

Channel and Frequency Planning

- [Meshtastic Channel Number Selection Guide](#)
- [Operating Multiple Channels on One Network](#)

Meshtastic Channel Number Selection Guide

Meshtastic has two distinct concepts that are easy to confuse. The **LoRa Frequency Slot** (`lora.channel_num`) selects the single center frequency the radio transmits on. This is separate from the up-to-8 **logical Channels** (each with its own name and PSK, indexes 0-7), which **all share the same frequency**. Adding logical channels gives you separate encrypted message streams — it does **not** change your RF frequency, provide RF isolation, or avoid collisions. Only changing the frequency slot (or the modem preset) actually moves you to a different frequency.

How the Frequency Is Chosen

Meshtastic does **not** compute the center frequency from a logical channel number using a simple base + spacing formula. Instead, the firmware divides the region's band into a fixed number of **frequency slots** (slot width = the preset's bandwidth) and picks one slot:

- If `lora.channel_num` is **0 / UNSET** (the factory default), the firmware computes a **hash of the PRIMARY channel's NAME** (a djb2 hash in `RadioInterface.cpp`) and uses it to select a slot.
- If `lora.channel_num` is set to **1-104**, that value is an **absolute slot index** and overrides the name hash.

For the **US 902-928 MHz band with the LongFast preset, the bandwidth is 250 kHz**, which yields **104 frequency slots** of 250 kHz each across the 26 MHz usable band. (125 kHz is the bandwidth of the Long Moderate / Long Slow presets, not LongFast — those presets have more slots.)

```
# US region, LongFast preset
# Bandwidth (slot width): 250 kHz
# Number of frequency slots: 104

# Default channel_num = 0/UNSET -> frequency comes from a HASH of
# the PRIMARY channel NAME, NOT from a linear base+offset formula.
# The default LongFast channel name hashes to slot 20 = 906.875 MHz.

# Slot-to-frequency formula (only when you set an explicit slot):
```

```
# freq_MHz = freq_start + (slot * bandwidth) + (bandwidth / 2)
# US: freq_start = 902.0 MHz, bandwidth = 0.250 MHz, slot is 1-based (1..104)
#
# Example, default LongFast slot 20:
# 902.0 + (20 * 0.250) + 0.125 ... is reached via the name hash,
# giving the documented default of 906.875 MHz.
```

Note that **906.875 MHz corresponds to frequency slot 20** (the result of hashing the default channel name), *not* "channel 0." There is no `base 902.0 + channel × 0.125` linear progression, and the slot index is not limited to 0-7 — those are the logical channels, a completely separate concept.

Why Change the Default Slot?

Most Meshtastic nodes ship on the default LongFast channel, which hashes to slot 20 (906.875 MHz). If your area already has significant Meshtastic traffic, you may benefit from staying on the default to maximize connectivity with existing nodes. However, if:

- You're running a private community network that wants to operate on its own frequency (note: changing frequency means those nodes can no longer hear the default mesh, and vice versa)
- You've identified interference at the default frequency with an SDR
- You want to operate on a less crowded frequency in a dense metropolitan area

...then setting an explicit frequency slot (or choosing a different primary channel name that hashes to a different slot) makes sense. Changing logical channels alone will **not** do this — logical channels all share the same frequency.

Frequency Coordination

For all nodes in your community network to hear each other, they must transmit on the **same frequency** — meaning the same modem preset *and* the same frequency slot (which means either the same primary channel name, since the name-hash sets the slot, or the same explicitly set `lora.channel_num`). They must also share the same channel name/PSK to decode each other's traffic. Keep a community coordination document:

```
# Check current frequency slot:
meshtastic --get lora.channel_num
```

```
# Set an explicit frequency slot (absolute slot index, 1-104 for US LongFast):  
meshtastic --set lora.channel_num 3  
  
# This selects frequency SLOT 3, an absolute slot - NOT an offset added  
# to 906.875. For US LongFast (250 kHz):  
# 902.0 + (3 * 0.250) + 0.125 = 903.375 MHz  
#  
# Setting channel_num to 0 reverts to the channel-name hash (default).
```

Bandwidth and Frequency-Slot Separation

Two independent networks on the same band coexist by transmitting on **different frequency slots** (set via different slot numbers, or different primary channel names that hash to different slots). Separating two networks by ≥ 2 slots reduces adjacent-channel interference, but it does **not** give complete RF isolation — LoRa's spectral skirts extend beyond the nominal bandwidth and the 902–928 MHz band is shared ISM spectrum, so some interference is always possible. (Separate channel names/PSKs alone provide only *logical* separation — same frequency, same airtime, same collisions.) Practical guideline:

Modem Preset	Bandwidth (slot width)	Separation for substantial isolation
LongFast	250 kHz	≥ 1 slot (250 kHz); ≥ 2 slots (500 kHz) for better margin
MediumFast	250 kHz	≥ 1 slot (250 kHz); ≥ 2 slots (500 kHz) for better margin
ShortFast	250 kHz	≥ 1 slot (250 kHz); ≥ 2 slots (500 kHz) for better margin
ShortTurbo	500 kHz	≥ 1 slot (500 kHz); ≥ 2 slots (1000 kHz) for better margin

If two networks operate on adjacent slots with the same bandwidth, they'll still experience some adjacent-channel interference and won't be completely isolated. Remember that putting your network on a different frequency slot from the local default mesh means the two networks can no longer relay for each other — that's the intended trade-off when you want a private frequency.

Operating Multiple Channels on One Network

Running multiple Meshtastic channels on the same network infrastructure enables privacy separation, role-based access control, and operational flexibility. This page covers design patterns for multi-channel community networks.

Why Multiple Channels?

- **Privacy tiers:** Public channel (open access) + private community channel (member-only key) + infrastructure/ops channel (operators only)
- **Functional separation:** General chat, emergency/tactical, telemetry/MQTT, admin coordination
- **Geographic zones:** Neighborhood channels within a metro-wide network
- **Inter-network bridging:** A bridge node can monitor a second network's channel while primarily operating on your community channel

Channel Capacity

Meshtastic supports up to 8 channels simultaneously on a single node (slots 0-7). Channel 0 is the primary channel (all nodes have it). Channels 1-7 are secondary and only active on nodes configured to use them.

All enabled channels share the same LoRa radio and the same RF frequency - the node transmits and receives on one frequency at a time. Adding channels does not add spectrum; it adds logical (name/PSK) separation only. Multiple channels increase channel utilization proportionally with message volume on each channel, because every channel's traffic competes for the same airtime.

Recommended Multi-Channel Configuration

The commands below use the official Meshtastic Python CLI. `--ch-add` creates a new secondary channel, and `--ch-index N --ch-set <field> <value>` sets a field on the channel at that index (see the Meshtastic CLI reference at meshtastic.org/docs/software/python/cli/).

```
# Channel 0: Public community channel
meshtastic --ch-index 0 --ch-set name "PDXMesh"
meshtastic --ch-index 0 --ch-set psk "community-key-base64=="
meshtastic --ch-index 0 --ch-set uplink_enabled true # optional: MQTT bridge
meshtastic --ch-index 0 --ch-set downlink_enabled false # prevent loops

# Channel 1: Operations/admin (limited to infrastructure operators)
meshtastic --ch-add
meshtastic --ch-index 1 --ch-set name "PDX-Ops"
meshtastic --ch-index 1 --ch-set psk "ops-key-base64=="
meshtastic --ch-index 1 --ch-set uplink_enabled false # keep off MQTT
meshtastic --ch-index 1 --ch-set downlink_enabled false

# Channel 2: Emergency (activated during events)
meshtastic --ch-add
meshtastic --ch-index 2 --ch-set name "PDX-EmComm"
meshtastic --ch-index 2 --ch-set psk "emcomm-key-base64=="
```

Channel Key Management

With multiple channels, key management becomes important:

- Store all channel keys in a password manager (Bitwarden, 1Password) shared with authorized operators
- Document which nodes have which channels configured - a node without channel 2 will miss emergency messages
- When a key is compromised, change only the affected channel key - other channels are unaffected
- Establish a key rotation schedule: community channel annually, ops channel quarterly, emcomm channel pre-activation

Traffic Impact

Each additional active channel increases channel utilization by the traffic that channel carries — all channels share one radio and one frequency, so their airtime adds up. Monitor total CU across all

channels:

- With 3 channels, target total CU under 20% combined (channel utilization is reported over a 1-minute window; firmware self-throttles, deferring its own transmissions, as CU rises around 25%)
- Low-activity channels (ops, emcomm) add minimal overhead: mostly periodic NodeInfo broadcasts
- A busy community-chat channel can contribute a substantial share of total CU; the exact percentage depends on message rate, packet size, and modem preset, so measure it on your own network rather than assuming a fixed figure