Your First Node — Step by Step

Practical walkthroughs for getting your first LoRa mesh message through on Meshtastic and MeshCore, and how to connect with your local community mesh.

- Getting Your First Message Through: Meshtastic
- Getting Your First Message Through: MeshCore
- Connecting to Your Local Community Mesh

Getting Your First Message Through: Meshtastic

This guide will take you from zero to sending a LoRa mesh message on Meshtastic, step by step. No prior RF or networking experience is required. If you get stuck, the **What to Check** sections at the end will help you diagnose the most common problems.

Step 1: Buy the Right Hardware

You need a LoRa development board with the Meshtastic firmware flashed onto it. The two most popular beginner choices are:

- **LILYGO T-Beam (v1.1 or v1.2)** - a combined LoRa radio, GPS module, and ESP32 microcontroller with an 18650 battery holder. It is the classic Meshtastic node. Costs $25 - 40 on AliExpress or Amazon.
- **Heltec LoRa 32 v3** - a compact board with a small built-in OLED display, LoRa radio, and ESP32. No built-in GPS, but lighter and cheaper ($20 - 25). A good choice if you want a pocket-sized node.

Make sure the board you buy is rated for **915 MHz** (for the USA). This should say "915MHz" or "US915" in the product listing. A 868 MHz EU version will not work on US community meshes.

You also need a short USB cable (usually USB-C or micro-USB depending on the board) to connect the device to your computer for flashing.

Step 2: Flash the Meshtastic Firmware

Meshtastic provides a web-based flashing tool so you do not need to install any software.

1. Open **Google Chrome** or **Microsoft Edge** (the tool requires a browser with Web Serial support, and Chrome or Edge is what the Meshtastic project recommends). Desktop Firefox added Web Serial support in version 151 (2026), but the Meshtastic project does not yet recommend it; Safari does not support Web Serial at all.

2. Go to flasher.meshtastic.org.
3. Connect your LoRa board to your computer with the USB cable.
4. Click **Get Started**. The site will prompt you to select a serial port - choose the one that appeared when you plugged in your device (usually labeled "USB Serial" or "CP210x" or "CH9102").
5. Select your device model from the dropdown list.
6. Click **Flash**. The tool will download the latest stable firmware and write it to the device. This usually takes a minute or two.
7. When flashing is complete, the device will reboot automatically. You should see the Meshtastic boot screen on the OLED display (if your board has one) or a blinking LED.

If the serial port does not appear, you may need to install a USB-to-serial driver for your board's USB chip. Recent LILYGO T-Beams use a WCH CH9102 chip (install the WCH driver; some early units used a CP2104). The Heltec v3 uses a Silicon Labs CP2102 chip (install the Silicon Labs CP210x driver). Search for the chip name + "driver" to find the official installer.

Step 3: Install the Meshtastic App

Meshtastic is controlled through a phone app that communicates with the node over Bluetooth or Wi-Fi.

- **Android:** install *Meshtastic* from the Google Play Store (it is free and open source).
- **iPhone / iPad:** install *Meshtastic* from the Apple App Store.

There is also a web-based client at client.meshtastic.org if you prefer to manage the node from your computer.

Step 4: Pair via Bluetooth

1. Open the Meshtastic app and tap the **+** button to add a new device.
2. The app will scan for nearby Bluetooth devices. Your node should appear, named something like "Meshtastic_XXXX" (where XXXX is a short ID derived from your device's MAC address).
3. Tap on it. The app will prompt you for a pairing PIN. Look at the OLED display on your node - the PIN is shown there. Enter it in the app.
4. Once paired, the app will sync with the device and show you the main interface: a map, a messages tab, and a node list.

Step 5: Configure Your Region and Channel

Before the radio will transmit, you must set your region:

1. In the app, go to the device configuration (the wrench/gear icon).
2. Find **LoRa Config → Region** and set it to **US**.
3. Tap Save (or the checkmark). The device will reboot briefly.

By default, Meshtastic uses the **LongFast** channel, which is the standard public channel used by community meshes across the US. Leave it on LongFast for now.

Step 6: Send Your First Message

1. Tap the **Messages** tab in the app.
2. Tap **Primary Channel** (or the LongFast channel name).
3. Type a message in the text box at the bottom and hit Send.

Your node will transmit the message over LoRa. If you are alone and there are no other Meshtastic nodes within range, the message will still appear in your chat history - it was sent, but there was no one to receive it. To verify two-way communication, you need either a second node or a nearby community mesh member.

With only a single node there is no way to send a message "to yourself" over the radio - the app and the node are the same endpoint, so nothing meaningful is transmitted or received over RF, and Meshtastic acknowledgments come from *another* node rebroadcasting or a recipient ACKing, never from a node hearing its own transmission. To sanity-check a single isolated node, instead confirm that the region is set, watch the channel-utilization / airtime statistics in the app to see the radio is active, and look for any nearby community node to appear in the node list. A genuine end-to-end radio test requires a second device or a nearby mesh member.

Step 7: Join the LongFast Channel

The LongFast channel is the default public channel. If there are other Meshtastic users within a few kilometers (or connected through relay nodes), your messages will reach them and theirs will reach you. The node list in the app will show you who is currently visible on the mesh.

What to Check If Nothing Works

The serial port never appeared in the flasher
: Install the USB-to-serial driver for your board's chip: the CH9102 (WCH driver) for recent T-Beams, or the CP210x (Silicon Labs driver) for the Heltec v3 and older T-Beams. On Windows, check Device Manager for a yellow warning icon on an unknown USB device.

Flashing failed with an error
: Some boards need to be put into "bootloader mode" before flashing. Hold the BOOT button while pressing RESET (or while plugging in the USB cable). Then try flashing again.

The Bluetooth device doesn't appear in the app
: Grant the app its requested Bluetooth permissions (on Android 11 and earlier, Location permission was also required for BLE scanning; on Android 12 and later, scanning uses the BLUETOOTH_SCAN permission and Location is not inherently required). The app may still ask for Location for position features. Try toggling your phone's Bluetooth off and back on.

Messages show "Waiting to send" indefinitely
: The region is not set. Go to LoRa Config and set it to US. The radio will not transmit until a region is configured.

Messages sent but no acknowledgement received
: This is normal if you are alone on the mesh - there is no one to acknowledge. Try moving outdoors and away from buildings for better range. Check meshmap.net (next chapter) to see if there are community nodes near you.

Getting Your First Message Through: MeshCore

MeshCore is a newer peer-to-peer LoRa mesh platform that differs from Meshtastic in some important ways. This guide walks you through buying hardware, flashing firmware, and sending your first message - including a plain-language explanation of what makes MeshCore distinct from a beginner's perspective.

Step 1: Buy Compatible Hardware

MeshCore runs on many of the same LoRa development boards as Meshtastic, including:

- **Heltec LoRa 32 v3** - compact, inexpensive ($20 - 25), has a built-in OLED display.
- **LILYGO T-Echo** - a sleek device with an e-ink display and GPS, popular for MeshCore "client" nodes.
- **LILYGO T-Deck** - has a built-in keyboard, making it usable without a phone for messaging.
- **RAK WisBlock** - modular system popular for building infrastructure repeater nodes.
- Most boards using the **nRF52840** or **ESP32 + SX1262** chip combination.

As with all North American mesh hardware, make sure your board is rated for **915 MHz**. Check the product description for "915MHz", "US915", or "ISM 915".

Step 2: Flash MeshCore Firmware

MeshCore provides its own web-based flashing tool:

1. Open **Google Chrome** or **Microsoft Edge**.
2. Go to flasher.meshcore.io.
3. Connect your board to your computer via USB.
4. Click **Connect** and select the serial port for your device.
5. Select your board model from the device list.
6. Select the firmware variant you want. For most beginners, choose the **Companion** (client) variant - this is the mode you use when connecting a phone app to the node over Bluetooth - or **Repeater** if you are building a fixed relay node. (Note: the exact label for

the companion/client variant can vary slightly between the flasher and app versions.)
7. Click **Flash** and wait for the process to complete (usually a minute or two).

If the serial port does not appear, the same driver fix applies as for Meshtastic: install the CP2102 or CH340 driver depending on your board's USB chip.

Step 3: Install the MeshCore App

- **Android:** Search for *MeshCore* on the Google Play Store, or sideload the APK from files.liamcottle.net/MeshCore (the official download location linked from the MeshCore FAQ).
- **iPhone / iPad:** Search for *MeshCore* on the App Store.
- **Web client:** Available at app.meshcore.nz.

Step 4: Configure the USA/Canada Preset

MeshCore uses channel presets to configure frequency, spreading factor, and bandwidth. For North America, you want to set the region/frequency to North America (915 MHz):

1. Open the MeshCore app and connect to your node via Bluetooth (the pairing process is similar to Meshtastic - the node's name will appear in the Bluetooth scan list).
2. Go to the node settings (gear icon).
3. Find the **Channel** or **Radio Config** section.
4. Select the North America / US-Canada (915 MHz) preset. The exact wording varies by app version. This configures the node for 915 MHz operation with appropriate spreading factor and bandwidth settings for community mesh use.
5. Save the settings. The node will apply the configuration and restart the radio.

Step 5: Pair and Send Your First Message

1. In the MeshCore app, go to the **Contacts** or **Chat** tab. Nearby nodes appear in Contacts after they send an advert - so your own node will not appear as a contact, and you need at least one other node in range to populate the list.

2. To test with just a single device, post a message to a public channel. To test the radio path end-to-end (a direct message), you need a second node or another MeshCore user within range.
3. Type a message and tap Send. Your node will transmit it. You will see it appear in the conversation.

Because MeshCore uses a more structured routing approach than Meshtastic, it distinguishes clearly between **direct messages** (private, sent to a specific node), **channel messages** (broadcast to everyone on the same channel), and **room server messages** (routed through a dedicated hub node). For your first test, a broadcast channel message is the easiest way to confirm your node is transmitting; to test a direct message you will need a second node or a nearby community node in range.

How MeshCore Feels Different from Meshtastic (Beginner's Perspective)

If you have used Meshtastic before, you will notice a few differences:

- **Explicit node roles.** In Meshtastic, every node implicitly relays for every other node. In MeshCore, you explicitly configure a node as a *client* (your personal node, does minimal relaying), a *repeater* (infrastructure node that forwards traffic), or a *room server* (a hub for group chat). This makes the network's behavior more predictable and reduces unnecessary transmissions, but requires a bit more intentional planning.
- **Different discovery model.** Meshtastic nodes automatically announce themselves and appear in a node list. MeshCore's discovery works slightly differently - you may need to add contacts manually or enable node broadcasting to see nearby nodes.
- **Stronger encryption by default.** MeshCore uses elliptic-curve Diffie-Hellman key exchange for direct messages, giving each conversation unique encryption keys. Note that Meshtastic also uses public-key cryptography for direct messages (since firmware 2.5.0); both systems use shared keys for channel (group) messages, so the comparison is really about implementation details rather than one being broadly "more robust" than the other.
- **Less plug-and-play for newcomers.** Meshtastic has more polished beginner documentation and a larger US-based community, so it is often easier to get your first message through quickly. MeshCore rewards the extra learning investment with a more controlled network architecture.

What to Check If Nothing Works

The flasher does not detect the device
: Try a different USB cable (many cables are charge-only and lack data wires). Try a different USB port. Install the USB-to-serial driver for your board's chip.

The app cannot pair with the node via Bluetooth
: Make sure the node is fully booted (wait a few seconds after flashing for it to finish booting). Grant the app Bluetooth and Location permissions on Android. If pairing fails, put the node in pairing mode if required (consult the MeshCore documentation for your board).

Messages are queued but not transmitted
: If your radio settings (frequency / spreading factor / bandwidth) don't match your region's preset, you won't be able to hear or reach anyone - verify the North America (915 MHz) preset is applied.

You cannot find other MeshCore users nearby
: MeshCore has fewer deployed nodes than Meshtastic in many areas. Check the MeshCore Discord (linked in the app's help section) for your region's activity level, or invite a friend to flash a second node for testing.

Connecting to Your Local Community Mesh

Having a single node transmitting into the void is technically functional, but the real magic of a mesh network only appears when you are connected to other people. This page explains how to find out whether there is already an active mesh in your area, what to do if there is not, and how to behave as a good citizen when you join an existing community network.

How to Find an Existing Local Mesh

meshmap.net

meshmap.net is a community-maintained map of active Meshtastic nodes worldwide. Nodes that are configured to share their GPS position (or have a manually set location) appear as pins on the map. Zoom into your city or region and see if there are any active nodes nearby. A cluster of pins usually indicates an active local community.

Note: meshmap.net only shows nodes whose packets reach the public MQTT server - either because the node itself uplinks via MQTT or because a nearby MQTT-connected node heard it over LoRa - and that share a position. There may be active mesh users in your area who simply are not showing up on the map. The absence of pins does not necessarily mean no one is on the mesh.

Local Ham Radio Clubs

Many early adopters of LoRa mesh technology are also licensed hams. Some ham radio clubs run Meshtastic or MeshCore infrastructure as an emergency communications resource. Search for your nearest amateur radio club and check their website or newsletter for mentions of "LoRa mesh", "Meshtastic", or "digital emergency comms." Many clubs hold regular nets or meetings where you can meet local mesh operators in person.

Reddit Communities

Two subreddits are particularly useful for finding local activity:

- **r/meshtastic** - the primary community for Meshtastic users. Post a message asking if there are users in your metro area. Many local community threads exist.
- **r/MeshCore** - the community for MeshCore users. (For broader mesh-networking discussion, look for an active general mesh subreddit and confirm it is current before posting.)

Search for your city name within these subreddits before posting - there may already be a thread or weekly check-in from your area.

Discord Servers

Both Meshtastic and MeshCore have official Discord servers:

- The **Meshtastic Discord** has community channels including regional discussion. It is searchable and very active.
- The **MeshCore Discord** is smaller but growing, with channels for regional coordination.

Many local groups also run their own Discord servers. Searching Discord for "[your city] + mesh" or "[your state] + Meshtastic" often turns up active local servers.

Facebook Groups and Nextdoor

In some regions, mesh community organizing happens on Facebook rather than Reddit or Discord, particularly in less tech-oriented areas. Searching Facebook for "Meshtastic [your state]" or "LoRa mesh [your city]" may surface local groups.

What to Do If There Is No Local Mesh Yet

If you look around and find no existing local community - do not be discouraged. Someone has to go first. Here is how to start a mesh from nothing:

1. **Get two nodes running.** Even a two-node mesh is a functioning network. Ask a friend or family member to flash a second device and test with you.
2. **Put a node somewhere high.** A well-placed node with clear line of sight on a rooftop, hilltop, or tower can dramatically extend the mesh's reach - a well-placed node can cover several kilometers, but actual range depends heavily on height, terrain, and obstructions. **Before placing a node on any rooftop or tower, get the property owner's written permission, confirm any insurance the site owner requires, and never climb a**

tower without proper training and equipment - hire a qualified climber for tower work.

3. **Post on local channels.** Post in your local Reddit, Nextdoor, or Facebook group. "Anyone else in [city] on Meshtastic?" is a simple, effective opener.
4. **Contact your local ARRL club.** Ham radio operators often have the infrastructure access (towers, power, internet backhaul) to anchor a community mesh and an existing interest in emergency communications.
5. **Be patient.** Most thriving mesh communities started with one or two people who set up a few nodes and told their friends. Growth is gradual.

Network Etiquette: How to Be a Good Mesh Citizen

Mesh networks are a shared commons. The channel bandwidth is limited, every transmission affects everyone nearby, and the community is built on mutual goodwill. Follow these guidelines when joining or operating on an existing network:

Do Not Spam

Resist the urge to send test messages every few minutes. Every transmission you send uses up channel airtime for everyone on the mesh. Send messages when you have something to say, not just to confirm the radio is working. If you need to test, use the Traceroute feature sparingly instead of broadcasting text messages - but remember that all test traffic still uses airtime too.

Match the Community Preset

Community meshes converge on a shared channel preset (most commonly Meshtastic's **LongFast** or a locally agreed preset). If you use a different frequency, spreading factor, or channel name, your node will not be able to communicate with others - it will be on a different "frequency" even if physically nearby. Check with local mesh operators or look at meshmap.net to confirm what preset your community uses before you configure your node.

Set a Sensible Node Name

Your node's short name and long name are visible to everyone on the mesh. Give your node a recognizable name - your callsign if you are a ham, or a memorable handle. "Node-1234" is anonymous and unhelpful. "KD9ABC-Home" or "TJ-Backpack" tells people who to contact if they have questions or want to connect.

Introduce Yourself

If there is an active local community, send a brief introduction over the mesh when you first connect: your name (or handle), your rough location, and what you are interested in. Most mesh communities are welcoming and will appreciate knowing a new node has joined.

Do Not Enable Router Mode Unless Needed

In Meshtastic, you can configure your node with the **Router** role (the older "Router and Client" role was removed in firmware 2.3.15). This causes the node to forward more traffic and with higher priority than a regular node. Only enable this if you have an elevated, well-connected node specifically intended to serve as network infrastructure. Running router mode on a mobile or indoor node often creates more interference than benefit.

Respect Private Channels

If you learn of a private channel key being used by a specific group, do not join that channel unless you are invited. Private channels are used for coordinated groups (emergency response teams, event staff, hiking clubs) who need a clean channel away from public traffic.

Building Toward a Resilient Regional Mesh

The long-term vision of community mesh networks is a resilient communications layer that functions independently of commercial infrastructure. Every node you add, every hilltop you reach, and every neighbor you recruit brings that vision closer to reality. The best meshes are built by communities - not by any single organization - and they grow through the same word-of-mouth enthusiasm that brought you here.

Welcome to the mesh.