

Ham Signaling

© Unixeer.com. Ham Signaling Project. A41UX

Introduction.....	2
The Modes Screen.....	2
Settings.....	3
Operating the Modes	3
Themes & Accessibility.....	6
Support & Privacy.....	6
About & Credits.....	7

Introduction

Ham Signaling is Unixeer's digital-modes companion for amateur radio: an app to decode and transmit the open digital modes that fill today's HF and VHF bands, using the audio from your radio. Point your rig's receive audio at your phone and Ham Signaling turns the tones into text; type and it keys the mode back out to answer. Fourteen modes ship in this build — FT8, FT4, JS8, WSPR, PSK31, PSK63, PSK125, RTTY, Hellschreiber, Olivia, Contestia, APRS, AX.25 Packet and Messaging. Every one of them is open — no proprietary codecs and no licensing walls. The work is signal processing, not paperwork.

What this build is

This is the first release of Ham Signaling. Every mode listed above transmits, and every one except the WSPR beacon also receives: pick a mode on the Modes screen, press Initiate, and its operating console opens with a live waterfall, a scrolling decode list and a message box. Around them sit the standard About, Support, Manual, Version-history and Licences & Notices pages, the family theme, weekly tips and error handling. Optional rig control reads your radio's dial over CAT so logged contacts carry their true frequency.

Open modes only

Ham Signaling deliberately covers only open amateur modes. It is not an image-mode app: SSTV and other picture modes live in Ham Imaging, and Ham Satting handles satellite SSTV, so those are intentionally absent here rather than duplicated.

You must be licensed

Transmitting any of these modes will put real signals on the amateur bands. Ham Signaling is an operating tool, not a licence — you must hold a valid amateur radio licence and stay within your privileges whenever you transmit. This applies in full to the automatic modes described later: when the app answers or calls for you, you are still the control operator and every transmission it makes is made under your callsign and your responsibility.

The Modes Screen

The Modes screen is the app's home and where you operate. At the top, the Mode dropdown lists the digital modes you have switched on in Settings → Digital modes, beside the Initiate button. Until you start, the space below shows the selected mode's tutorial — how to use it, best practice and worked examples. Press Initiate and that mode's operating console opens right there, in place of the tutorial.

Initiate and Terminate

Initiate opens the console for the selected mode; press Listen in the console to start hearing, and Stop to pause. Terminate closes the console and releases the microphone. Only one console runs at a time, which keeps the audio path and the rig connection unambiguous. While you are

operating, an (i) in the top bar reopens that mode's tutorial, so the how-to is always one tap away.

When the list is empty

If you turn every mode off, the Modes screen shows a short hint pointing you to Settings → Digital modes to enable the modes you operate. Turn any mode back on and it reappears here immediately.

Settings

Settings is organised into three panels plus the resets. Everything auto-saves the moment you change it — there is no Save button — and every value is stored only on your device.

Your Details

The first panel holds your operator details, the app's look and feel, and how the app transmits. Enter your callsign and Maidenhead grid locator (both optional; they tag your decodes, fill the CQ the app suggests, and pre-fill the Support form) and a contact email. Below them are the appearance controls: a font-size slider, a dark-mode switch, a weekly-tips switch, and a Waterfall theme dropdown offering the Ocean, Classic, Green and Matrix palettes. Then comes “Keep decoding in background”, and under it the Operating mode choice described in “Auto-QSO sequencing” below.

Digital modes

The second panel lists every mode Ham Signaling ships — FT8, FT4, JS8, WSPR, PSK31, PSK63, PSK125, RTTY, Hellschreiber, Olivia, Contestia, APRS, AX.25 Packet and Messaging — each with its own on/off switch and its own tutorial. Turn on the modes you operate and they appear on the Modes screen. A “Reset digital-mode settings” button at the foot of this panel turns every mode back on and clears their per-mode settings. The third panel is Rig control (CAT), described further on.

Resets and storage

A separate “Reset all settings to defaults” button below the panels restores every app setting — callsign, grid, email, theme, font size and tips — to its default. Both resets ask for confirmation first and cannot be undone. Nothing you enter in Settings ever leaves the device: there is no account and no Unixeer server.

Operating the Modes

This section describes how operating actually works, screen by screen. Everything here is in the app now.

Audio in from the rig

You feed your radio's receive audio to the app either acoustically — the phone's microphone near the radio's speaker — or over a cable or USB audio interface, the same way desktop digital-mode software takes soundcard audio. A cable is the better path: it is quieter, it is unaffected by the room, and it gives the weak-signal modes several more decibels to work with. Transmit goes back out the same way, through the phone's speaker or that cable, and your radio is keyed by its own VOX or by your hand — or, if you turn it on, over CAT.

A live waterfall

A scrolling waterfall shows the received passband, coloured with the palette you chose in Settings, with a zoom for the HF audio window. Tap the waterfall to tune: on RTTY and PSK the decoder locks onto the signal you tapped and follows its drift by itself, while on FT8, FT4 and JS8 the tap places your transmit offset in a clear part of the band.

A decode list

Below the waterfall, decoded traffic scrolls in a list — callsigns, grids, reports and messages as each mode produces them, timestamped and ready to work. Your own transmissions echo into the same list, colour-coded, each with a one-tap re-use. A reply button on a decode fills the message box with the standard next transmission.

Hellschreiber is the exception: it has no decode-to-text, so its console paints the received dots and you read the letters by eye. When the two stations' clocks differ slightly the text leans; drag the Slant control under the picture until the letters stand upright, and its reset button re-centres it. The setting is remembered.

On FT8 and FT4 each line also carries two numbers before the message: the signal report in dB, and DT — how many seconds that station started after the top of the slot, by this phone's clock. DT is worth watching. One station with an odd DT is that station's problem; when every station you hear shares the same offset, the clock that is wrong is yours, and the app will say so under the status line. It matters because a drifted clock fails silently: you decode everyone perfectly, your own transmissions land outside their windows, nobody answers, and the band merely looks unlucky.

If you key the radio by hand rather than with VOX or CAT, watch the line above the message box. The app puts half a second of silence at the front of every transmission so that keying EARLY costs nothing — press PTT as the countdown reaches zero, not when you hear the tones begin. By then the cushion is spent, and the sync pattern the far end needs to lock on has already been clipped.

Auto-QSO sequencing

For the weak-signal modes — FT8 and FT4 only — Ham Signaling can run a whole contact for you. The choice lives in Settings → Your Details, under “Keep decoding in background”, as three options: Manual, Auto Hunt and Auto CQ.

Manual is the default and is exactly how the app behaves everywhere else: nothing is transmitted unless you type a message and press Send, or tap Reply on a decode and then Send.

Auto Hunt answers a station calling you, or answers someone else's CQ, and then carries the exchange to the end by itself — signal report, R-report, RR73, 73 — and writes the contact to the log, with nothing typed. It works one station at a time and picks the strongest one it can work. It never calls CQ itself.

Auto CQ calls CQ and works whoever comes back to it. It answers replies to your own CQ, but does not chase other stations' CQs — choose Auto Hunt for that.

Both automatic modes stop rather than run on: they give up on a station that stops replying, abandon an exchange that stops making progress, will not work the same station twice within the hour, and stop calling after a few unanswered CQs rather than calling into a dead band. The console shows in plain words what the app is doing and why — which station, and the exact message it is about to send — and you can return to Manual at any moment, which stops an exchange in flight immediately.

You remain the control operator. The app transmits only while an automatic mode is selected and the console is listening, and it keys your radio exactly as you already do.

ADIF logging into Ham Logging

Completed contacts are captured automatically on FT8 and FT4, and a bookmark button logs RTTY, PSK, Olivia, Contestia, Hellschreiber, JS8, Packet and Messaging contacts by hand. APRS is a monitor, so a heard beacon is never logged as a contact. The whole log exports as an ADIF file you can share straight into Ham Logging — or any other logger — carrying the frequency and band when rig control is connected.

Rig control (CAT)

Ham Signaling can read your radio's dial frequency and mode over CAT, so every logged contact carries the frequency it really happened on. It connects over your local network (Hamlib rigctlId, Icom WLAN, Kenwood KNS, Elecraft K4, TCI, or a serial-over-TCP bridge), over Android USB-OTG, or over Bluetooth. A searchable picker covers around ninety radios and labels each one honestly — verified on real hardware, or implemented per specification.

Ham Signaling never changes a setting on your radio. It does not touch your frequency, your mode or your split, and there is no option that makes it do so. The one thing it can write is PTT: on the rigctlid connection you may set Transmit keying to CAT PTT, and the app will then key and unkey your radio for each transmission instead of relying on VOX. It refuses to key a radio that is already transmitting, releases it when a transmission ends and when you leave a console, and releases it on a time limit if anything goes wrong. If the app is force-quit while transmitting it cannot release the radio until you open it again, so keep your radio's own time-out timer set. The default is VOX, and with the default nothing is ever sent to your radio.

Themes & Accessibility

Ham Signaling ships the family's light and dark themes on the standard ocean-blue accent. Toggle dark mode from Settings → Your Details; the change applies instantly across every screen.

Font scaling

The font-size slider scales text across the whole app from 80% to 140%, so you can size the interface for a bench display or a bright field day without leaving the app.

Waterfall themes

The Waterfall theme dropdown — Ocean, Classic, Green and Matrix — sets the colour scheme used by the decode waterfall on every operating console.

Support & Privacy

Support

The Support page sends a message to the developer through your own email app, pre-filled with your callsign, email and the app version so a report carries the context it needs. You can reach Unixeer at unixeer@gmail.com.

Privacy

Everything Ham Signaling stores — your callsign, grid, email and settings — stays on your device. There is no account, no Unixeer server, no analytics and no ads.

Microphone

Ham Signaling uses your microphone only to listen to your radio's audio so it can decode signals — never for anything else. On the air, your transmissions are, as always, public amateur radio.

About & Credits

About

The About page shows the current version, a summary of the app, and links to this manual, the Support form, the version history, the Licences & Notices page and the Privacy Policy.

Credits

Ham Signaling is part of the Unixeer family of amateur-radio apps by Yousuf AL Balushi, A41UX. Unixeer™ — RF for Life. The FT8 and FT4 engine is built on ft8_lib by Kārlis Goba, the JS8 modem on fate by Robert Morris, AB1HL, and both use KISS FFT by Mark Borgerding; their licences are reproduced in full on the Licences & Notices page in the app.

Ham Signaling © 2026 Project. A41UX. All rights reserved, Unixeer™.