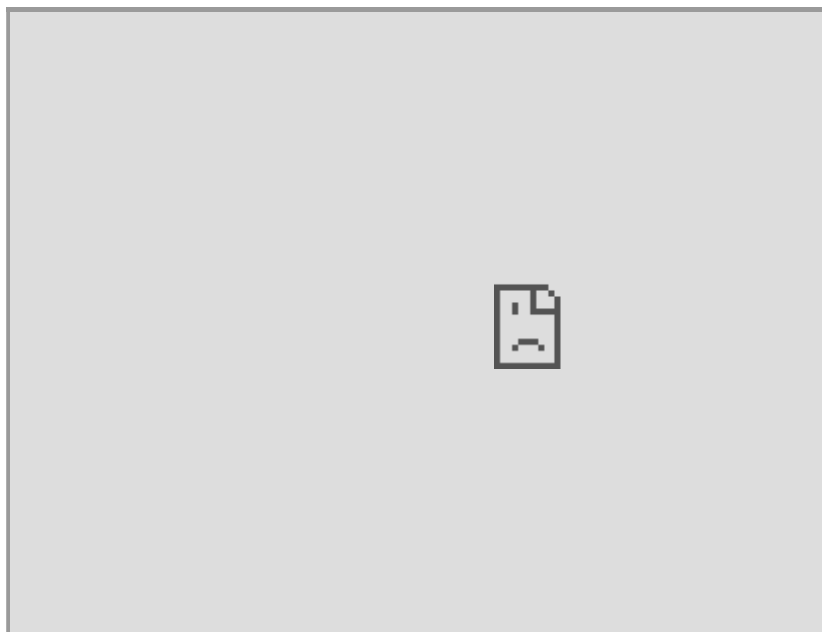
- Hold the button for 3 seconds to switch the openSPOT2 between Wi-Fi access point (AP) mode and normal mode.
  - In **AP mode**, the openSPOT2 broadcasts its own Wi-Fi network called **openSPOT2 AP**. The LED is pulsating white in access point (AP) mode and solid white if a Wi-Fi client is connected. You can use this mode to set the openSPOT2's Wi-Fi networks. See the [First steps](#) section for more information.
  - In **normal mode**, the openSPOT2 is trying to connect to one of the Wi-Fi networks which have been set on the [Network page](#), [Wireless settings](#) section (or using the [Initialization setup](#) in AP mode).

Here's a video showing how to switch the openSPOT to AP mode and connect it to a Wi-Fi network:



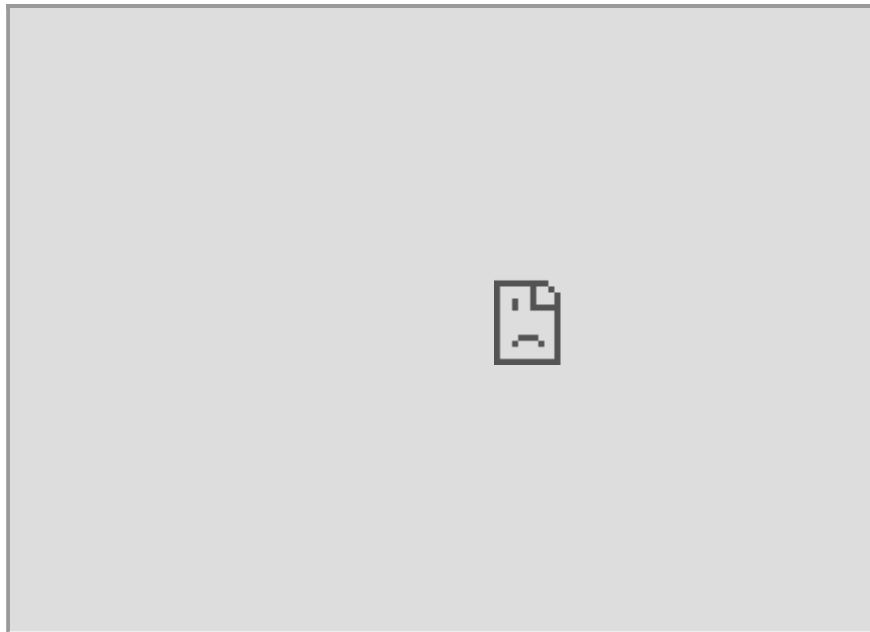
If there's no Wi-Fi network set on the Network page (Wireless settings section) then the openSPOT2 automatically switches to AP mode.

- Hold the button for 30 seconds to perform a full factory reset. In this case all configuration profiles will be cleared.



- Press the button 3 times quickly to enable/disable [powersaving](#).

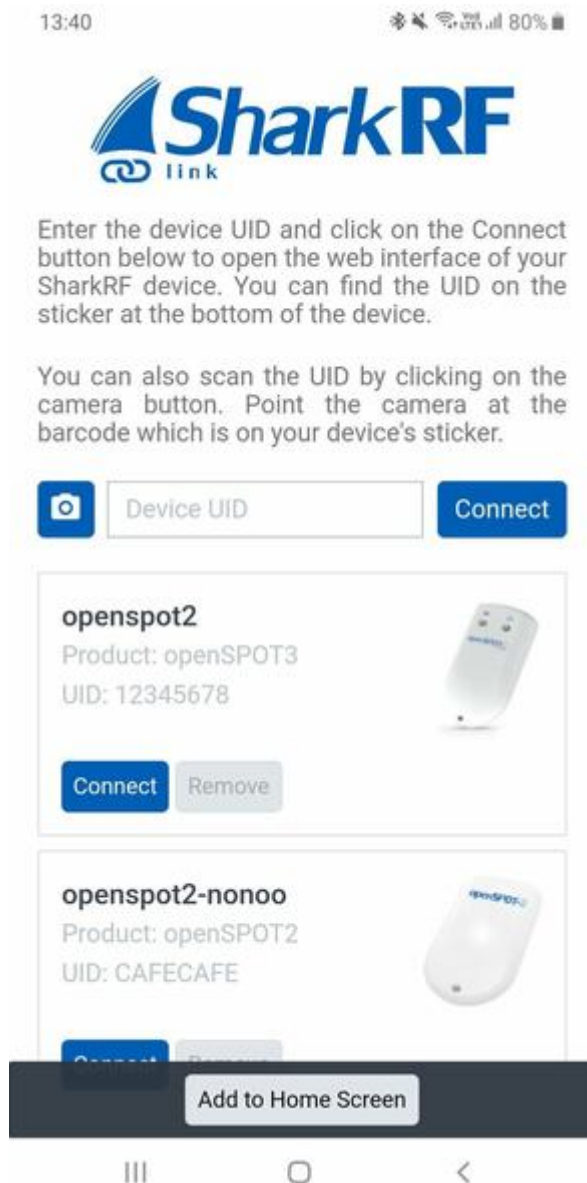
## Opening the web interface



You can use [SharkRF Link](https://sharkrf.link) to easily open the web interface of the openSPOT2. Enter the **device UID** (you can find this ID on the **sticker at the bottom of the openSPOT2**), then click on the **Connect** button.



All entered device UIDs will be saved in your web browser's local storage so you don't have to enter them again. Just click on each openSPOT2's Connect button to open their web interfaces.



Please keep in mind that your computer/tablet/phone with which you've opened [SharkRF Link](#) needs to be on the same network the openSPOT2 is connected to.

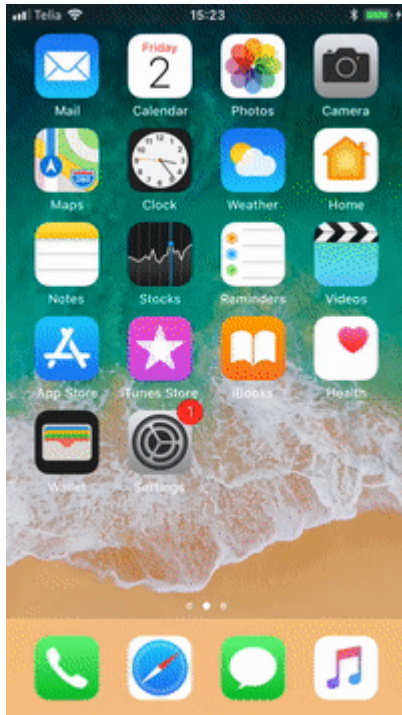
You can also add the [SharkRF Link](#) application to your phone's home screen by clicking on the **Add to Home Screen** button at the bottom of the page on your mobile phone's browser.



If you are using an iPhone and the **Add to Home Screen** button does not appear at the bottom of the [SharkRF Link](https://sharkrf.link) page, you can use the following method to add the app icon to your home screen:

- Open [SharkRF Link](https://sharkrf.link) in the browser
- Click on the iPhone's **Share icon** at the bottom
- Click on the **Add to Home Screen** button

You can see the whole process on this animation:



## Other ways to open the web interface

You can also open the web interface by entering <http://openspot2.local> or <http://openspot2/> or the openSPOT2's IP address into the web browser.

Please keep in mind that your computer/tablet/phone needs to be on the same network the openSPOT2 is connected to.

There are several methods to find out the openSPOT2's IP address:

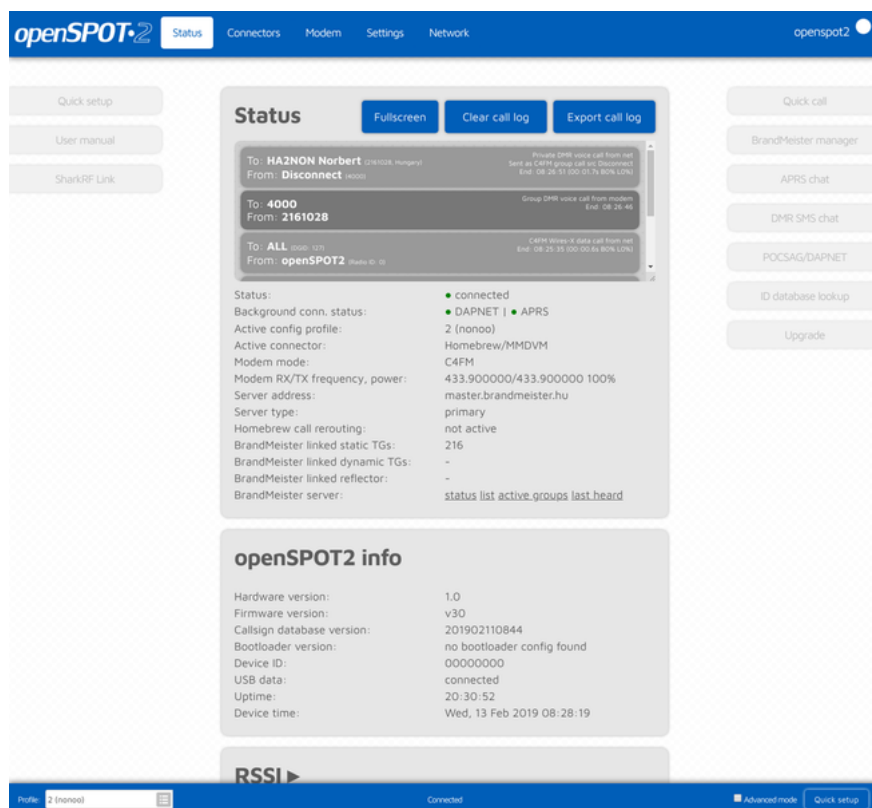
- Look at the DHCP clients list in your router's web interface.
- If [voice announcements](#) are enabled on the openSPOT2, you can query the active IP address:
  - **DMR**: start a call to the IP address query ID (by default it's 9997). In DMR mode, voice announcements are sent to TG9, so to hear them, make sure you have TG9 in your radio's RX group list.
  - **D-STAR**: start a call to the repeater info callsign `.....A` (7 spaces and an A, replace dots with spaces), or enter DTMF code B.
  - **C4FM/Fusion**: enter the [Special command/cross mode private call DTMF code](#) (by default it's \*) followed by DTMF code B.
  - **NXDN**: start a call to the IP address query ID (by default it's 9997).
- On Android and iOS phones, you can use MDNS/Bonjour and network discovery applications. A recommended app for Android is [Fing Network Tools](#).

**Notes:** Android phones do not support LLMNR or MDNS name resolving so <http://openspot2/> and <http://openspot2.local/> won't work. Use [SharkRF Link](#) or enter the IP address of the openSPOT2 into the browser to open the web interface.



If <http://openspot2/> and <http://openspot2.local/> do not work on your non-Android device then make sure multicast data is not disabled on your Wi-Fi network. Some Wi-Fi access points and routers have multicast data disabled by default. This is required for LLMNR and MDNS name resolving.

## About the web interface



The openSPOT2's **hostname** is displayed at the right of the header. The device's **IP address** is displayed by hovering the mouse above the hostname. The hostname will be hidden on small displays (mobile phones). In this case, you can get the current IP address on the [Network page](#), [Network settings](#) section. You can also see the device's IP address on the [Network page](#), [Current configuration](#) section.

When the web interface is communicating with the openSPOT2, a **loader animation** (2 pulsating circles) is displayed next to the hostname at the right of the header.

The currently active [configuration profile](#) number and name is displayed at the left of the footer. You can also switch between configuration profiles there. Please read the [Notes on configuration profiles](#) section for more information about configuration profile switching.

The openSPOT2's **status** is displayed at the middle of the footer.

The **Advanced mode** checkbox is at the right of the footer. If this is checked, all advanced mode settings show up. Advanced mode is not needed for normal operation.

## Status page

You can see information about the openSPOT2's current status on this page.

### Call log

The screenshot shows the 'Status' page interface. At the top, there are three buttons: 'Fullscreen', 'Clear call log', and 'Export call log'. Below these are three call log entries, each in a grey bubble. The first entry is from 'Parrot (9990)' to 'HA2NON Norbert (2161005, Hungary)'. The second and third entries are from 'HA2NON (Radio ID: E58Gb, IDs: 2161005 (DMR) 61005 (NXDN), Norbert, Hungary)' to 'ALL'. Each entry includes details like 'End' time and signal quality (BER and L0%).

The call log displays all calls coming from the network, from the modem, or from the openSPOT2 itself (voice announcements). In modes where IDs are used as destination and source IDs (ex. DMR, NXDN) the IDs are matched to the ID database and replaced with callsigns. You can get more information about an ID or callsign if you hover the mouse over it.

Each call log entry has information about the call at the top right corner of the entry bubble. BER is displayed in percent after the letter B, packet loss (in case of network calls) is displayed in percent after the letter L. The call duration is also displayed after a call ends.

You can export all call log entries in CSV format by clicking on the **Export call log** button.

If the **Limit call log lines** is checked (it's checked by default) then only a limited number of call log entries are allowed to be in the call log. This is good for limiting the web browser CPU/memory use.

### Log

The screenshot shows the 'Log' page interface. At the top, there are two checked checkboxes: 'Limit lines' and 'Autoscroll'. To the right are two buttons: 'Clear log' and 'Export log'. Below these is a text area displaying system messages with timestamps, such as '15:01:41 spk-remote: got first packet' and '15:01:41 nvmm-csd: looking up id 9 type 0'.

The device log is only shown if *Advanced mode* is enabled.

You can see all device log messages coming from the openSPOT2 here. All log lines are shown here except lines coming from the Wi-Fi module handler. Use the [USB serial console](#) log if you also want to see these messages.

If **Limit lines** is checked (it's checked by default) then only a limited number of log lines are allowed to be in the log. This is good for limiting the web browser CPU/memory use.

You can export the log lines to a .txt file by clicking on the **Export log** button.

## openSPOT2 info

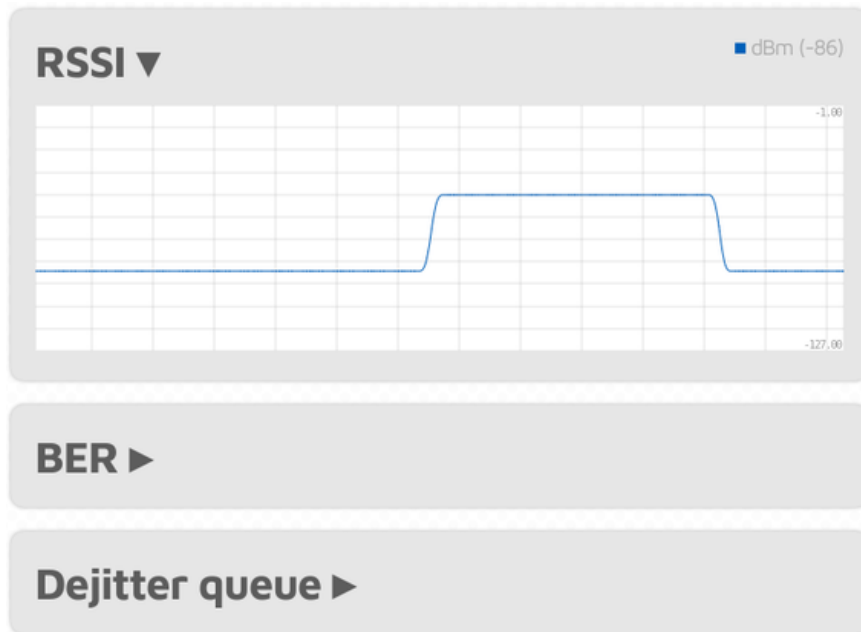
### openSPOT2 info

Hardware version:	1.0
Firmware version:	v9
Callsign database version:	201809251436
Bootloader version:	v5
Device ID:	12345678
USB data:	connected
Uptime:	04:18:01
Device time:	Wed, 10 Oct 2018 15:06:55

You can see basic information about the device in this section.

The **USB data** field shows **connected** when the openSPOT2 is connected to a host with USB enumeration capabilities (like a computer) using a USB cable which supports data transfer, not just charging. The [USB serial console](#) can only be used if this field shows connected. The supplied USB cable supports data transfer.

## Graphs



The **RSSI graph** shows receive signal strength from the modem. The **BER graph** shows received Bit Error Ratio during a call. BER is calculated also for packets received from the network, not just from the modem. The **dejitter queue graph** (only shown if *Advanced mode* is enabled) shows the number of packets waiting in the dejitter queue.

### About the dejitter queue

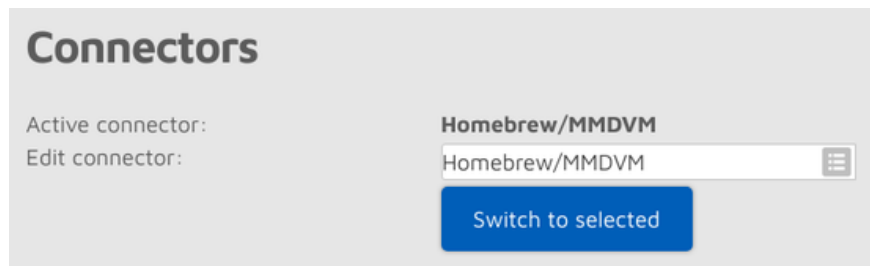
All packets received from the network will first go to this queue to eliminate **network jitter**.

If the dejitter queue **length is too short**, the **graph's line can touch zero** during a call and the **audio can stop for a moment** as the modem runs out of frames to transmit.

If the queue's length is **too long**, the delay can be uncomfortably high.

You can use this graph to set a proper dejitter queue length. There should be at least a few packets in the dejitter queue when receiving a call from the network. You can set the dejitter queue length on the [Network page](#) (the setting only shows up if *Advanced mode* is enabled). As a **general rule**, if you have a good network connection to the selected connector's server, set dejitter queue length to 500ms, and if you have a 3G/4G internet connection or voice stutters a lot, try setting the length to at least 1000ms.

## Connectors page



In openSPOT2 terms, a connector is the **protocol** which you use to connect the openSPOT2 to a server.

The easiest way to choose which connector to select is using the [Quick Setup](#).

You can activate a connector by selecting it from the **Edit connector** list and clicking on the **Switch to selected** button. You can edit a connector's settings by selecting it from the **Edit connector** list. Don't forget to click on the **Save** button after changing a connector's settings.

### Notes on connectors

Each connector has its own assigned **RX/TX frequencies** and **modem modes**. These frequencies and modem modes will be used when the connector becomes activated.

Some connectors have **Add server** and **Remove** buttons. These can be used to add and remove **custom servers** in the server list. Note that these custom servers are stored in the browser's local storage (they won't show up using another browser).

Some connectors have a **Keepalive interval** setting. This means the server requires the openSPOT2 to periodically send **keepalive messages** to the server. The **RX timeout** setting specifies the time after the last received valid network packet needed for the openSPOT2 to consider the connection is dead and it should reconnect. If the openSPOT2 **frequently disconnects** from the server, try increasing the RX timeout.

If a connector has both the **Keepalive interval** and **RX timeout** settings, then make sure the **RX timeout** is set to a greater value than the *\*Keepalive interval*.

The **Connect retry interval** sets the delay between server connect retries.

As a general rule, try to **use a server nearest to your location** to avoid packet loss and high latency.

Please see the [Connectors](#) section for more information about each connector.

### General connector settings

### General settings

Change to Null connector after last call (sec, 0 to disable):

Change after last call from:  Modem  Modem or network

Save

This section is only available if *Advanced mode* is enabled.

By setting the **Change to Null connector after last call** other than 0 will change the active connector to the Null connector after the last received call (minimum timeout value is 5 minutes). For example, if you set a timeout of 600 seconds, openSPOT2 will change to the Null connector if there is no call received in the last 10 minutes.

## Modem page

You can change the modem settings on this page.

If you change the modem mode, RX or TX frequency on this page, the values will be copied to the currently active connector's settings. If *Advanced mode* is turned off, both RX and TX frequencies will be automatically set to the same value (the separate RX and TX frequency fields will be hidden).

If the **Idle modem mode** is activated then the openSPOT2's modem won't transmit and will ignore all received transmissions.

In case you have *half deviation* enabled in your C4FM/Fusion radio (some models don't have this setting and have *half deviation* mode enabled by default), use the **C4FM Half Deviation** modem mode.

### Frequency

#### Frequency

Save

Modem frequency (MHz):

Transmit power:  13 dBm (19.95 mW)

The **Modem receive frequency offset** setting can be used to correct the frequency difference (RX offset) between the openSPOT2 and the used transmitter. This setting is only available if *Advanced mode* is enabled.

Setting an RX offset may be needed in case others report that transmissions coming from your transceiver have voice quality issues. This should rarely happen as the **openSPOT2 has a built-in, factory calibrated, extreme accurate clock reference**. You can also use the [AutoCal](#) special connector to find out the required RX offset value, or you can use the openSPOT2's built-in [local echo service](#) to find out the RX offset value yourself by listening back your recorded voice.

### CW ID

#### CW ID

Save

Enabled

Audible CW ID (enable modulation)

ID to transmit:

Speed (WPM):

Interval (sec):

TX delay (sec):



You can set a CW ID which the openSPOT2 will transmit at the specified interval with the specified speed. You can use letters, numbers, slash (/) and space characters in the ID.

The openSPOT2 transmits an **unmodulated carrier wave** by default, so you'll only be able to hear it using a radio which is capable of CW/SSB demodulation. You can turn on **Audible CW ID**, so the carrier wave will be modulated, and you'll be able to hear it on a regular FM receiver. Note that if **Audible CW ID** is turned on, the openSPOT2 transmits the CW ID without going to RX between dits and dahs, so you won't be able to transmit using your transceiver while the CW ID is transmitted.

Before the openSPOT2 starts to transmit the CW ID, it waits at least **TX delay** seconds after the last received call. An incoming call (both from a radio or from the network) will interrupt the CW ID transmission, so enabling the CW ID does not interfere with the openSPOT2's normal operation.

## Other modem settings

### Modulation Save

Modulation mode:

Inner deviation (Hz):

### Other settings Save

Call hang time (ms):

BCLO level (dBm, 0 - off):

Ignore RX after TX (ms):

Sensitivity level:

Filter gain:

Auto AGC

AGC low threshold (dBm):

AGC high threshold (dBm):

External VCO

Reset modem config to defaults

If *Advanced mode* is enabled, you can tweak various other modem parameters at the bottom of the modem page.

The **Call hang time** setting specifies how many milliseconds the openSPOT2 will wait after the last valid frame has been received to consider a call has been finished with a timeout.

If the **BCLO level** (BCLO - Busy Channel Lock Out) setting is not 0, then the openSPOT2's modem will only start a transmission if it's RSSI meter does not measure a signal strength above the specified BCLO value in dBm.

The **Ignore RX after TX** setting specifies a time for which all received frames by the modem will be ignored after the last TX has ended. This is useful if you want to use the openSPOT2 to link a repeater to the internet through RF and the repeater has a lengthy internal buffer and it is still transmitting even after the openSPOT2 has finished transmitting (which means the openSPOT2 will start receiving it's own stream again).

The modem's **Sensitivity level** can be changed in 6 levels. Note that increasing sensitivity decreases the linearity of the filters, which may result in incorrectly demodulated bits (which means increased BER). By default, sensitivity is low (level set to 5 or 6) for modem modes using 4FSK.

The sensitivity of the modem's filter can be changed with the **Filter gain** setting. Note that increasing sensitivity decreases the linearity of the filters, which may result in incorrectly demodulated bits (which means increased BER). By default, filter gain is set to 8.

The modem's AGC (Auto Gain Control) is configured by software for optimal values for low BER in 4FSK demodulation modes. However, you can turn automatic AGC on with the **Auto AGC** setting, but it is recommended to keep this checkbox off so the openSPOT2's AGC can work on it's own.

When the AGC is turned on, you can set the **AGC low/high threshold** values. These values will be considered as low and high thresholds for the Auto AGC circuitry in the modem.

The **External VCO** checkbox should be checked when an external VCO is applied to the openSPOT2. Currently there's no external VCO used by the openSPOT2. Note that applying an external VCO is considered as an after-market modification and voids the warranty of the openSPOT2.

You can also reset all modem settings back to defaults by clicking on the **Reset modem config back to defaults** button.

All of the settings above are stored independently for each modem mode in the current profile.

## Settings page

You can change the general settings of the openSPOT2 on this page.

### Configuration profile

#### Configuration profile

**Change active profile** Change

Active profile slot: **1 (default)**

Change to profile slot:  ⋮

**Profile names** Save

Profile #1:

Profile #2:

Profile #3:

Profile #4:

Profile #5:

**Copy** Copy

Source:  ⋮

Destination:  ⋮

**Export/import file** Export Import

Profile:  ⋮

Include passwords in export

Status:

Progress:

You can use 5 different configuration profile slots. You can switch between them on the web interface, or by using your radio (see the [Control calls](#) section for more information).

You can rename profiles, copy one to another slot or export/import to/from a file.

By setting the **Change after last call** setting to non-zero (this setting is available if *Advanced mode* is enabled), the openSPOT2 will switch to the specified profile slot after a timeout since the last call was received.

The **Don't show init setup on login** checkbox is checked if the [Initialization setup](#) has been completed.

### Notes on configuration profiles

- Please keep in mind that all configuration profiles are **empty by default**, so if you switch to a previously unused profile, **the openSPOT2 will start in Wi-Fi AP mode** as there will be no configured Wi-Fi networks to connect.
- The openSPOT2 **reboots** on profile switch.
- Wi-Fi networks and passwords are stored for each profile separately.
- The web interface password is stored for each profile separately.

### Firmware upgrade

#### Firmware upgrade

Save  
 Automatic upgrade to latest version  
 Automatic upgrade data files  
Auto upgrade at (hh:mm):

The openSPOT2 will **automatically** check for firmware and data upgrades periodically, as specified by the **Check interval** setting (only available if *Advanced mode* is enabled). If **Automatic upgrade** settings are checked then the openSPOT2 will **automatically download** upgrades.

Data upgrades are applied immediately, but upgrading the firmware requires a reboot to the bootloader. If a new firmware has been downloaded, a reboot will be automatically scheduled for the (local) time set by the **Auto upgrade at** setting. By default it's set to 4am next day. The reboot will only be performed if there were no calls received for at least **Auto upgrade RX/TX delay** seconds (this setting is only available if *Advanced mode* is enabled).

**Do NOT unplug the openSPOT2 during the firmware upgrade!**

### SharkRF Network

#### SharkRF Network

Device login

You can log your device in to the SharkRF Network. Currently only developers and alpha testers have access to the network's features. Eventually, these features will be available for all openSPOT2 users.

### Voice announcements

### Voice announcements

Enabled

Save

The openSPOT2 has an own, built-in voice announcement system. Some announcements are played locally from the openSPOT2's built-in flash memory, but some are played using our open source [voice announcement server](#).

Modem mode notes:

- **DMR:** voice announcements are sent to TG9, so to hear them, make sure you have TG9 in your radio's RX group list.
- **NXDN:** voice announcements are sent to TG0.
- **P25:** voice announcements are sent to TG9.

If you enable *Advanced mode*, the following settings will show up:

- You can shorten profile announcements by checking the **Shortened profile announcement** checkbox.
- If you don't want to hear the linked reflector/talkgroups list, then you can shorten BrandMeister announcements by checking the **Shortened BrandMeister announcement** checkbox.
- You can disable built-in voice announcements by checking the **Use server only** checkbox. Note that startup voice announcements played right after the openSPOT2 boots will still be played from the openSPOT2's flash memory.
- You can set your own voice announcement server's host and port.
- **Net state announcement int.** sets the interval of the announcement if the openSPOT2 can't connect to a Wi-Fi network.

See the [Control calls](#) section for more information about [Profile query/change base ID](#), [Connector query ID](#) and [IP address query ID](#).

## Location settings

### Location settings

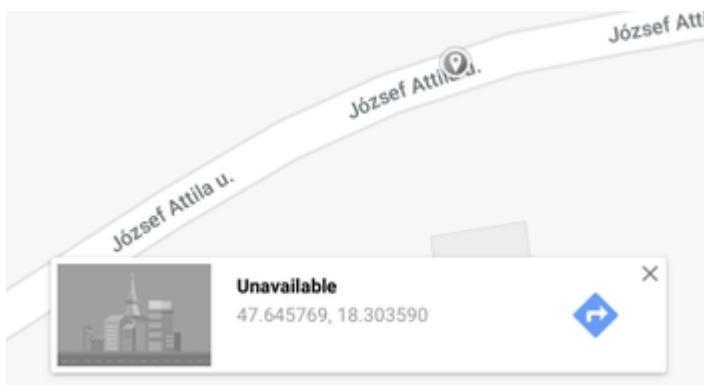
Click [here](#) if you want to set up sending the device location to the APRS network.

QTH locator ( <a href="#">map</a> ):	<input type="text" value="JN97MM14fi"/>
Latitude (decimal degrees):	<input type="text" value="+47.518142"/>
Longitude (decimal degrees):	<input type="text" value="+19.01024"/>
Height (AGL, meters):	<input type="text" value="13"/>
Height (ASL, meters):	<input type="text" value="155"/>
Description (city, country):	<input type="text"/>

Save

Setting location data is optional. It is used by some connectors like the [Homebrew/MMDVM](#) to report the location of the openSPOT2 to the connector's network.

Latitude and longitude is in decimal degrees. [Google Maps](#) will display a location's coordinates in decimal degrees if you click on it:



Both height values refer to the height of your openSPOT2 (they are used separately).

## DMR settings

### DMR settings Save

Color code (CC):

Default cross mode source DMR ID:

The modem will only process DMR calls with a matching **Color code**.

If the source callsign of a cross mode call has no associated DMR ID, and it can't be parsed into a number, then the **Default cross mode source DMR ID** will be used for the outgoing DMR call. If you are the only person using the openSPOT2, **set this to your DMR ID**, and then you don't need to change the callsign of your radio to your DMR ID.

If *Advanced mode* is enabled, the following settings show up:

If a call is received by the modem to the **Echo ID**, then the openSPOT2's built-in [local echo service](#) gets activated. It will record the call and plays it back after the call is finished. Both private and group calls are answered with a private echo call reply.

If the **Allow only IDs as cross mode callsigns** setting is enabled, and the source callsign of a cross mode call has no associated ID in the database, and also it can't be parsed into a DMR ID, then the call will be ignored.

If the **Force source ID from modem** is set to other than 0, the source DMR ID of all voice calls received by the modem will be replaced with this ID.

If the **Send no in-band data to network** setting is enabled, then in-band data coming from the modem (like the talker alias or in-band GPS position information) won't be sent to the network.

If the **Force talker alias to DMR network** is set, then this talker alias will be sent for all calls coming from the modem.

If the **Send no in-band data to modem** setting is enabled, then the openSPOT2 will remove all in-band data from calls coming from the network. By default this setting is enabled. If you have a radio which supports **talker alias** decoding (like Hyteras with at least firmware version v8), **turn off** this setting to let the openSPOT2 pass through in-band data to your radio. If you have a radio which does not support talker alias decoding, **turn on** this setting to prevent issues with decoding voice calls containing in-band data (voice stuttering, calls dropping etc.).

## D-STAR settings

**D-STAR settings** Save

Echo callsign:

- Transmit RX confirmation
- Auto mute DTMF commands
- Ignore DTMF commands after 10 secs

This section is only available if *Advanced mode* is enabled.

If a call is received by the modem to the **Echo callsign**, then the openSPOT2's built-in [local echo service](#) gets activated. It will record the call and plays it back after the call is finished.

If the **Transmit RX confirmation** checkbox is checked, the openSPOT2 will transmit a short call confirmation after the PTT is released on the D-STAR radio.

If the **Auto mute DTMF commands** checkbox is checked and a call is ongoing from the modem, and a DTMF sequence is being entered, then the call to the network will be automatically ended, so others won't be annoyed by hearing your DTMF tones on the same reflector/gateway.

If the **Ignore DTMF commands after 10 secs** checkbox is checked, then all DTMF commands will be ignored after 10 seconds have elapsed since call start. This is useful to avoid any mistakenly entered and decoded DTMF commands.

## C4FM/Fusion settings

### C4FM settings Save

Allow data calls to network  
 Ignore Wires-X button commands  
 Don't convert Wires-X server names to uppercase  
 Don't send automatic Wires-X connect messages  
 Hide profiles in Wires-X "all" reply  
 Auto mute DTMF commands  
 Ignore DTMF commands after 10 secs

Special command/cross mode private call DTMF code:

Cross mode group call DTMF code:

Transmit RX confirmation

Default callsign for cross mode calls:

Echo DGID:

Only receive DGID from modem

DGID from modem:

Replace DGID to modem enabled

Replace DGID to modem:

Replace DGID to net enabled

Replace DGID to net:

This section is only available if *Advanced mode* is enabled.

- **Allow data calls to network:** if enabled, data calls will be forwarded from the modem to the network. This is not required for normal operation.
- **Ignore Wires-X button commands:** Wires-X button commands received by the modem will be ignored in case this settings is enabled.
- **Do not convert Wires-X server names to uppercase:** all server names sent to the transceiver are converted to uppercase by default, and space characters are replaced with dashes. This conversion can be turned off with this setting. The conversion is done because some radios (like the Yaesu FT70) do not show lowercase characters on their displays, and space characters prevent the radio from saving the current server to a category.
- **Don't send automatic Wires-X connect messages:** the openSPOT2 automatically sends Wires-X connect messages in case it connects or disconnects to/from a server. These automatic messages can be disabled with this setting.
- **Hide profiles in Wires-X "all" reply:** if this setting is enabled, configuration profiles won't be listed on the first page for a Wires-X "all" server search.
- **Auto mute DTMF commands:** if a call is ongoing from the modem, and a DTMF sequence is being entered, and the first character is a **Special command / cross mode private call** or **Cross mode group call** DTMF code, the outgoing call to the network will be automatically ended, so others won't be annoyed by hearing your DTMF tones on the same reflector/talkgroup.
- **Ignore DTMF commands after 10 secs:** if this checkbox is checked, then all DTMF commands will be ignored after 10 seconds have elapsed since call



start. This is useful to avoid any mistakenly entered and decoded DTMF commands.

- **Special command/cross mode private call DTMF code:** this code is used as the first character for entering a control DTMF code, or starting a private call in cross modem modes. See more info at each [connector's description](#), or at the [Control calls](#) section.
- **Cross mode group call DTMF code:** if a connector is active which supports group calls and C4FM cross mode, then this code can be used to start group calls to the ID entered after this code.
- If **Transmit RX confirmation** is checked, then the openSPOT2 will transmit a short call confirmation after the PTT is released on the C4FM radio.
- **Default callsign for cross mode calls:** this callsign will be used on C4FM networks for calls coming from a radio which does not have a callsign (ex. DMR, NXDN ID) instead of using the IDs as callsigns.

Note that if the incoming call's talker alias is decoded, then it will be used instead of this default callsign setting.

- If a call is received by the modem to the **Echo DGID**, then the openSPOT2's built-in [local echo service](#) gets activated. It will record the call and plays it back after the call is finished.
- If the **Only receive DGID from modem** checkbox is checked, then calls only to the specified **DGID from modem** will be processed by the modem.
- If the **Replace DGID to modem enabled** checkbox is checked, then the DGID in all frames received from the network will be replaced with the specified *Replace DGID to modem* value.
- If the **Replace DGID to net enabled** checkbox is checked, then the DGID in all frames received from the modem will be replaced with the specified **Replace DGID to net** value.

## NXDN settings

**NXDN settings**
Save

Radio Access Number (RAN):

Default cross mode source ID:

- The modem will only process NXDN calls with a matching **Radio Access Number** if it is set other than 0.
- If a call is coming from a mode which does not have ID information (ex. C4FM), or the source ID is outside the NXDN ID range (0-65535), then the **Default cross mode source ID** will be used as the source ID for the call. This ID will always be used if the **Always use this ID in cross mode** checkbox is checked.

If you use a non-NXDN network with your NXDN radio, you can set this to your NXDN ID.

If *Advanced mode* is enabled, the following settings show up:

- If a call is received by the modem to the **Echo ID**, then the openSPOT2's built-in [local echo service](#) gets activated. It will record the call and plays it back after the call is finished. Both private and group calls are answered with a private echo call reply.
- If a call is coming from a mode which does not have ID information (ex. C4FM), or the destination ID is outside the NXDN ID range (0-65535), then the **Default cross mode destination ID** will be used as the destination ID for the call. This ID will always be used if the **Always use this ID in cross mode** checkbox is checked.
- If the **Allow only IDs as cross mode callsigns** setting is enabled, and the source callsign of a cross mode call has no associated ID in the database, and also it can't be parsed into an NXDN ID, then the call will be ignored.

## P25 settings

### P25 settings

Save

Network Access Code (NAC):

Default cross mode source P25 ID:

The modem will only process P25 calls with a matching **Network Access Code (NAC)**. Note that the **openSPOT2 requires the NAC in decimal format** (example: if you have the NAC 0x293 in your radio then you have to enter 659 in the openSPOT2).

If the source callsign of a cross mode call has no associated DMR/P25/CCS7 ID, and it can't be parsed into a number, then the **Default cross mode source P25 ID** will be used for the outgoing P25 call. If you are the only person using the openSPOT2, **set this to your DMR/P25/CCS7 ID**, and then you don't need to change the callsign of your radio to your DMR/P25/CCS7 ID.

If *Advanced mode* is enabled, the following settings show up:

If a call is received by the modem to the **Echo ID**, then the openSPOT2's built-in [local echo service](#) gets activated. It will record the call and plays it back after the call is finished. Both private and group calls are answered with a private echo call reply.

If the **Allow only IDs as cross mode callsigns** setting is enabled, and the source callsign of a cross mode call has no associated ID in the database, and also it can't be parsed into a DMR/P25/CCS7 ID, then the call will be ignored.

## POCSAG settings

### POCSAG settings

[Reset frequency](#) [Save](#)

POCSAG frequency (MHz):

Bitrate:

TX delay in other modem modes (sec):

#### Allowed timeslots

Do not allow DAPNET to override timeslots

<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input checked="" type="checkbox"/> 4	<input checked="" type="checkbox"/> 5	<input checked="" type="checkbox"/> 6	<input checked="" type="checkbox"/> 7
<input type="checkbox"/> 8	<input type="checkbox"/> 9	<input type="checkbox"/> A	<input type="checkbox"/> B	<input checked="" type="checkbox"/> C	<input checked="" type="checkbox"/> D	<input checked="" type="checkbox"/> E	<input checked="" type="checkbox"/> F

This section is only available if *Advanced mode* is enabled.

If the modem is transmitting POCSAG messages then it will switch to the **POCSAG frequency**. The **TX delay in other modem modes** setting specifies how many seconds the openSPOT2 should wait after the last call has been finished to switch to POCSAG mode and start transmitting the queued messages.

There are 16 timeslots for POCSAG, these timeslot settings can be overridden by the DAPNET server when the DAPNET connector is connected unless the **Do not allow DAPNET to override timeslots** checkbox is checked. Each timeslot is 6.4 seconds long. The openSPOT2 will only start transmitting POCSAG messages if the current timeslot is allowed.

## Lock settings

### Lock settings

[Save](#)

ID #1 (0 to disable):

ID #2 (0 to disable):

ID #3 (0 to disable):

Callsign #1 (empty to disable):

Callsign #2 (empty to disable):

Callsign #3 (empty to disable):

This section is only available if *Advanced mode* is enabled.

These settings allows you to lock the openSPOT2 to the given IDs and/or callsigns. If lock is enabled, the openSPOT2's modem will only allow calls originating from the given IDs or callsigns.

## Miscellaneous settings



If *Advanced mode* is enabled, you can set the openSPOT2's LED brightness with the slider. Please note that the new brightness setting will be only applied by the openSPOT2 at the next LED state change.

There are also some buttons for reboot and config profile reset actions.

## Network page

You can change the network settings of the openSPOT2 on this page.

The screenshot displays the 'Network status' section of the openSPOT2 web interface. It features a blue button labeled 'Switch wireless mode' in the top right corner. The main content is organized into two sections: 'Current network status' and 'Current configuration'.

Current network status	
Current wireless mode:	wireless client
Status:	connected to bombegy (6c:3b:6b:45:9f:a9, channel 6)
RSSI:	-55 dBm ▶
Internet connection quality:	● (ok)
Wireless regulatory domain:	FCC

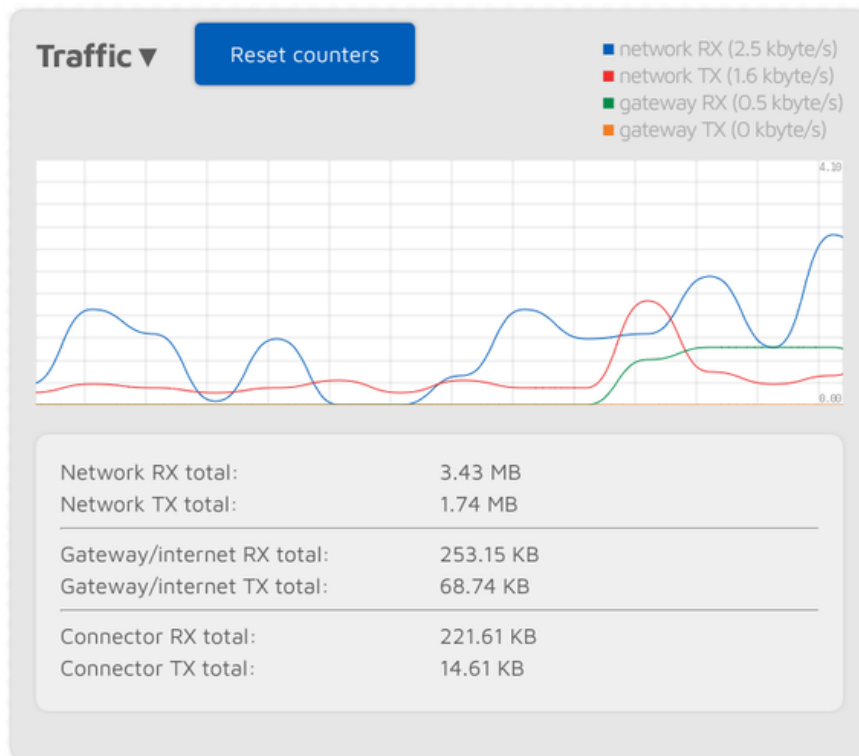
  

Current configuration	
Mode:	DHCP
IP address:	192.168.3.106
Netmask:	255.255.255.0
Gateway:	192.168.3.1
DNS server #1:	192.168.3.1
DNS server #2:	not set

The **Network status** section shows current network status. The **Wi-Fi RSSI graph** can be shown by clicking on the little arrow next to the RSSI value. The **Internet connection quality** is determined by the openSPOT2 by periodically measuring the network latency. You can get the measured latency value by hovering the mouse above the quality display circle. The **Wireless regulatory domain** is set by the country of operation.

The openSPOT2 can be switched between AP and normal modes by clicking on the **Switch wireless mode** button. This is the same as pressing the [button](#) for 3 seconds.

## Network traffic



The **traffic graph** can be shown by clicking on the arrow at the top of the Traffic section. The **Network RX/TX total** shows total Wi-Fi traffic of the openSPOT2, the **Gateway/internet RX/TX total** shows the traffic coming/going from/to the gateway (which is usually the internet traffic consumed by the device), and the **Connector RX/TX total** shows the traffic used by the currently active connector.

The openSPOT2 uses very little internet traffic. Depending on the active connector, the data rate estimates are as follows:

- Quiet channel is 36-180 kB / hour
- Active channel is 3-6 MB / hour

## Network settings

The figure shows the 'Network settings' configuration form. It includes a 'Clear password' button and a 'Save' button. The form contains the following fields:

- Country of operation: Hungary
- Change web int. password: (show)
- Hostname: openspot2

The web interface of the openSPOT2 is not password protected by default, which means anyone on the local network can open it. You can set a password by entering it in the **Change web int. password** input field. To reset the original behavior and clear the password, click on the **Clear password** button.

The **Hostname** is used by openSPOT2's built-in LLMNR and MDNS server which means you'll be able to open the web interface by entering <http://hostname/> or <http://hostname.local/> into the browser. The hostname is also displayed at the

top right corner of the web interface. If you have **multiple openSPOT2s on the same network**, make sure you use different hostnames for each of them.

The following settings are only available if *Advanced mode* is enabled.

Please see the [About the dejitter queue](#) section of the user manual for more information on the **Dejitter queue length** setting.

The openSPOT2 checks for internet connection availability and quality periodically. The interval of these checks is set by the **Internet conn. check interval** setting.

The openSPOT2 can send its log to a syslog server set by the **Syslog server** field. Check the **Send log to syslog server** checkbox to enable using the syslog server.

Please see the [Powersaving](#) section of the user manual for more information about the **Power saving** feature.

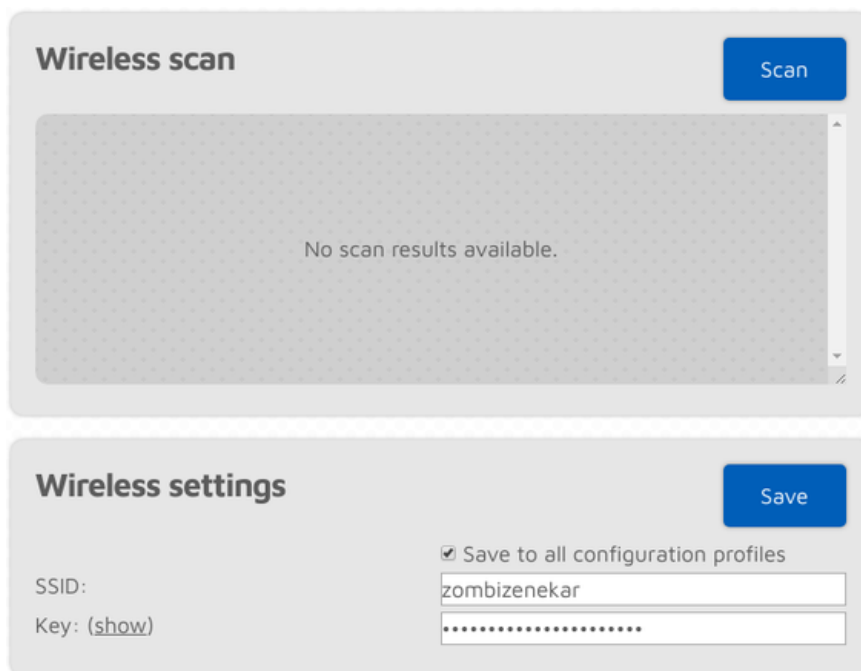
## NTP settings

This section is only available if *Advanced mode* is enabled.

The openSPOT2 synchronizes date and time using the Network Time Protocol (NTP). By default, the NTP server advertised by the Wi-Fi network's DHCP server is used for synchronizing, but if the DHCP server does not advertise an NTP server, the host set at the **NTP server** setting will be used.

**Note:** The web interface loads the time from the openSPOT2 only **when the web interface is opened**. After that, the browser is updating the displayed clock and if the browser, or the web interface's browser tab is sent to the background then the time will be updated more irregularly due to the browser's CPU power saving mechanism. This causes the displayed openSPOT2 device time to be incorrect, but **this does not affect the openSPOT2's clock, it will stay correct**. Reload the web interface to make the displayed time accurate again.

## Wireless settings



If *Advanced mode* is enabled, you can define **up to 5 Wi-Fi networks (per configuration profile)** to connect. If the first configured Wi-Fi network is not available, then the openSPOT2 will try connecting to the second one. If it's not available either, then it'll try the third one and so on. After trying connecting to all of the configured networks, it restarts the sequence with the first network.

If *Advanced mode* is enabled, the *Wireless scan* list will have a **Save** button for all previously unsaved networks instead of the **Connect** button. Clicking on the **Save** button lets you easily save the network to one of the 5 available Wi-Fi network slots.

If you are manually adding a connection to an iPhone's shared internet connection then make sure you **copy and paste** the phone's Wi-Fi SSID as the iPhone uses a special apostrophe character which looks nearly the same as the one you can enter with your keyboard, but it's different, and all characters need to match.

If the **Save to all configuration profiles** is active then the configured Wi-Fi networks will be saved to all profiles and will be common between profiles.

If no networks are configured then the openSPOT2 **switches to access point (AP) mode** and starts broadcasting it's own Wi-Fi network. This is indicated by the LED:



In this case, complete the steps described in the [Initialization setup](#) section of the user manual to set a Wi-Fi network for the openSPOT2.



The AP mode SSID can be set in the **AP SSID** input field. By default there's no Wi-Fi password set for the openSPOT2's AP mode, but you can set one at the **AP key** input field. The AP mode Wi-Fi channel can be set in the **Channel number** input field. If the **Always start in AP mode** checkbox is checked, then the openSPOT2 will always switch to AP mode after powered on even if there are configured Wi-Fi networks in the current configuration profile.

If a BSSID is configured for a network then it'll be used instead of the SSID. WEP encryption is also supported if the network key length is 5 or 13 characters.

## IP settings

**IP settings** Save

IP configuration mode:  ⋮

Override DHCP DNS servers

This section is only available if *Advanced mode* is enabled.

The openSPOT2 has two IP configuration modes:

- **DHCP:** This is the default mode. The device will use the Wi-Fi network's DHCP server for acquiring an IP address.
- **Static:** a static IP address can be set for the openSPOT2. This is only recommended for advanced users.

The DNS server settings which are advertised by the DHCP server can be overridden if the **Override DHCP DNS servers** checkbox is checked. If the openSPOT2 can't connect to a server (for ex. the openSPOT2 shows connector connecting after switching D-STAR reflectors from your radio, but it can connect using the web interface otherwise), you can try overriding the DNS servers to 8.8.8.8 and 8.8.4.4.

## MAC settings

**MAC settings** Clone Restore Save

Current MAC address:

Browser client MAC address: d4:81:d7:e3:b7:7f

Factory MAC address: 3c:a0:67:58:62:66

AP MAC address: 3c:a0:67:58:62:67

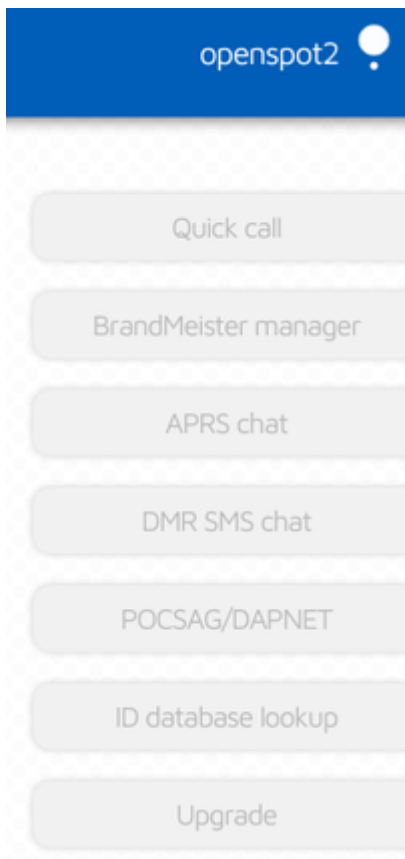
This section is only available if *Advanced mode* is enabled.

You can change/clone the Wi-Fi MAC address of the openSPOT2 here. This is useful if you try to use a Wi-Fi network which uses a captive portal for authentication (example: hotel or airport networks). In this case, do the following steps:

- Connect to the Wi-Fi network with your phone/tablet/computer, log in on the captive portal to enable the internet access for your phone/tablet/computer's MAC address.
- Switch the openSPOT2 to access point (AP) mode by holding the [button](#) for 3 seconds.
- Connect your phone/tablet/computer to the **openSPOT2 AP**, but exit the [Initialization setup](#) and open the Network page.
- Set the captive portal Wi-Fi network's SSID and key at the **Wireless settings** section.
- Click on the **Clone** button at the MAC settings section (make sure **Advanced mode** is enabled so the MAC settings section is visible).
- Switch the openSPOT2 back to normal mode by holding the [button](#) for 3 seconds, or by clicking the **Switch wireless mode** button at the top of the Network page. The openSPOT2 will connect to the captive portal Wi-Fi network with your phone/tablet/computer's already logged in MAC address.

Please keep in mind that **MAC cloning may cause packet loss** and the network connection of both devices may get **periodically interrupted**.

## Side panels / Tools page



If you are using a computer, or a screen width enough width, the Tools page gets hidden and side panels will show up on the web interface. On small screens they are hidden, and their contents can be found on the Tools page.

### Quick call

**Quick call** Private call 4000 Group call 4000 Call

Destination ID: 216 (Hungary) 216

Group call  Private call

**Shortcuts** Clear shortcut Save as shortcut

Empty	Empty	Empty	Empty
Empty	Empty	Empty	Empty
Empty	Empty	Empty	Empty

The quick call feature is available when the [Homebrew/MMDVM connector](#) is active.

Reflector and talkgroup linking (and unlinking) is **done by the network, not the openSPOT2**. The openSPOT2 has to make quick calls, which are the same as pressing PTT for a short time on a radio (kerchunking). These empty calls will also show up on the BrandMeister Network's dashboard.

Quick call is useful for **linking/unlinking talkgroups and reflectors**. Starting a short private call to a reflector ID (between 4000 and 5000) will link the reflector. Starting a short group call to a talkgroup ID will link that TG. On the BrandMeister network, reflectors are unlinked by starting a **private** call to ID 4000, and both reflectors and talkgroups are unlinked by starting a **group** call to ID 4000.

Quick call shortcuts can be stored in the openSPOT2's current configuration profile. If you hover the mouse cursor over a saved talkgroup's button, or the button is pressed for a few seconds then the (BrandMeister) talkgroup's name will appear.

If the **Auto group call 4000 before shortcut call** checkbox is checked, then the openSPOT2 will automatically start a quick call to TG4000 to unlink everything before quick calling the desired ID.

## BrandMeister Manager

## BrandMeister manager

### API key

API key:

Get key
Check key
Save

### Actions

Drop current call

### Static talkgroups

Add

1 (Local)
⋮

Add static talkgroup ID:

Currently linked static TGs: 216 ✕, 3100 ✕

### Dynamic talkgroups

Drop all
Quick call

1 (Local)
⋮

Quick call talkgroup ID:

Currently linked dynamic TGs: -

### Reflector

Save

4000 (Disconnect)
⋮

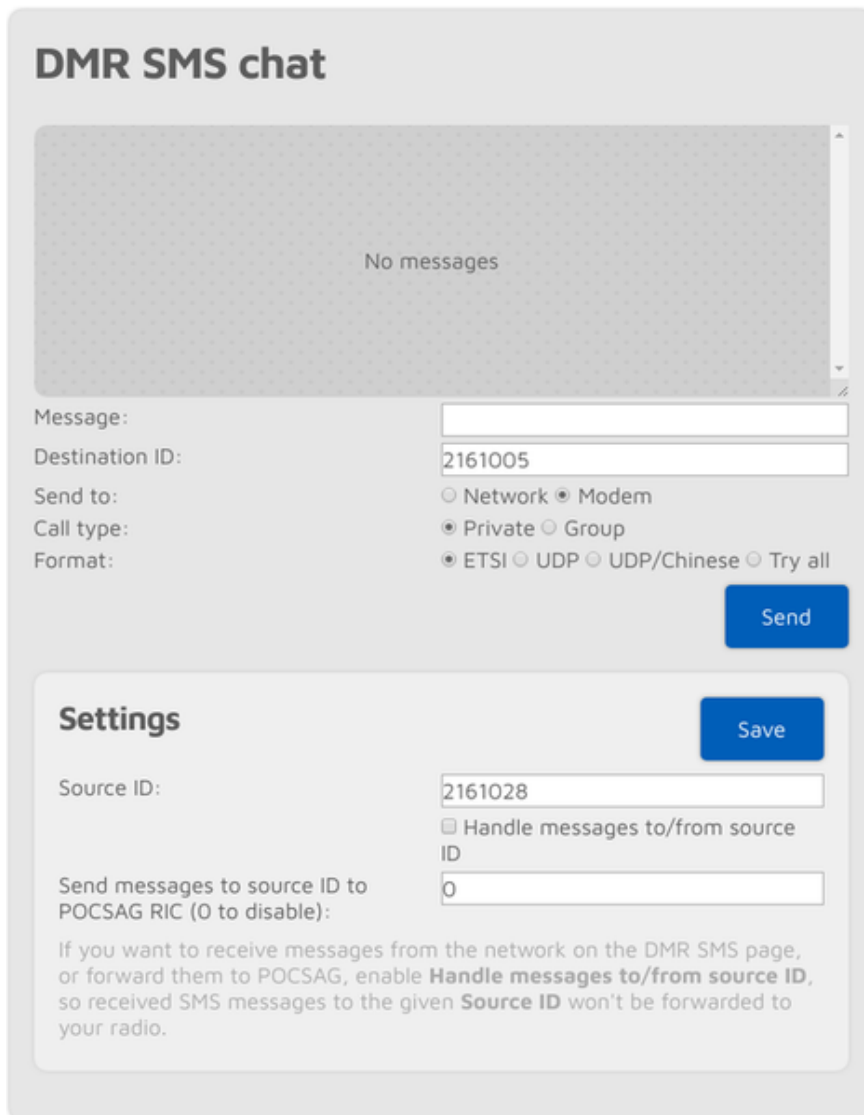
Active ID (4000 - not linked):

The BrandMeister Manager (BMM) is available when the Homebrew connector is active, and it is configured for a BrandMeister server. With BMM, you can link/unlink **static** and **dynamic** talkgroups, and you can interrupt the currently ongoing call.

A BrandMeister **API key** is needed to use the BMM. You can get the API key by clicking on the **Get key** button, and logging into the [BrandMeister SelfCare](#). Copy and paste the API key to the **API key** input field and click on the **Save** button. The API key's validity can be checked by clicking on the **Check key** button.

Note that only BrandMeister API v1 keys are stored in the openSPOT2. v2 keys are stored by your browser.

## DMR SMS chat



This feature is available when the modem is in DMR mode, or the currently active connector supports DMR SMS sending.

Sent messages are displayed on the right, received messages are displayed on the left of the chat box. You can click on each message for additional information.

The modem radio button is disabled if the modem is not in DMR mode. Messages sent to openSPOT2's [DMR echo ID](#) (it's 9999 by default) will be echoed back to the sender. Note that the openSPOT2 only stores the last received message, so you have to keep open this page in your browser to receive more than one message at once.

Use ETSI format if you want to send a message to Hytera radios, UDP format for Motorola radios, and UDP/Chinese format for other Chinese radios (Tyt, Retevis, etc.).

If you want to receive messages from the network on the DMR SMS page, you have to check the **Handle messages to/from source ID** checkbox. If it's enabled, received SMS messages to the given **Source ID** won't be forwarded to your radio.

If you want to send received DMR SMS messages as POCSAG messages, check the **Handle messages to/from source ID** checkbox, and set a **POCSAG RIC** to send messages to.

You can simply use the ETSI format if you want to send messages through the network. Note that the network must know about the location of your **Source ID**. If you want to receive DMR SMS messages from the network, you have to set the **Source ID** to the DMR ID you set at the connector settings, or if you want to use another ID, then you have to start a short call from that DMR ID with your radio, so the network will know that your custom DMR ID can be reached through this openSPOT2.

## POCSAG/DAPNET

The screenshot displays the POCSAG/DAPNET interface. At the top, there is a header "POCSAG/DAPNET". Below it is a message queue with three items: "Garage Strom an" (RIC: 144475, 11:49:55), "Garage Strom an" (RIC: 260789, 11:49:55), and "ha2non" (RIC: 8, 11:49:59). The queue is currently empty, and the status is "Waiting for timeslot 4... (6 seconds)".

State: Waiting for timeslot 4... (6 seconds)  
Current timeslot: 3 (Tue, 20 Nov 2018 11:50:08)

Send to:  DAPNET  Modem  
Message:   
Destination callsign: HA2NON  
Destination transmitter group: Hungary (ha-all)   
Type:  Message  Alert message  Alert

Buttons: Clear queue, Send

**DAPNET login information**

DAPNET callsign: HA2NON  
DAPNET user password: (show) .....

Use your DAPNET user (not transmitter) callsign and password key here. These are required for sending messages to DAPNET.

You can see the POCSAG message queue here. Queued unsent messages are displayed with a blue background, sent messages are gray. All messages received from DAPNET or sent from this interface are first placed in the POCSAG message queue.

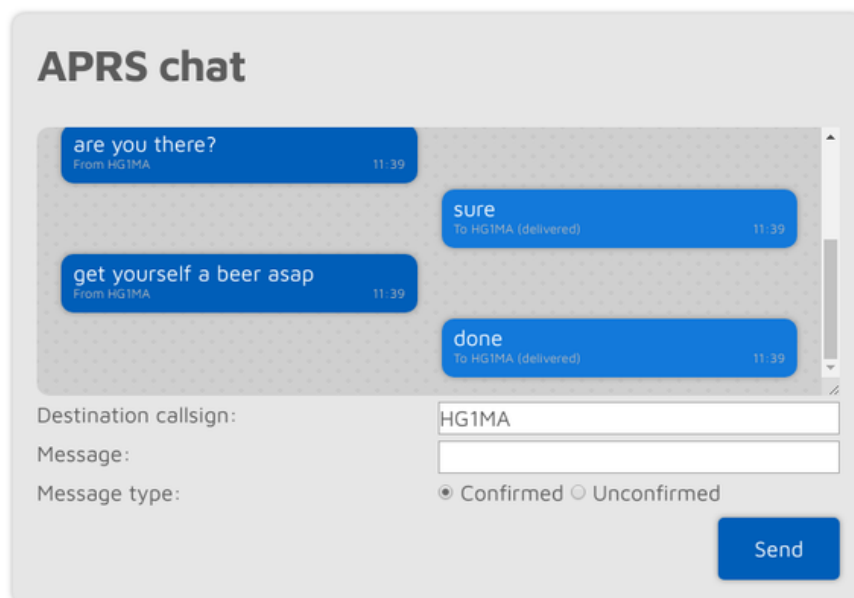
Note that the displayed current timeslot number and time is calculated by the browser and is not equal to the openSPOT2's internal device time and timeslot framing which is more precise.

If the **Send to** radio button is set to Modem then you can directly add messages to the queue which will be transmitted by the openSPOT2.

If you want to send a message to DAPNET, you have to set your DAPNET **user** callsign and user password first. Note that these have to be the DAPNET **user** callsign and password, **not** the transmitter callsign and auth key you've set for the DAPNET connector. Multiple destination callsigns and transmitter groups can also be specified in a comma separated list.

DAPNET messages are sent by your browser to the DAPNET API. Note that sometimes the DAPNET API server does not respond to message send requests and times out, but correctly processes received messages.

## APRS chat



You can send and receive messages via the APRS network. This feature is available when the [APRS connector](#) is enabled.

Sent messages are displayed on the right, received messages are displayed on the left of the chat box. The openSPOT2 stores the last 10 messages in its internal memory, so you can receive messages even if the web interface is not open. You'll get a notification about unread messages if you open the web interface. If you reboot the device then all stored messages will be lost.

Confirmed messages need to be acknowledged by the recipient. If the recipient is unavailable (which means no acknowledge is received for a specific time), then the openSPOT2 retries sending a few times before the message send times out. You can cancel the confirmed message send by clicking on the Cancel button.



If a message bubble is clicked then its callsign will be set as the message send destination callsign.

## ID database lookup

### ID database lookup Lookup

ID or callsign:

Preferred ID type:  DMR/P25/CCS7  NXDN

#### Results Close

DMR/P25/CCS7 IDs:	2161005, 2161028
NXDN IDs:	61005
Callsign:	HA2NON
Name:	Norbert
City:	N/A
State:	N/A
Country:	Hungary

You can look up IDs or callsigns and display information about them. The lookup will be done through the network if the openSPOT2 has not downloaded a valid ID database.

If the **Auto match ID and callsign fields** checkbox is checked, then all callsigns and IDs will be matched when saving connector settings, and a warning will be displayed if the entered callsign does not match the entered ID according to the database. This is to prevent typos or invalid IDs when trying to connect to a network, as most servers do not allow connections with invalid IDs or callsigns.

If the **Don't look up group calls** checkbox is checked, then the openSPOT2 won't display names for group IDs in the call log.

## Upgrade

### Upgrade ★ Upgrade data Upgrade firmware Check

State: Firmware and data upgrades available!

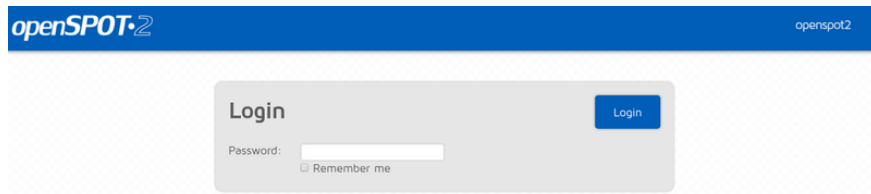
#### Available upgrades

Device firmware:	v9 ( <a href="#">changelog</a> ) <small>(stable version, 1.16 MB)</small>
Data / callsign database:	201809251436 <small>(stable version, 3.14 MB)</small>

Information about firmware and data upgrades are displayed here. You can also request an immediate upgrade availability check, and manually start the upgrade process if auto upgrades are disabled on the Settings page, [Firmware upgrade](#) section.

**Do NOT unplug the openSPOT2 during a firmware upgrade!**

## Login page



This is the page where you can log in to the web interface. The openSPOT2 does not have a web interface password by default, so the login page is not displayed initially. You can set a login password on the Network page, [Network settings](#) section.

If you check the **Remember me** checkbox, the openSPOT2 will not ask for a login password until the device gets rebooted, or the browser forgets the login cookie.

## Null connector

### Null connector

Save

Modem frequency (MHz):

Modem mode:

The Null connector is a special connector - as it's name says - it does not connect anywhere. This is the default active connector when the openSPOT2 is unconfigured.

## Homebrew/MMDVM connector

**DMR/Homebrew/MMDVM**
Save

Modem frequency (MHz):   
 Modem mode:

Protocol:  Homebrew  MMDVM  
 Server:   
 Callsign:   
 DMR ID:   
 Server password: ([show](#))   

BM hotspot security settings

Route cross mode calls to ID:   
 Group call  Private call

You can use this connector to connect to a network which supports the **Homebrew** or the **MMDVM** protocol, like BrandMeister, DMRplus, Phoenix, DMR-MARC, XLX.

The easiest way is to use the [Quick setup](#) to connect to these networks.

This connector supports 2 protocols: **Homebrew** and **MMDVM**. Use the Homebrew protocol to connect to BrandMeister network servers, and the MMDVM protocol to connect to DMRplus, Phoenix, DMR-MARC or XLX servers.

The default server password for BrandMeister servers is *passwOrd* or *password*, but some servers do not use these defaults. In this case take a look at the [BrandMeister wiki](#), or ask the server's admin for the password if the openSPOT2 won't connect with the default one.

If you can't connect to a BrandMeister server because the **authentication has failed**, then turn off **Hotspot Security** on your [BrandMeister SelfCare](#) page, or use the **hotspot password** you've set there as the server password.

If you are using a BrandMeister server in Homebrew protocol mode, you can see the list of currently linked static, dynamic talkgroups and reflectors on the openSPOT2's Status page.

### Cross mode usage

The Homebrew/MMDVM connector is a DMR connector, but it supports C4FM and NXDN cross modem modes.

In cross modem modes, you can set the **Route cross mode calls to ID** to the talkgroup ID you want to talk to (set the call type to **Group call**). If you use a cross modem mode which is ID-based (like NXDN), then this ID (and call type) will be automatically updated based on what ID you call on your radio. This ID will be automatically used if you call the ID 0, or a callsign-based modem mode is used (like C4FM).

You can also update the **Route cross mode calls to ID** with the [Quick call](#) feature.

### C4FM notes

If you use a C4FM radio, make sure it is in digital narrow (DN) mode, otherwise your calls can't be converted and only silence will be sent to the network.

You can update the **Route cross mode calls to ID** with a C4FM radio using DTMF codes. You can start a short private call to an ID by entering the [Special command/cross mode private call DTMF code](#) (by default it's \*) followed by the destination ID in DTMF code. You can start a short group call to an ID by entering the [Cross mode group call DTMF code](#) followed by the destination ID in DTMF code.

So for example with the default DTMF codes, you can link TG3100 by entering DTMF code `#3100`, and link reflector 4770 by entering `*4770`. If you want to try the BM parrot, enter DTMF code `*9990`, release the PTT, then if you press the PTT again, the call will be forwarded to private call 9990, which is usually the BM server's parrot service ID.

You can also use the Wires-X button menu of your C4FM radio to select/search for servers.

## Advanced settings

If *Advanced mode* is enabled, the following settings become available:

You can set a **backup server** which will be used if the openSPOT2 can't connect to the primary server for **Backup server activate connect timeout** seconds. If the backup server is disconnected then the primary one will be tried automatically again.

You can set the **DMO mode TDMA channel** for the network. DMO mode is enabled if the connector's RX and TX frequencies match. This setting specifies which TDMA channel is used for sending calls to the server. On the BrandMeister network the DMO mode links the two timeslots together, so this setting is irrelevant.

### Auto connect

The **Auto connect to ID** will be quick called when the openSPOT2 connects to the server. Please see the [Quick call](#)'s description for more information about this feature.

Using the auto connect feature is **not needed (and not recommended)** on **BrandMeister**, as this network supports auto static talkgroups: the first talkgroup you call **after** calling TG4000 will be set as **auto static**. Auto static talkgroups will not be unlinked after a timeout like dynamic talkgroups (this timeout is usually 15 minutes, but this depends on server settings). Example: start a short call to TG4000. Then start a short call to TG3100. Now TG3100 will be auto static, which means it won't be unlinked after a timeout.

If you are using auto connect to a talkgroup, it is advisable to check the **Disconnect TG/ref. on auto connect** checkbox. If it's checked, the openSPOT2 will automatically quick call TG4000 after it is connected to the server, so the currently active TG/reflector will be disconnected. This is useful if you only want to use a talkgroup, but the server always links you to a reflector automatically.

Note that reflector and talkgroup linking (and unlinking) is **done by the network, not the openSPOT2**. The openSPOT2 has to make quick calls, which are the same as pressing PTT for a short time on a radio (kerchunking). These empty calls will also show up on the BrandMeister network's dashboard.

### Call rerouting

You can enable rerouting by setting the **Reroute DMR TG9 calls to ID** other than 0. If rerouting is active, all voice calls are rerouted to and from TG9. This makes using talkgroups easier, as you don't have to set them in the RX group list on your radio.

For example, if you set the reroute ID to 3100/group call, then if you call TG9 on your radio, your call will be sent to TG3100, and if someone talks on TG3100, the call will be sent to TG9 on your radio, so it will receive it even if you don't have TG3100 in your RX group list.

Another example: if you set the reroute ID to 9990/private call, then if you call TG9, your call will go to the parrot service 9990, and the parrot reply will come back to you to TG9.

Rerouting is dynamic, so if you start a short voice call to a new reflector/talkgroup, the reroute ID will be automatically set to the new reflector/talkgroup with the appropriate call type.

### SSIDs

You can optionally use SSIDs to connect **multiple hotspots** to the BrandMeister network. For example, if your DMR ID is 2161005, and you have 2 hotspots, use SSID 01 and 02 for them. Append the number to your DMR ID in the Homebrew connector settings DMR ID field like this: 216100501 and 216100502 on the other hotspot. BrandMeister will automatically link you as 2161005 with SSID 01 and 02. This way you **don't need multiple DMR IDs** for multiple hotspots.

### Changing modules on XLX servers

To unlink the currently linked module, call DMR ID 4000. To connect to module A, call DMR ID 4001, for module B call DMR ID 4002 and so on.

## DMRplus dongle connector

### DMR/DMRplus dongle

Save

Modem frequency (MHz):	<input type="text" value="433.900000"/>
Modem mode:	<input type="text" value="DMR"/>
Server:	<input type="text" value="IPSC2-EA-Hotspo"/>
DMR ID:	<input type="text" value="2161028"/>
Reflector:	<input type="text" value="4371 EA-1 Spain DMRplus"/>

You can use this connector to connect to a network which supports the DMRplus dongle protocol like DMRplus, Phoenix, DMR-MARC, XLX.

The easiest way is to use the [Quick setup](#) to connect to these networks.

You can change the used reflector ID by starting a short private call on your radio to the desired reflector ID (between 4000 and 5000).

### Cross mode usage

The DMRplus dongle connector is a DMR connector, but it supports C4FM and NXDN cross modem modes. If you use a C4FM radio, make sure it is in digital narrow (DN) mode, otherwise your calls can't be converted and only silence will be sent to the network.

If you are using a C4FM radio, you can switch reflectors from your radio by entering **Cross mode group call DTMF code** (by default it's \*) followed by the reflector ID in DTMF code, or use the Wires-X button menu to select/search for servers.

Example: to switch to reflector 4770, enter DTMF code `*4770` .



## DCS/XLX connector

**D-STAR/DCS/XLX**
Save

Modem frequency (MHz):

Modem mode:

Server:

Module:

Callsign:

Check D-STAR registration

You can use this connector to connect to a network which supports the DCS protocol like DCS or XLX.

The easiest way is to use the [Quick setup](#) to connect to these networks.

DCS and XLX servers usually do not need a valid D-STAR registration, but you can check your callsign's registration by clicking on the **Check D-STAR registration** button.

If you want to talk on a DCS or XLX reflector, set the URCALL on your radio to CQCQCQ, and the local module to D in the openSPOT2. Make sure you have not entered any space characters in your radio before or after CQCQCQ in the URCALL.

## Switching reflectors/gateways using your radio

### URCALL/your call method

You can switch reflectors using your radio by starting a short call to their names as the destination callsign (URCALL), like DCS001 A, or XLX001 A. Set the desired remote module as the last character of the URCALL. You can switch to gateways the same way. You can also switch to REF/XRX servers by this method. Switching to REF/XRF servers or gateways this way changes the openSPOT2's active connector to the [REF/XRF connector](#).

### DTMF method

Here are some example DTMF command sequences:

Switching to REF servers:

- \*1c or \*01c or \*001c will change the active connector to the [REF/XRF connector](#) and connects to REF001 module C.

Switching to XRF servers:

- `B8B` or `B08B` or `B008B` will change the active connector to the [REF/XRF connector](#) and connects to XRF008 module B.

Switching to XLX servers:

- `C5B` or `C502` or `C0502` connects to XLX005 module B.
- `C9D` or `C904` or `C0904` connects to XLX009 module D.

Switching to DCS servers:

- `D5A` or `D501` or `D0501` connects to DCS005 module A.
- `D1C` or `D103` or `D0103` connects to DCS001 module C.

## REF/XRF connector

**D-STAR/REF/XRF (DPlus/DExtra)**
Save

Modem frequency (MHz):

Modem mode:

List:

Server:

Module:

Callsign:

Reflectors  Gateways

Check D-STAR registration

You can use this connector to connect to a network which supports the Dplus protocol like REF or XRF.

The easiest way is to use the [Quick setup](#) to connect to these networks.

If you want to use a **reflector**, set the URCALL on your radio to CQCQCQ, and the local module to D in the openSPOT2. If you want to use a **gateway**, set the URCALL to the gateway's callsign and remote module. Remote module should be the last character of the URCALL. In this case, set the local module to a space character in the openSPOT2.

Note that if you are connecting to a repeater, you have to set the **Reflector/gateway ID** connector setting to the callsign of the repeater.

Some (classic) REF and XRF servers need a valid D-STAR® registration. If you connect to such a server with an invalid callsign registration, then the server may only serve a **read only** connection, which means you won't be able to transmit to the server. A warning notification will be shown on the openSPOT2's web interface, and also you'll see a warning at the *Connected to* field on the [Status page](#).

You can check your callsign's registration by clicking on the **Check D-STAR registration** button. Make sure you have registered your local module on the D-STAR gateway registration system. If you have a registration, you have probably registered the "space" local module, in this case set your openSPOT2's local module to a space (empty) character.

Try remote module E on any reflector for the REF/XRF echo service (may not be enabled on some servers).

### Switching reflectors/gateways using your radio

#### URCALL/your call method

You can switch reflectors using your radio by starting a short call to their names as the destination callsign (URCALL), like REF001 A, or XRF001 A. Set the desired remote module as the last character of the URCALL. You can switch to gateways the same way. You can also switch to DCS/XLX servers by this method. Switching to DCS/XLX servers this way changes the openSPOT2's active connector to the [DCS/XLX connector](#).

### DTMF method

Here are some example DTMF command sequences:

Switching to REF servers:

- \*1C or \*01C or \*001C connects to REF001 module C.

Switching to XRF servers:

- B8B or B08B or B008B connects to XRF008 module B.

Switching to XLX servers:

- C5B or C502 or C0502 will change the active connector to the [DCS/XLX connector](#) and connects to XLX005 module B.
- C9D or C904 or C0904 will change the active connector to the [DCS/XLX connector](#) and connects to XLX009 module D.

Switching to DCS servers:

- D5A or D501 or D0501 will change the active connector to the [DCS/XLX connector](#) and connects to DCS005 module A.
- D1C or D103 or D0103 will change the active connector to the [DCS/XLX connector](#) and connects to DCS001 module C.

### Notes

The openSPOT2 will automatically set the local module to a space character if you are switching to a gateway, and to local module D if you are switching to a reflector using your radio.

## FCS connector

**System Fusion/FCS**
Save

Modem frequency (MHz):	<input style="width: 90%;" type="text" value="433.900000"/>
Modem mode:	<input style="border-bottom: 1px solid #ccc;" type="text" value="C4FM"/> <span style="font-size: 0.8em; vertical-align: middle;">⋮</span>
Server / room number:	<input style="border-bottom: 1px solid #ccc;" type="text" value="099 (ECHO)"/> <span style="font-size: 0.8em; vertical-align: middle;">⋮</span>
Callsign:	<input style="width: 90%;" type="text" value="HA2NON"/>
DMR/CCS7 ID:	<input style="width: 90%;" type="text" value="2161028"/>

You can use this connector to connect to a network which supports the FCS protocol like the FCS network.

The easiest way is to use the [Quick setup](#) to connect to these networks.

You can change rooms with your C4FM radio by entering the [Special command/cross mode private call DTMF code](#) DTMF code (by default it's \*) followed by the desired room number (for example \*99 for room 99). You can switch FCS servers if you use 3 digits. Example: entering \*303 will switch to FCS003 room 03.

You can also use the Wires-X button menu of your C4FM radio to select/search for servers.

### Cross mode usage

The FCS connector is a C4FM connector, but it supports DMR, NXDN and P25 cross modem modes.

You can change rooms by starting a private call to the desired room number. You can switch FCS servers if you use 3 digits. Example: starting a private call to ID 303 will switch to FCS003 room 03.

## YSFReflector connector

**System Fusion/YSFReflector**
Save

Modem frequency (MHz):	<input style="width: 90%;" type="text" value="433.900000"/>
Modem mode:	<input style="border-bottom: 1px solid #ccc;" type="text" value="C4FM"/> <span style="font-size: 0.8em; vertical-align: middle;">⋮</span>
Server:	<input style="border-bottom: 1px solid #ccc;" type="text" value="HU Hungary / BM TG216"/> <span style="font-size: 0.8em; vertical-align: middle;">⋮</span>
Callsign:	<input style="width: 90%;" type="text" value="HA2NON"/>

You can use this connector to connect to a network which supports the YSFReflector protocol, like YSFReflector servers.

The easiest way is to use the [Quick setup](#) to connect to these networks.

### Switching YSFReflector servers using your radio

You can switch YSFReflector servers with your radio by entering the [Special command/cross mode private call DTMF code](#) DTMF code (by default it's \*) followed by the desired server's ID (5 numbers).

You can also use the Wires-X button menu of your radio to select/search for servers.

### Cross mode usage

The YSFReflector connector is a C4FM connector, but it supports DMR, NXDN and P25 cross modem modes.

You can change servers using DMR, NXDN and P25 radios by calling their IDs with a private call.

## NXDNRreflector connector

The screenshot shows a configuration window for the NXDNReflector connector. At the top left is the title 'NXDNReflector' and at the top right is a blue 'Save' button. Below the title is a form with four fields:
 

- 'Modem frequency (MHz):' with a text input containing '433.900000'.
- 'Modem mode:' with a dropdown menu showing 'DMR'.
- 'Server:' with a dropdown menu showing 'Alabama-Link, 31010'.
- 'Callsign:' with a text input containing 'HA2NON'.

You can use this connector to connect to a network which supports the NXDNReflector protocol, like NXDNReflector servers.

The easiest way is to use the [Quick setup](#) to connect to these networks.

### Talkgroup handling

NXDNRreflector servers handle only one talkgroup. You can find each server's talkgroup ID in the server list, or if you enable *Advanced mode*, then the talkgroup ID setting will be shown.

- If you use NXDN modem mode, then make sure you send your calls as a group call to the talkgroup ID, otherwise the server will ignore your calls. Also make sure you have the talkgroup ID in your SelCall list (if SelCall is enabled), otherwise you won't hear incoming calls on your radio.
- If you use cross modem modes, then send your calls as a group call to the talkgroup ID, otherwise the server will ignore your calls.

If you use a DMR radio, then make sure the talkgroup ID is set as the TX contact for the current channel in your radio, or the talkgroup ID is in an RX group list which is assigned to the current channel of your radio, otherwise you won't hear incoming calls from the server.

You can also set the **Default cross mode destination ID** to the talkgroup ID and enable **Always use this ID in cross mode**. These settings are available on the [Settings page, NXDN settings](#) section. This way your cross mode calls will be always sent to the correct talkgroup ID.

### Switching NXDNReflector servers using your radio

You can switch NXDNReflector servers with your radio by calling the desired server's ID with a private call.

### Cross mode usage

The NXDNReflector connector is an NXDN connector, but it supports DMR and C4FM cross modem modes.

You can switch NXDNReflector servers with your DMR radio by calling the desired server's ID with a private call.

You can switch NXDNReflector servers with your C4FM radio by entering the [Special command/cross mode private call DTMF code](#) DTMF code (by default it's \*) followed by the desired server's ID (5 numbers).

You can also use the Wires-X button menu of your C4FM radio to select/search for servers.



## P25Reflector connector

**P25Reflector**
Save

Modem frequency (MHz):

Modem mode:

⋮

Server:
  
Callsign:

⋮

You can use this connector to connect to a network which supports the P25Reflector protocol, like P25Reflector servers.

The easiest way is to use the [Quick setup](#) to connect to these networks.

### Talkgroup handling

P25Reflector servers handle only one talkgroup. You can find each server's talkgroup ID in the server list, or if you enable *Advanced mode*, then the talkgroup ID setting will be shown.

If the **Enable group call rerouting** setting is enabled (it's enabled by default), then all calls from your P25 radio will be forwarded to the correct talkgroup ID, and all calls from the network will be forwarded to talkgroup set at **Reroute group calls from net to ID** to your radio. In this case, make sure you have this talkgroup in your radio's selective call list (if SelCall is enabled), or set this talkgroup as the TX contact for the radio's current channel, otherwise you won't hear incoming calls on your radio.

Rerouting can be useful if you can't modify your radio's configuration (you don't have a programming cable or software for the radio). In this case, keep rerouting enabled, and set the **Reroute group calls from net to ID** to the talkgroup you already have in your radio.

If the **Enable group call rerouting** setting is disabled, then you have to send your calls to the correct talkgroup (the P25Reflector's TG ID), and incoming calls will be sent to their original destination ID (the P25Reflector's TG ID).

If you use C4FM modem mode, then your calls will be automatically sent to the P25Reflector's TG ID regardless of the **Enable group call rerouting** setting.

### Switching P25Reflector servers using your radio

You can switch P25Reflector servers with your radio by calling the desired server's ID with a private call.

### Cross mode usage

The P25Reflector connector is an P25 connector, but it supports C4FM cross modem mode. Make sure you use VW (Voice Wide) mode on your C4FM radio, otherwise your calls sent to the server will be silent.

You can switch P25Reflector servers with your C4FM radio by entering the [Special command/cross mode private call DTMF code](#) DTMF code (by default it's \*) followed by the desired server's ID, or use the Wires-X button menu to select/search for servers.

## DAPNET connector

**DAPNET**
Save

This connector uses the POCSAG frequency which can be set on the Settings page, POCSAG settings section.

Modem mode:

Server:

Use your DAPNET transmitter login and auth key here. These are not the same as your DAPNET user account. A separate transmitter registration is needed. See [this page](#) for more information.

Transmitter callsign:

Transmitter auth key: ([show](#))

Stay connected in background

Transmit time messages

ROT1 decoder enabled

**Only transmit specified RICs**

Enabled

RIC #1:	<input style="width: 100%;" type="text" value="161028"/>
RIC #2:	<input style="width: 100%;" type="text" value="0"/>
RIC #3:	<input style="width: 100%;" type="text" value="0"/>
RIC #4:	<input style="width: 100%;" type="text" value="0"/>
RIC #5:	<input style="width: 100%;" type="text" value="0"/>
RIC #6:	<input style="width: 100%;" type="text" value="0"/>
RIC #7:	<input style="width: 100%;" type="text" value="0"/>
RIC #8:	<input style="width: 100%;" type="text" value="0"/>

You can use this connector to connect to a network which supports the **DAPNET** protocol.

The easiest way is to use the [Quick setup](#) to connect to these servers.

This connector supports the POCSAG modem protocol which is the protocol used by pagers. You can find more information about POCSAG and DAPNET [here](#).

The openSPOT2 acts as a transmitter on DAPNET, so make sure you use your DAPNET **transmitter** login and auth key for connecting the openSPOT2. These are not the same as your DAPNET user account. A separate transmitter registration is needed. See [this page](#) for more information.

This connector can stay active in the background if **Stay connected in background** is checked, which means you can use the openSPOT2 the same way as before with other active connector, but it will function also as a personal DAPNET/POCSAG transmitter. The openSPOT2 will automatically broadcast DAPNET/POCSAG messages when there's no call coming from the network or modem for POCSAG TX delay seconds (this can be set on the [Settings page, POCSAG settings section](#)), so ongoing calls won't be interrupted by POCSAG messaging. If the DAPNET connector is running in the background then you can see it's connection status on the Status page.

If **Transmit time messages** is unchecked then the openSPOT2 won't transmit time messages from the DAPNET server. Some messages are ROT1 encoded, these will be automatically decoded if the **ROT1 decoder** is enabled. Note that not all ROT1 encoded messages can be recognized automatically so there can be received messages which won't be automatically decoded.

You can specify 8 RICs (IDs in POCSAG terminology) to filter messages to. If the **Enabled** checkbox is checked, then the openSPOT2 will only transmit POCSAG messages sent to these RICs by the DAPNET server.

## DAPNET registration

You'll need 2 accounts to use DAPNET: one personal and one transmitter account. Here's how to create these accounts.

First, open a new ticket at [support.hampager.de](https://support.hampager.de) and select **New DAPNET Account with RIC** as the Help Topic. Fill in all required data and create the ticket. After this open a new ticket again and select **New DAPNET Transmitter** as the Help Topic. Fill in all required data and create the ticket.

You can set your callsign for both the personal and the transmitter account. For the transmitter registration, set the **Transmitter Type** to personal, and the **Antenna Type** to Omni. **Transmitter output power** is 0.02 for the openSPOT2. You can set the **Antenna gain in dBi** to 1 dB.

## APRS connector

**APRS**
Save

This special background connector can be used for setting up a connection to the APRS network. This enables device location broadcasting, forwarding GPS position data from transceivers, and messaging (APRS chat).

Server:

Callsign:

Enable in background

**Device location**

Allow uploading device location

QTH locator ([map](#)):

Latitude (decimal degrees):

Longitude (decimal degrees):

Height (ASL, meters):

Location symbol on map ([select](#)):

Location comment:

**D-STAR location forwarding**

Enable

**C4FM location forwarding**

Enable

You can use this connector to connect to the APRS network. The easiest way is to use the [Quick setup](#) to set up the connection.

This connector can stay active in the background if the **Enable in background** checkbox is checked, which means you can use the openSPOT2 the same way as before with another active connector, but it will function also as an APRS client.

The openSPOT2 will automatically receive APRS chat messages and if enabled, it can transmit the device location or position information received from D-STAR and C4FM/Fusion radios. If the APRS connector is enabled then you can see its connection status on the Status page.

You can also enable forwarding incoming APRS chat messages to your POCSAG receiver by setting a RIC at **Send incoming messages to POCSAG RIC**.

If *Advanced mode* is enabled, then you can set forced a SSID, symbol and APRS comment for forwarded location data.



## SharkRF IP Connector Client connector

**SharkRF IP Connector Client**
Save

Modem frequency (MHz):

Modem mode:

Server:

Server address:

Add server
Remove

ID:

Server password: [show](#)

Callsign:

You can use this connector to directly connect to another openSPOT, openSPOT2, or a network which supports the SharkRF IP Connector protocol, or our open source [SharkRF IP Connector Protocol Server](#). The protocol's documentation can be found on [GitHub](#), so you can also develop your own application for the openSPOT2.

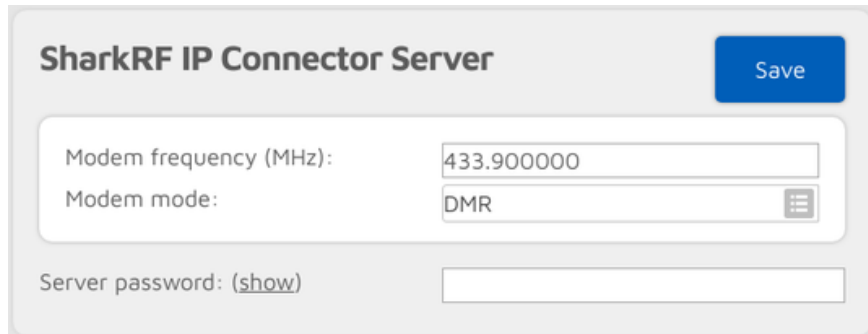
The easiest way is to use the [Quick setup](#) to connect to a SharkRF IP Connector server.

If you want to connect directly to another openSPOT or openSPOT2, make sure that the another device is set up as the server, and the UDP port used by this connector (by default it's 65100) is correctly set up on the server's router (it's open on the firewall and is forwarded to the server openSPOT/openSPOT2 on it's local network).

All modem modes are supported by this connector, with automatic cross mode conversion between the following modes:

- DMR, C4FM (DN mode) and NXDN.
- C4FM (VW mode) and P25.

## SharkRF IP Connector Server connector



The screenshot shows a web-based configuration interface for the SharkRF IP Connector Server. At the top left, the title "SharkRF IP Connector Server" is displayed. In the top right corner, there is a blue "Save" button. Below the title, there are three input fields: "Modem frequency (MHz)" with the value "433.900000", "Modem mode" with the value "DMR" and a dropdown menu icon, and "Server password" with a "(show)" link and an empty input field.

You can use this connector to accept connections from another openSPOT and openSPOT2 devices. This connector only supports 1 active connection, so if you want to link more than 1 device, you can use our open source [SharkRF IP Connector Protocol Server](#). The protocol's documentation can be found on [GitHub](#), so you can also develop your own application for the openSPOT2.

The easiest way is to use the [Quick setup](#) to set up this connector.

Make sure that the UDP port used by this connector (by default it's 65100) is correctly set up on your router (it's open on the firewall and is forwarded to the openSPOT2's IP address).

All modem modes are supported by this connector, with automatic cross mode conversion between the following modes:

- DMR, C4FM (DN mode) and NXDN.
- C4FM (VW mode) and P25.



## AutoCal connector

### AutoCal Save

AutoCal analyzes an incoming transmission and tries to find out the frequency difference (RX offset) between the openSPOT2 and the transmitter.

Make sure the correct modem mode is selected. Hold the PTT button on your radio until all 3 phases are completed.

Modem frequency (MHz):	<input type="text" value="433.900000"/>
Modem receive frequency offset (Hz):	<input type="text" value="0"/>
Modem mode:	<input type="text" value="NXDN"/>
Phase:	<b>Phase 1/3</b>
Status:	AutoCal not active
Progress:	<input type="text"/>
Last found offset:	N/A

AutoCal is a special connector - it does not connect to anywhere. It analyzes an incoming transmission from the modem and tries to find out the frequency difference (RX offset) between the openSPOT2 and the transmitter.

openSPOT2's oscillator has a frequency stability of 0.5ppm, and every openSPOT2 is factory calibrated by us, so you'll rarely need to use this special connector. We suggest you to only run AutoCal if the openSPOT2 shows BER above 1% for your radio's transmission, as errors in the voice stream usually won't be noticeable if the BER is below 1%.

To use AutoCal, set the modem mode to the mode your radio uses and the modem frequency to the radio's transmit frequency, then click on the **Save** button. Hold the PTT button on your radio until all 3 phases are completed. The found RX offset will be displayed at the **Last found offset** field and it will be automatically saved for the current modem mode.

Keep in mind, that you can always check your own voice quality by using the openSPOT2's built-in [local echo service](#).

If the AutoCal does not enter Phase 2, then hold the PTT on your transceiver for at least the time it takes for the progress bar to go from 0% to 100%. Make sure the transceiver is transmitting on the AutoCal frequency (433.900 MHz by default). If the AutoCal still does not enter Phase 2, then the RX offset between the transceiver and the openSPOT2 is greater than 500Hz. What you can do is to switch frequency on the transceiver or on the openSPOT2, or enter an RX offset greater than 500Hz manually on the modem page and use the openSPOT2's built-in [local echo service](#) to find out the lowest BER.

## Powersaving

If powersaving is enabled, and it's active, the openSPOT2 consumes much less power. Powersaving is active if the standby LED sequence is blinking in a much lower interval. The openSPOT2 can only enter powersaving state if the web interface is not open and there are no ongoing calls.



Enabling powersaving reduces the LED's brightness, turns off the [USB serial console](#) feature, and puts the Wi-Fi interface into power save mode. The built-in LLNMR and MDNS servers may not function properly in this mode, so opening the web interface by entering the openSPOT2's hostname into the browser (example: <http://openspot2.local>) may not work reliably. Use [SharkRF Link](#) to open the web interface.

## Control calls

There are some special IDs and callsigns which trigger built-in services and actions when a call is received to them from your radio.

### Local echo service

If the local echo service's ID or callsign is called, the openSPOT2 records the voice transmission and plays it back after the call from the radio is finished. This is useful for hearing your own voice back to test the connection between your radio and the openSPOT2. If you have local echo voice quality issues, see the [AutoCal](#) special connector.

The local echo service can be triggered if the call is sent to the following IDs/callsigns:

- **DMR, NXDN and P25:\*\*** ID 9999. Both private and group calls to this ID are answered with a private call.
- **D-STAR:** call is sent to `.....E` (7 spaces and an E, replace dots with spaces). *Note that currently the D-STAR® echo reply is limited around 2 seconds.*
- **C4FM:** call DGID is set to 99.

Note that only a few seconds can be recorded and played back by the local echo service. Exact length also depends on the used modem mode.

### Current configuration profile query

The current configuration profile's number can be queried by calling the following IDs/callsigns:

- **DMR, NXDN and P25:** ID 9000. Both private and group calls are interpreted.
- **D-STAR:** call is sent to `.....P` (7 spaces and an P, replace dots with spaces), or DTMF code A is entered.
- **C4FM:** [Special command/cross mode private call DTMF code](#) (by default it's \*) is entered followed by DTMF code A.

The current configuration profile number will be announced by the voice. The DMR/NXDN/P25 profile query ID can be changed on the Settings page, [Voice announcements](#) section.

### Change configuration profile

The active configuration profile can be changed by calling the following IDs/callsigns:

- **DMR, NXDN and P25:** ID 9000+configuration profile number. Example: to switch to config profile 1, start a short call to 9001. Both private and group calls are interpreted.

- **D-STAR:** call is sent to .....1 (7 spaces and the configuration profile number, replace dots with spaces), or DTMF code A is entered followed by the configuration profile number.
- **C4FM:** [Special command/cross mode private call DTMF code](#) (by default it's \*) is entered followed by DTMF code A, then the desired configuration profile number.

## Connector status query

The currently active connector's status can be queried by calling the following IDs/callsigns:

- **DMR, NXDN and P25:** ID 9998. Both private and group calls are interpreted.
- **D-STAR:** call is sent to .....I (7 spaces and an I, replace dots with spaces), or DTMF code D or O is entered.
- **C4FM:** [Special command/cross mode private call DTMF code](#) is entered followed by DTMF code D.

The connector status will be announced by voice. The DMR/NXDN/P25 connector status query ID can be changed on the Settings page, [Voice announcements](#) section.

## Active IP address query

The active IP address can be queried by calling the following IDs/callsigns:

- **DMR, NXDN and P25:** ID 9997. Both private and group calls are interpreted.
- **D-STAR:** call is sent to .....A (7 spaces and an A, replace dots with spaces), or DTMF code B is entered.
- **C4FM:** [Special command/cross mode private call DTMF code](#) is entered followed by DTMF code B.

The IP address will be announced by voice. The DMR/NXDN/P25 IP address query ID can be changed on the Settings page, [Voice announcements](#) section.

## D-STAR unlink command

The openSPOT2 is designed to be always connected, however, the unlink (U) command is supported and activates the Null connector (which - as its name says - does not connect anywhere).

You can unlink if a call is sent to .....u (7 spaces and a U, replace dots with spaces), or DTMF code # is entered.

## Cross modem modes

The openSPOT2 supports the following cross modes:

- You can use your **DMR** transceiver to access **C4FM** and **NXDN** networks
- You can use your **C4FM** transceiver to access **DMR**, **NXDN** and **P25** networks
- You can use your **NXDN** transceiver to access **DMR** and **C4FM** networks
- You can use your **P25** transceiver to access **C4FM** networks

Note that cross mode using a D-STAR transceiver/network is not supported, as D-STAR uses an older, incompatible voice codec, and it requires built-in audio transcoding hardware, which only the [openSPOT3](#) has.

### Using a DMR connector with a C4FM radio

After you've activated the DMR connector, switch the modem to C4FM mode (or C4FM Half Deviation mode, if your radio needs that). Only calls in DN (Digital Narrow) mode will be converted, if you transmit in other modes, DMR users will only hear silence.

The source DMR ID of your calls from the C4FM radio will be set to the ID associated with the source callsign. If there's no ID associated, then the source callsign will be parsed into a number and this will be used as the source DMR ID. If it can't be parsed into a number, then the [Default cross mode source DMR ID](#) will be used. You can set this to your DMR ID.

The callsign of incoming DMR calls will be set to the callsign associated with the DMR ID. If there's no callsign associated in the database, then the [Default callsign for cross mode calls](#) will be used. If it's not set, the callsign will be the source DMR ID as text.

Keep in mind that the audio gain used by DMR and C4FM transceivers differ, and correcting this would require audio transcoding hardware which the OS2 does not have, so the audio coming from the network may sound distorted on your C4FM transceiver.

### Using a DMR connector with an NXDN radio

After you've activated the DMR connector, switch the modem to NXDN mode.

The source DMR ID of your NXDN calls will be set to the same value as the NXDN call unless the **Always use this ID in cross mode** setting is enabled for the **Default cross mode source ID** on the Settings page, [NXDN settings](#) section.

Your NXDN calls will be sent to the same DMR ID as the NXDN call's destination ID unless the **Always use this ID in cross mode** setting is enabled for the **Default cross mode destination ID** on the Settings page, [NXDN settings](#) section.

The destination and source NXDN IDs for an incoming DMR call will be set to the same value as the incoming call's destination and source ID unless the **Always use this ID in cross mode** setting is enabled for the destination/source NXDN IDs on the Settings page, [NXDN settings](#) section (or if the incoming DMR call's destination or source ID is out of the NXDN ID range (0-65535)).

Private DMR calls are converted to NXDN group calls.

## Using a C4FM connector with a DMR radio

After you've activated the C4FM connector, switch the modem to DMR mode. Only calls in DN (Digital Narrow) mode will be converted to your DMR radio, if someone transmits in other modes on the C4FM network, you will hear silence on your DMR radio.

Your outgoing callsign will be the callsign associated with the DMR source ID. If there's no callsign associated to the ID, then the callsign will be the [Default callsign for cross mode calls](#). If this is not set, your outgoing callsign will be your radio's DMR ID as text.

An incoming call's source DMR ID will be set to the ID associated with the source callsign. If there's no ID associated, and the source callsign can be parsed into a number, it will be used as the source DMR ID. Otherwise the [Default cross mode source DMR ID](#) will be used.

All calls coming from the C4FM network will be sent as a group call to DMR ID 9 (TG9). Make sure TG9 is the TX contact of your DMR radio's current channel, or it's in an RX group list which is assigned to the current channel in the radio, otherwise the squelch won't open and you won't hear the calls.

Keep in mind that the audio gain used by DMR and C4FM transceivers differ, and correcting this would require audio transcoding hardware which the OS2 does not have, so the audio coming from the network may sound quiet on your DMR transceiver.

## Using a C4FM connector with an NXDN radio

After you've activated the C4FM connector, switch the modem to NXDN mode. Only calls in DN (Digital Narrow) mode will be converted to your NXDN radio, if someone transmits in other modes on the C4FM network, you will hear silence on your NXDN radio.

Your outgoing callsign will be the callsign associated with the NXDN source ID. If there's no callsign associated to the ID, then the callsign will be the [Default callsign for cross mode calls](#). If this is not set, your outgoing callsign will be your radio's NXDN ID as text.

An incoming call's source NXDN ID will be set to the ID associated with the source callsign. If there's no ID associated, and the source callsign can be parsed into a number, it will be used as the source NXDN ID. Otherwise the [Default cross mode source ID](#) will be used.

All calls coming from the C4FM network will be sent as a group call to the [Default cross mode destination ID](#). If you don't hear calls on your radio then make sure this ID is in the selective call list (or SelCall is turned off), or it's the TX contact of the current channel.

Keep in mind that the audio gain used by NXDN and C4FM transceivers differ, and correcting this would require audio transcoding hardware which the OS2 does not have, so the audio coming from the network may sound quiet on your DMR transceiver.

## Using a C4FM connector with a P25 radio

After you've activated the C4FM connector, switch the modem to P25 mode. Only calls in VW (Voice Wide) mode will be converted to your P25 radio, if someone transmits in other modes on the C4FM network, you will hear silence on your P25 radio.

Your outgoing callsign will be the callsign associated with the P25 source ID. If there's no callsign associated to the ID, then the callsign will be the [Default callsign for cross mode calls](#). If this is not set, your outgoing callsign will be your radio's P25 ID as text.

An incoming call's source P25 ID will be set to the ID associated with the source callsign. If there's no ID associated, and the source callsign can be parsed into a number, it will be used as the source P25 ID. Otherwise the [Default cross mode source P25 ID](#) will be used.

All calls coming from the C4FM network will be sent as a group call to P25 ID 9 (TG9). Make sure TG9 is in the selective call list of your P25 radio's current channel, or SelCall is turned off, or TG9 is set as the TX contact for the current channel, otherwise the squelch won't open and you won't hear the calls.

## Using an NXDN connector with a DMR radio

After you've activated the NXDN connector, switch the modem to DMR mode.

The source NXDN ID of your DMR calls will be set to the same value as the DMR call unless the **Always use this ID in cross mode** setting is enabled for the **Default cross mode source ID** on the Settings page, [NXDN settings](#) section.

Your DMR calls will be sent to the same NXDN ID as the DMR call's destination ID unless the **Always use this ID in cross mode** setting is enabled for the **Default cross mode destination ID** on the Settings page, [NXDN settings](#) section, or the destination DMR ID is out of the NXDN ID range (0-65535). In this case, the **Default cross mode destination ID** will be used for the NXDN call.

The destination and source DMR IDs for an incoming NXDN call will be set to the same value as the incoming call's destination and source ID unless the **Always use this ID in cross mode** setting is enabled for the destination/source NXDN IDs on the Settings page, [NXDN settings](#) section.

Make sure the destination ID of the NXDN calls (the NXDN talkgroup) is the TX contact of your DMR radio's current channel, or it's in an RX group list which is assigned to the current channel in the radio, otherwise the squelch won't open and you won't hear the calls.

## Using an NXDN connector with a C4FM radio

After you've activated the NXDN connector, switch the modem to C4FM mode (or C4FM Half Deviation mode, if your radio needs that). Only calls in DN (Digital Narrow) mode will be converted, if you transmit in other modes, NXDN users will only hear silence.

All C4FM calls will be sent to the [Default cross mode destination ID](#) on the NXDN network.

The source NXDN ID of your calls from the C4FM radio will be set to the ID associated with the source callsign. If there's no ID associated, then the source callsign will be parsed into a number and this will be used as the source NXDN ID. If it can't be parsed into a number, then the [Default cross mode source ID] will be used. The default ID will be used if the **Always use this ID in cross mode** setting is enabled for the **Default cross mode source ID** on the Settings page, [NXDN settings](#) section.

The callsign of incoming NXDN calls will be set to the callsign associated with the NXDN ID. If there's no callsign associated in the database, then the [Default callsign for cross mode calls](#) will be used. If it's not set, the callsign will be the source NXDN ID as text.

Keep in mind that the audio gain used by NXDN and C4FM transceivers differ, and correcting this would require audio transcoding hardware which the OS2 does not have, so the audio coming from the network may sound distorted on your C4FM transceiver.

## Using a P25 connector with a C4FM radio

After you've activated the P25 connector, switch the modem to C4FM mode (or C4FM Half Deviation mode, if your radio needs that). Only calls in VW (Voice Wide) mode will be converted, if you transmit in other modes, P25 users will only hear silence.

The source P25 ID of your calls from the C4FM radio will be set to the ID associated with the source callsign. If there's no ID associated, then the source callsign will be parsed into a number and this will be used as the source P25 ID. If it can't be parsed into a number, then the [Default cross mode source P25 ID](#) will be used. You can set this to your DMR/P25/CCS7 ID.

The callsign of incoming P25 calls will be set to the callsign associated with the P25 ID. If there's no callsign associated in the database, then the [Default callsign for cross mode calls](#) will be used. If it's not set, the callsign will be the source P25 ID as text.



## USB serial console

```

minicom -D /dev/ttyACM1
Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyACM1, 12:46:37

Press CTRL-A Z for help on special keys

sercon: inf: SharkRF openspot2 (openspot2)
SRF-OSW-1.0 hu v0009 12345678 up 557 wl 5.5.3 GS2K BUILDER 26104
diag 0,0000007,00000008,000001c3,000001a1,000001a2,000001a3,000001a4
Ready.

h
sercon: available commands:
log - toggle log
inf - devinfo
rbt - reboot
rbc - config reset
rbb - reboot to btldr
rsp - reset wwv passwd
ccp (nr) - get/switch config profile
wsc - wifi scan
ccc [country code] - change country
net - print net cfg
nst - print net stats
pwr (0|1) - pwr saving
apm - toggle ap mode
apa (0|1) - start in ap mode
aps [ssid] - ap ssid
apk [key] - ap key
apc [nr] - ap channel
mac [mac] - change mac
sts [ssid] - connect ssid
stb [bssid] - connect bssid
stk [key] - connect key
nmd [mode] - change net mode
nip [ip] - change ip
ntm [mask] - change netmask
npr [ip] - change gw
nd1 [ip] - change dns1
nd2 [ip] - change dns2
nhn [hostname] - change hostname
ready.

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyACM1

```

The openSPOT2 shows up as a USB serial port (COM port) when plugged into a computer. You can reach the serial console by opening this USB serial port with serial terminal software like [PuTTY](#), [RealTerm](#), [TeraTerm](#), minicom etc. To find out which is your openSPOT2's serial port device path (COM port number), look at the device list of your computer (in your operating system's control panel for example). You can use the following settings for opening the serial port, but in theory any configuration would work: baud rate 115200, 8 data bits, no parity bits, 1 stop bit.

The openSPOT2's serial console does not require drivers on modern operating systems, as it shows up as a standard USB ACM device with our official USB ID 0x1fc9:0x8125. You can download a driver for Windows XP, 7 and Vista [here](#). Some users reported that Windows 8.1 also requires this driver, but it can only be installed if the signed driver requirement is turned off.

The serial console can be used to change basic network settings, reboot the openSPOT2, switch it to bootloader mode, see its debug log, reset the web interface password and many more. The list of available commands can be queried with the **help** command.

## API

You can use both an HTTP/WebSocket and a UDP application programming interface to develop your own applications which use the openSPOT2.

You can find more information about the API protocols on our GitHub pages:

- [openSPOT2 HTTP/WebSocket API](#)
- [SharkRF IP Connector Protocol](#) (a [server application](#) is also available)
- [SharkRF Voice Announcement server](#)

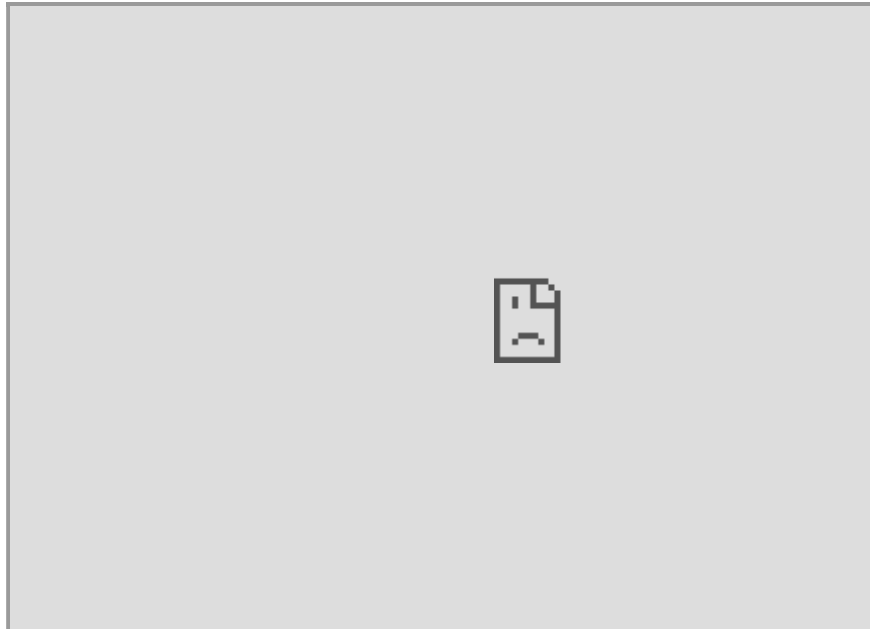
## Factory reset

**Hold the button for 30 seconds** to perform a full factory reset. In this case all configuration profiles will be cleared.

**Please only use a ballpoint pen to press the button, don't use a sharp item like a needle!**

### Factory reset using the web interface

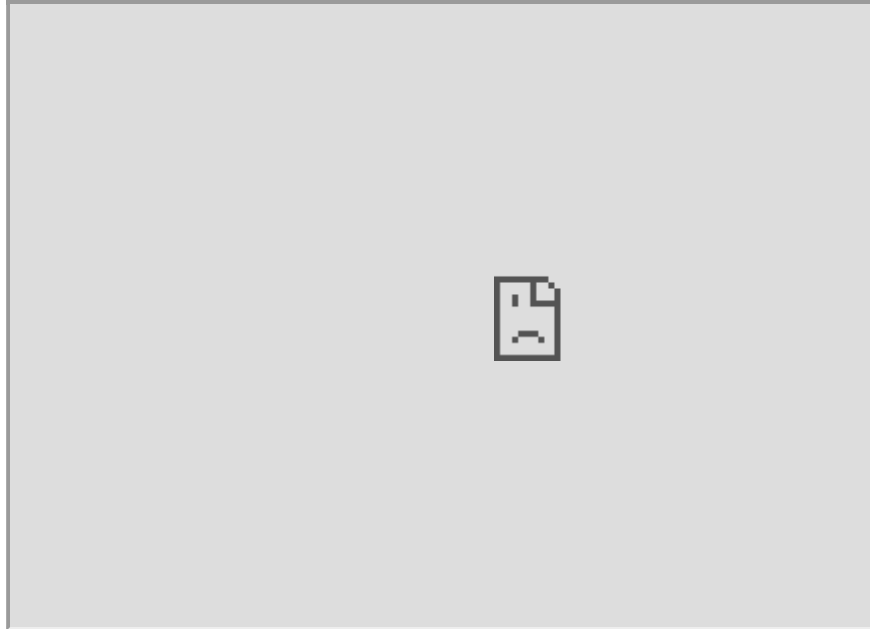
The openSPOT2 can be reset to factory defaults if you click on the **Reset all config profiles** button at the bottom of the [Settings](#) page, or you hold the [button](#) for at least 30 seconds (until the LED turns off). These methods will reset all configuration profiles to the default settings.



If you only want to reset the active configuration profile to defaults, you can use the **Reset config profile** button at the bottom of the [Settings](#) page, or you can enter the **rbc** serial console command.

## Troubleshooting

Please make sure you **always use the latest firmware version**. Here's a video which shows how you can upgrade to the latest firmware instantly without waiting for the scheduled automatic upgrade:



**Do NOT unplug the openSPOT2 during the firmware upgrade!**

If you can't find an answer to your issue, then please send an [email](#) to us, or ask help in our [community forum](#)!

### The web interface can't be opened

Use [SharkRF Link](#) to open the web interface.

### SharkRF Link does not open the openSPOT2 web interface

Make sure your web browser device (phone/tablet/computer) is connected to the same Wi-Fi network the openSPOT2 is using.

Check if the openSPOT2 is connected to the Wi-Fi network by looking at its LED. Please see the [LED states](#) section for possible LED states. If the openSPOT2 is not connected to the Wi-Fi network according to the LED, see the next troubleshooting question's answer on how to set up the openSPOT2's Wi-Fi connection.

Check the Wi-Fi client isolation in your Wi-Fi router's settings. It should be turned off.

### The web interface only loads partially

Make sure you use a supported browser (Chrome, Firefox or Safari), and that the browser is updated to the latest version.

If you're trying to do the initialization setup (the openSPOT2's LED is white), then place the openSPOT2 close to your browser device (phone/tablet/computer) during the setup.

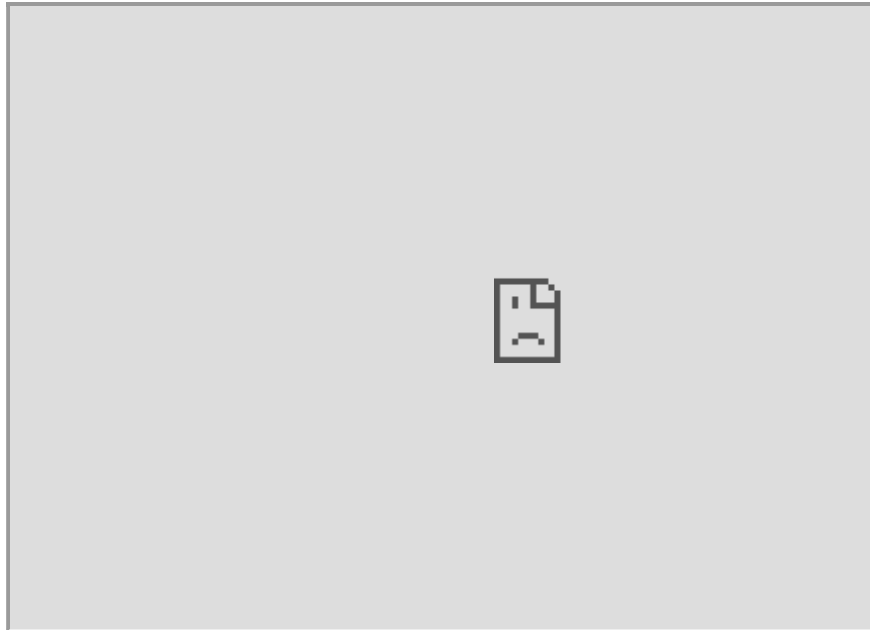
Otherwise, try placing the openSPOT2 closer to the Wi-Fi access point (your Wi-Fi router).

## The openSPOT2 can't connect to the Wi-Fi network

Please first check the LED color on the device. You can see the list of LED states on [this](#) page. The openSPOT2 can't connect to the Wi-Fi network only if the LED is pulsating blue for a longer period of time (at least 1-2 minutes). If you see other LED colors than blue, then the issue is not with connecting to the Wi-Fi network.

- You can reconfigure the Wi-Fi connection if you switch back the openSPOT2 to access point mode by pressing the button for 3 seconds.
- The openSPOT2 will start broadcasting its own Wi-Fi network called **openSPOT2 AP**, indicated by white LED. You can connect to this network using your browser device (phone/tablet/computer). The initialization setup will pop up, where you can select the country of operation and configure a Wi-Fi network for the openSPOT2 to connect to.
- Make sure you've entered the Wi-Fi password case sensitive.
- After you've chosen the Wi-Fi network and entered the password, the openSPOT2 stops broadcasting its own Wi-Fi network and starts connecting to the selected network. At that moment, your browser device (phone/tablet/computer) will disconnect from the openSPOT2's network. What you have to do now is to connect to the exact same Wi-Fi network with your browser device as you've configured for the openSPOT2.
- If you have WEP encryption on your router, then make sure the password is in ASCII, not hex. The openSPOT2 only supports ASCII WEP passwords. The best is to switch the encryption to WPA2 or WPA because it's much safer.
- Make sure you have enabled 2.4GHz Wi-Fi, not just 5GHz.
- Place the openSPOT2 close to the Wi-Fi router.
- Make sure the Wi-Fi bandwidth is set to max. 20Mhz in the Wi-Fi router (see explanation [here](#)).
- If it still does not work, try switching Wi-Fi channel on your Wi-Fi router. Only channels 1, 6 or 11 should be used (see explanation [here](#)).

Here's a video which shows how to switch the openSPOT2 to AP mode and configure it for connecting to a Wi-Fi network:



You can also find more information on the [Connecting the openSPOT2 to a Wi-Fi network](#) page on the steps required to set up the openSPOT2's Wi-Fi connection.

## The openSPOT2 can't connect to my phone's Wi-Fi network

Note for iPhones: the phone's hotspot settings page needs to be open all the time during the openSPOT2 Wi-Fi setup because the iPhone will only broadcast it's own Wi-Fi network if the internet share hotspot settings page is open.

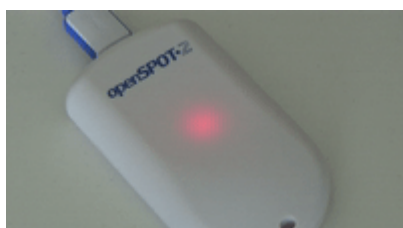
First you should connect your openSPOT2 to your home Wi-Fi network and then when it's connected (LED is green or green/yellow) open the web interface using [sharkrf.link](#). After that you can set up additional Wi-Fi networks if you turn on *Advanced mode* (checkbox at the bottom right corner of the web interface) and go to the [Network page](#), [Wireless settings](#) section. Enter your phone's hotspot SSID and key as the 2nd Wi-Fi network.

After that if the openSPOT2 can't connect to your home Wi-Fi network it will also try connecting to your phone's Wi-Fi network.

## I'm not hearing anything on my radio

Please read the [Setting up your transceiver](#) section.

Check if your radio's LED is on when the openSPOT2 is transmitting. The openSPOT2 is transmitting if it's LED is blinking red:



If it's blinking alternating red/orange, then BCLO is on:



If this is the case then make sure it's turned off by setting it to 0 at the [Other modem settings](#) section.

If your radio's LED does not turn on when the openSPOT2 is transmitting, then make sure they are on the same RX and TX frequencies.

Make sure the openSPOT2's modem is set to the same mode as your radio, and the following settings are set:

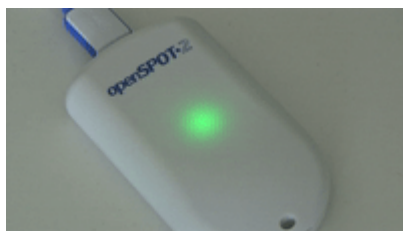
- **DMR radio:** the color code (CC) match in your radio and the openSPOT2 (example: set it to the default 1). Make sure the talkgroup you want to hear is set as the TX contact of your DMR radio's currently active channel, or the talkgroup is in an RX group list which has been assigned to the currently active channel. You should also add TG9 to your the RX group list and assign it to the currently active channel to hear the openSPOT2's voice announcements.
- **C4FM radio:** the DGID is set to the same value both in your radio and the openSPOT2 (example: set it to the default 0).
- **NXDN radio:** the radio access number (RAN) match in your radio and the openSPOT2 (example: set it to the default 0). Make sure the talkgroup you want to hear is in your NXDN radio's contact list, or it is set as the TX contact of the currently active channel.

## No one is hearing me on the network

Please read the [Setting up your transceiver](#) section.

Check where your radio is sending your call by looking at the [Call log](#) on the openSPOT2's web interface. Make sure your radio is sending the call to the desired talkgroup or reflector.

The openSPOT2's LED should blink green when you press the PTT on your radio:



If not, then make sure the openSPOT2 and your radio are on the same RX and TX frequencies.

Make sure the openSPOT2's modem is set to the same mode as your radio, and the following settings are set:

- **DMR radio:** the color code (CC) match in your radio and the openSPOT2 (example: set it to the default 1).
- **C4FM radio:** the DGID is set to the same value both in your radio and the openSPOT2 (example: set it to the default 0).
- **NXDN radio:** the radio access number (RAN) match in your radio and the openSPOT2 (example: set it to the default 0).

If you are using a **D-STAR radio**, then make sure your D-STAR network registration is valid on [this](#) page. If you use a Kenwood TH-D74A transceiver, then make sure the *Direct Reply* feature is turned off, as it basically causes automatic callsign routing, which could cause one's transmission to not be heard on the used reflector.

## The openSPOT2 can't connect to the selected server



If the LED is pulsating like this, then the openSPOT2 is trying to connect to the selected server, but it can't for some reason. If it's not pulsating like on the animation, check the [LED states](#) section for more information about the current LED state.

The server may be currently offline, or there may be a connection issue between your openSPOT2 and the server. You can try using another server, or enable the [Override DHCP DNS servers](#) setting.

Also check if the server password is set correctly (in case there's a server password), and that your connector settings are valid (example: if there's a callsign setting, make sure your callsign is valid, if there's a DMR/CCS7 or NXDN ID setting, make sure your ID is valid for your callsign).

If you want to connect to the BrandMeister network, and the openSPOT2 won't connect to any BrandMeister server, then make sure you use the hotspot security password you've set on your [BrandMeister SelfCare](#) page. If hotspot security is turned off on your BrandMeister SelfCare page, then you have to use the default server password, which is *passwOrd*. The UK server uses the default password *passw1rd*.

## There are dropouts in incoming calls, voice is stuttering

Make sure you use a server nearest to your location to avoid network dropouts and high latency. Please keep in mind, that packet loss and high latency can occur in VoIP (Voice over IP) streams even if you have a really high quality



network connection.

Check the dejitter queue graph on the Status page for invalid seqnum errors. These indicate that the openSPOT2 does not receive all network packets so there are missing parts from the incoming call's stream.

If there are no invalid seqnum errors, but the dejitter queue graph shows that the queue gets empty during a call, then you can try increasing the dejitter queue length at the [Network settings](#) section.

Try to set another Wi-Fi channel in your Wi-Fi network (see the next troubleshooting entry for more info). The currently used one may be overcrowded, which you may not notice during web browsing, as the web uses a protocol which resend missing data frames, but this can't be done in real time VoIP streams so the packet loss issue becomes instantly noticeable.

### **Wi-Fi disconnects (blue LED) or internet connection errors (red/blue LED)**

Try to set another Wi-Fi channel in your Wi-Fi network. The currently used one may be overcrowded. It is advisable to use Wi-Fi [channels 1, 6 or 11](#). Do not use the others as they only cause more interference. Also try placing the openSPOT2 closer to your Wi-Fi router / access point.

### **Color code mismatch warnings on the web interface**

Make sure the color code programmed for your radio's current channel is matching the openSPOT2's [color code](#). If they match, then probably there's an offset between the radio's TX frequency and the openSPOT2's RX frequency. Please try to use [AutoCal](#), or manually enter different frequencies in +-100 Hz steps at the RX offset on the [Modem](#) page.

### **The device can't check firmware upgrades, it shows retrying**

Make sure the NTP, HTTP and HTTPS outgoing traffic is not disabled in your router's firewall. You can also try turning the **Use DHCP server if available** option off.

### **Upgrade downloads are slow**

Downloading upgrades is a low priority task so it won't interfere with normal operation and also the device won't consume much power during the download.

### **I want to manually upgrade the firmware**

- Download [this](#) file to your computer
- Turn off the openSPOT2 by unplugging the USB cable
- Press and hold the button
- Plug the USB cable into the computer while holding the button pressed

- Keep the button pressed for 1-2 seconds after the USB cable has been plugged in, then release it
- The LED will become yellow and a flash drive will appear on your computer called BOOTLOADER
- Copy the previously downloaded file to this BOOTLOADER drive
- The LED will start blinking alternating red/green for about 2-3 minutes. After that it should be steady green
- The openSPOT2 will automatically reboot in 1 minute after the LED became steady green, but now you can unplug it safely and then replug it to power on the openSPOT2 and continue using it (if the Wi-Fi module firmware is also upgraded then the LED will flash red/yellow for about 2-3 minutes)

### **The device does not turn on, the LED is dark**

Probably the openSPOT2 was plugged out during a firmware upgrade. Press the button and the upgrade process will continue. If the LED is still dark and there's no way to get it light up, contact us at [info@sharkrf.com](mailto:info@sharkrf.com).

### **DMR ID is not in the database warnings on the web interface**

A valid DMR ID is only required by the network/server you use. The openSPOT2 only displays a warning about that and does not care if your ID is valid or not. We update the DMR ID database weekly, usually on Mondays. Wait for the next update and your DMR ID will be there.

## Specifications

- Dimensions: 85 x 48 x 15 mm
- Weight: 32 grams
- Operating temperature range: for indoor use, -10 - +60 °C
- Power supply: 5 V DC through the USB port
- Power consumption:
  - Max. 150 mA @ 5 V (0.75 W)
  - Avg. 115 mA @ 5 V (0.575 W)
  - Avg. 65mA @ 5 V (0.325 W) with enabled powersaving
- UHF modem:
  - RF power output: max. 13 dBm (20 mW)
  - Receive/transmit frequency range: 421-458 MHz (JP: 430-440MHz, TW: 430-432MHz)
  - Receiver sensitivity:
    - Gaussian 2FSK: -114 dBm @ 9.6 kbps
    - Raised Cosine 4FSK: -109 dBm @ 9.6 kbps
  - Oscillator frequency stability: 0.5ppm
  - Emission designators: A1A (CW ID), F1E/F1D (D-STAR, System Fusion, P25), FXE/FXD (DMR), F1E/F1W (NXDN), F1D (POCSAG)
- Wi-Fi module:
  - Supported protocols: IEEE 802.11b/g/n
  - RF power output: +15 dBm (802.11b 1Mbps), +14 dBm (802.11g 6Mbps), +14 dBm (802.11n MCS0)
  - Receive/transmit frequency range: 2.4 - 2.495 GHz
  - Receiver sensitivity: -91 dBm (802.11b 1Mbps), -88 dBm (802.11g 6Mbps), -88 dBm (802.11n MCS0)

This device meets the requirements of the following standards and directives: EN 55024:2010; EN 55032:2012; EN 61000-3-2:2014; EN 61000-3-3:2013; RoHS2:2015/863/EU; WEEE:2012/19/EU; 47 CFR Part 15, Subpart C



This device complies with Part 15C of the FCC Rules.

The data sheet which can be used to apply for usage permit in Japan is available for download [here](#).

## Acknowledgement

Thanks to our alpha testers for their continuous work: Lajos Horváth HA5OGR and István Száraz HA7PTY, and to everyone who use beta firmwares and help us finding bugs by reporting them on our [community forum](#)!

## Open source licences

Here are the licences for various open source software used in the openSPOT2.

### **alertify.js**

The MIT License

Copyright (c) 2009-2014 Stuart Knightley, David Duponchel, Franz Buchinger, António Afonso

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### **jquery.js**

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## **ladda.js**

Copyright (C) 2016 Hakim El Hattab, <http://hakim.se>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## **pure.css**

Copyright 2013 Yahoo! Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## **js-sha256.js**

Copyright (c) 2015 Chen Yi-Cyuan

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## **spin.js**

The MIT License Copyright (c) 2011-2015 Felix Gnass [fgnass at gmail dot com]

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## **sha2.c**

AUTHOR: Aaron D. Gifford – <http://www.aarongifford.com/>

Copyright (c) 2000-2001, Aaron D. Gifford All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTOR(S) "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## **heatshrink**

Copyright (c) 2013-2015, Scott Vokes (vokes.s@gmail.com) All rights reserved.

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

## **jsmn**

Copyright (c) 2010 Serge A. Zaitsev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy,



modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## HamGridSquare.js

Copyright 2014 Paul Brewer KI6CQ License: MIT License

<http://opensource.org/licenses/MIT> or CC-BY-SA

Javascript routines to convert from lat-lon to Maidenhead Grid Squares typically used in Ham Radio Satellite operations and VHF Contests

Inspired in part by K6WRU Walter Underwood's python answer

<http://ham.stackexchange.com/a/244> to this stack overflow question: How Can One Convert From Lat/Long to Grid Square

<http://ham.stackexchange.com/questions/221/how-can-one-convert-from-lat-long-to-grid-square>

## GetGitRevisionDescription.cmake

Original Author: 2009-2010 Ryan Pavlik [rpavlik@iastate.edu](mailto:rpavlik@iastate.edu),

[abiryan@ryand.net](mailto:abiryan@ryand.net) <http://academic.cleardefinition.com> Iowa State University HCI Graduate Program/VRAC

Copyright Iowa State University 2009-2010.

Boost Software License – Version 1.0 – August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Select box background image

Icons made by Google <https://www.flaticon.com/authors/google> from [www.flaticon.com](http://creativecommons.org/licenses/by/3.0/) is licensed by Creative Commons BY 3.0 <http://creativecommons.org/licenses/by/3.0/>

## jquery-ui.js

Copyright jQuery Foundation and other contributors, <https://jquery.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/jquery/jquery-ui>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code contained within the demos directory.

CC0: <http://creativecommons.org/publicdomain/zero/1.0/>

## **bpt.c**

Copyright (C) 2010-2016 Amittai Aviram <http://www.amittai.com> All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## **openssl**

Copyright (c) 1998-2018 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-

core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

## Smoothie Charts

Copyright (c) 2010-2013, Joe Walnes 2013-2018, Drew Noakes

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.