Ham Cockpit

Radio Amateur's integrated envoronment

User's Guide

Copyright (c) 2020 Alex Shovkoplyas VE3NEA

Please visit the Ham Cockpit web site for general information about **Ham Cockpit**.

Table of Contents

Installation

To install *Ham Cockpit*

- download the setup package
- double-click on the downloaded zip file to open it in File Explorer
- double-click on the HamCockpitSetup.exe file in the zip
- follow the on-screen instructions

Play with the program and get familiar with its functions. Once you feel that you need more plugins, proceed to installing the additional plugins package.

To install additional plugins

- download the additional plugins package, HamCockpitPlugins.zip
- right-click on the downloaded zip file, click on *Properties* in the popup menu, tick the *Unblock* checkbox, and click on *OK*.
- double-click on the zip file to open it in File Explorer
- extract all files from the zip to the folder where *Ham Cockpit* is installed
- start *Ham Cockpit*, click on *View/ Plugin Settings* in the menu and configure the installed plugins.

To uninstall *Ham Cockpit*

On Windows 10

- right-click on the *Start* button
- click on *Apps and Features* in the popup menu
- select *Ham Cockpit* in the list of installed programs
- click on the *Uninstall* button

Configuration

Virtually all functions in *Ham Cockpit* are implemented in the plugins. To configure the program, just configure the settings of the installed plugins.

Click on the *View/ Plugin Settings* command in the menu to open the *Plugin Settings* dialog.
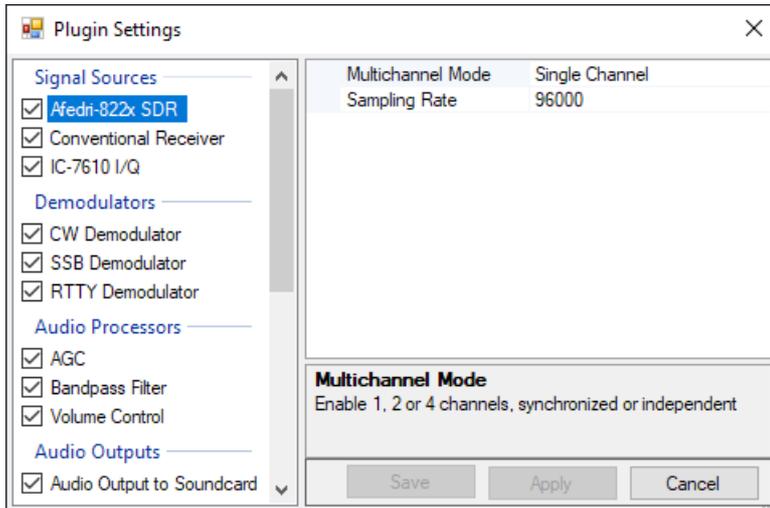
The left panel of the *Plugin Settings* dialog shows all installed plugins. Click on the plugin to select it for configuration and edit its settings on the right panel. Click on the *Save* button to save all changes and close the dialog. The *Apply* button saves the changes but does not close the dialog.

To reset the setting to its default value, right-click on the setting name and click on Reset in the popup menu.

Click on the checkbox next to each plugin name to enable or disable the plugin. Another way to disable a plugin permanently is to simply delete its DLL file from the installation folder.

For some plugin types the order is important. For example, *Audio Processors* are applied to the signals in the same order in which they appear in the *Plugin Settings* dialog. Drag and drop such plugins to change their order.

⬜ **Warning**

Some of the plugins that come with *Ham Cockpit* perform audio input and output, control the radio via a USB or RS-232 port, or work with other non-standard hardware in various ways. Be sure to configure them properly before you start using them.

DSP Pipeline

The plugins that perform digital signal processing are organized in a *DSP Pipeline*.

When working with the SDR radios and other devices that produce wideband quadrature (I/Q) signals, the pipeline looks like this:

- **Signal source** receives I/Q data from an external device and controls the radio settings. Example: IC-7610 I/Q plugin.
- **I/Q processors** process I/Q data received from the signal source.
- **Demodulator** converts I/Q data to the audio suitable for listening.
- **Audio processors** process demodulated audio in different ways. Examples are Bandpass Filter and AGC plugins.
- **Audio Output** sends processed audio to a soundcard or another device.
- **Transmitter** controls the transmit frequency, Split mode and T/R switch in the transmitter.

If the radio outputs already demodulated signals, such as audio output of the conventional transceivers, the pipeline includes only a subset of stages:

- **Signal source** receives audio data from an external device and controls the radio settings. Example: Conventional Receiver plugin.
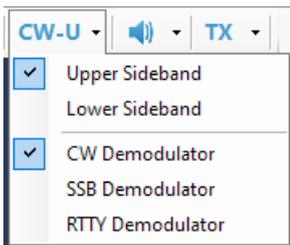- **Audio processors**
- **Audio Output**
- **Transmitter**

The *DSP Pipeline* is represented visually with a toolstrip on the program toolbar. By clicking on the toolbar buttons, the user can start and stop the signal source, turn the audio on and off and switch the radio to the RX or TX mode.

The drop-down lists attached to the buttons allow the user to select the signal source
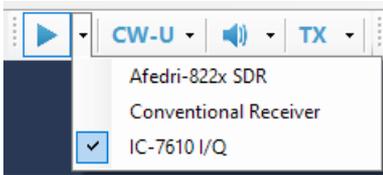
demodulator

and transmitter
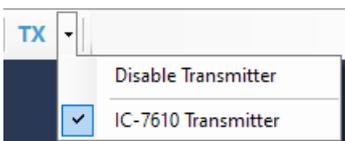
In case of an audio data source, the program receives audio signals already demodulated in the receiver. When such signal source is selected, the demodulator plugins are not used, and the Mode dropdown list has commands to change the mode in the radio via its CAT interface:

CW-U ▾ | 🔊 ▾ | TX ▾

✓	Upper Sideband
	Lower Sideband
✓	CW
	SSB
	DIG

Docking

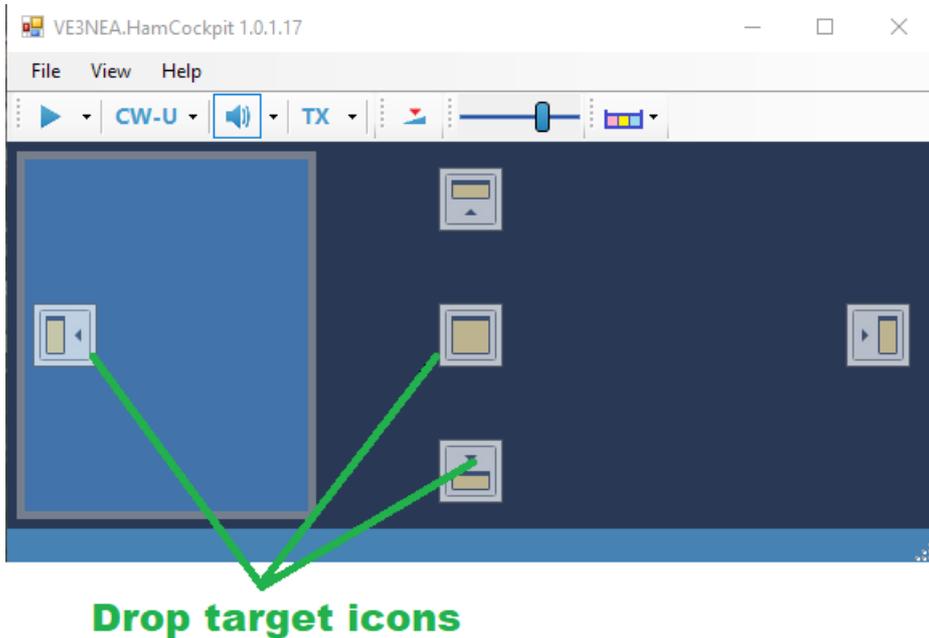Visual plugins installed in *Ham Cockpit* may be opened and closed using the *View* menu and arranged in the desired layout using drag-and-drop. The visual panels may be docked inside the docking area of the main program. To dock a panel, drag it to the main window with a mouse and drop it on one of the drop target icons that appear when the panel is over the docking area:

The panels may be arranged in many different ways, play with the docking system to see what is possible:
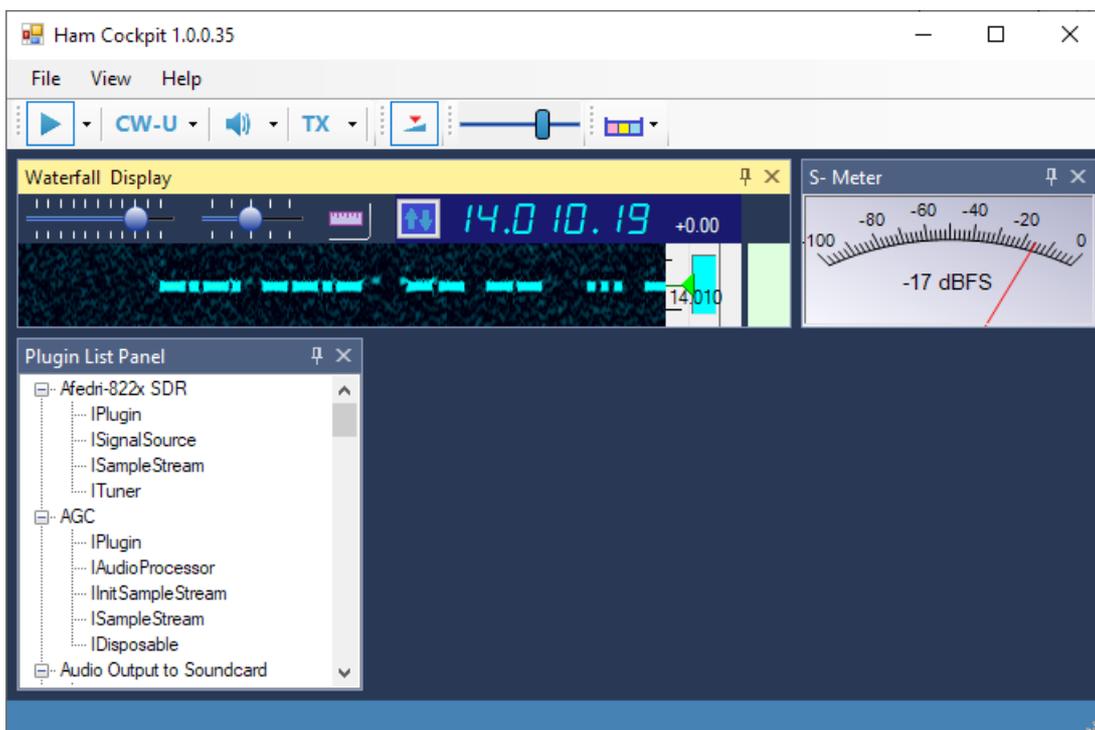
The plugin panels may be left floating if desired:
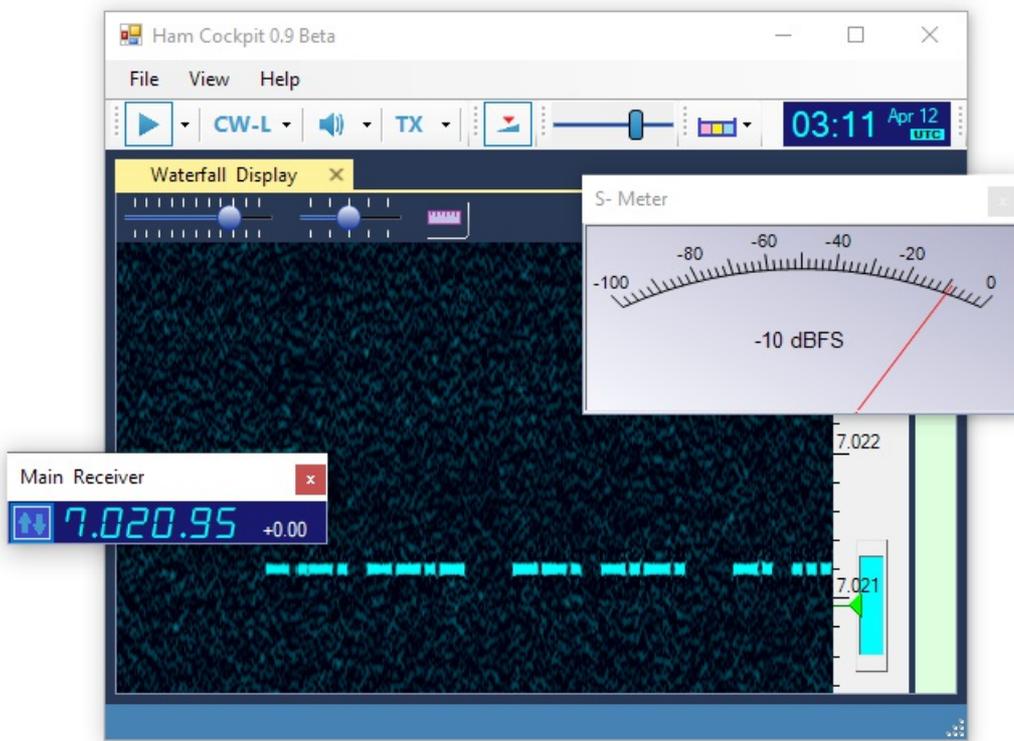
The program saves the panel layout on exit and restores it on the next start.

Afedri-822x SDR Plugin

Purpose

This plugin is an interface to the Afedri-822x receiver. Works with the single, double and quad channel models. Only the Ethernet connection is supported, not the USB connection.

Settings

- **Multichannel Mode**

  The plugin allows starting the radio in the single, double and quad channel modes if supported by the hardware. However, currently there are no plugins that could make use of multiple data streams provided by this plugin, so it is recommended to use the single-channel mode even if the radio has two or four channels.

- **Sampling Rate**

  This radio supports a discrete set of sampling rates. If an unsupported sampling rate is entered, it is replaced with the nearest supported value. 96 kHz is recommended since currently there are no plugins that could benefit from a higher sampling rate.

Audio Output to Soundcard Plugin

Purpose

This plugin sends the audio processed in the program to the soundcard for listening. It is recommended to use a pair of headphones instead of a speaker, especially when copying weak signals.

Settings

- **Latency**

  Experimental. Keep the default value.

- **Soundcard**

  The soundcard to which the audio is sent.

Bandpass Filter Plugin

Purpose

This plugin implements a digital bandpass filter for the received audio. The bandwidth of the filter is adjustable by the user.
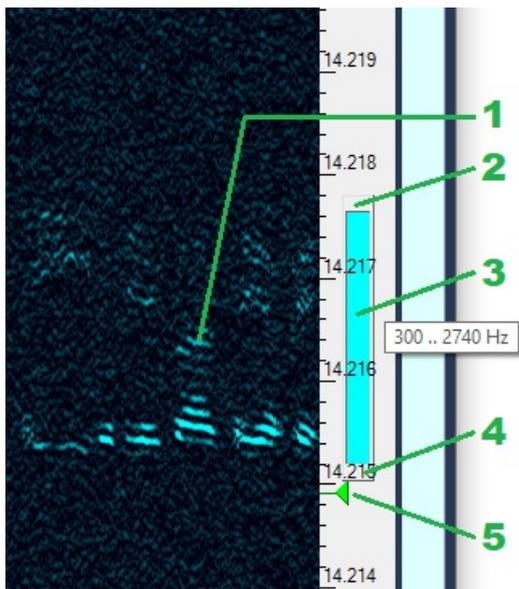
Settings

- **Latency**

  There is a trade-off between the sharpness of the filter skirts and the delay introduced by the filter. Increase the latency for a sharper filter, or decrease it for faster response. The default latency is 10 ms.

Visual Representation

Some plugins, such as Waterfall Display and *Band Scope*, provide visual representation of the filter and allow the user to adjust the filter passband to match the operating conditions. On the screenshot below, the blue rectangle **3** shows the extent of the filter passband. You can change the upper and lower cutoff frequencies of the filter by dragging the edges **2** and **4** of the filter rectangle. Since the SSB signal **1** being received is also shown, it is easy to match the filter to the signal.

The green triangle **5** indicates the dial frequency of the radio. In the SSB mode it is the frequency of the suppressed carrier.

In the CW mode, only the bandwidth of the filter is adjustable, the passband is always centered at the dial frequency. The next screenshot shows a CW signal and an unmodulated carrier near it. You could get rid of the carrier by setting the filter passband to its minimum, 20 Hz, but the sound of such narrowband audio is not pleasant to the ear. Normally you want to have the filter only as narrow as is needed to suppress the interference. Using the visual filter control in the waterfall display, you can set the filter bandwidth in such a way that the upper cutoff is just below the carrier, to have nice audio and no interference.
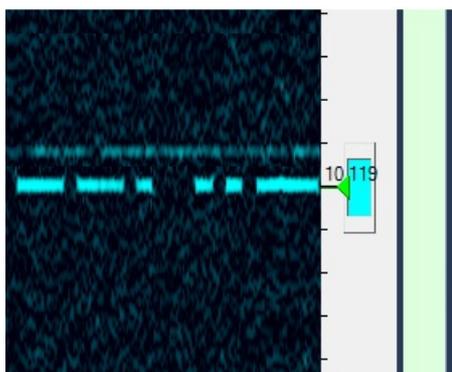
Band Plan Plugin

Purpose

This is a non-visual plugin that works behind the scenes and provides the band plan data to other plugins. In particular, the Frequency Display plugin receives information from this plugin when it needs to switch to the next or previous band, and the Waterfall Display plugin uses band plan information to paint the band map.

Settings

- **Mode Colors**

  Background colors for different Ham modes. Click on the ellipses button [...] to open the collection editor:
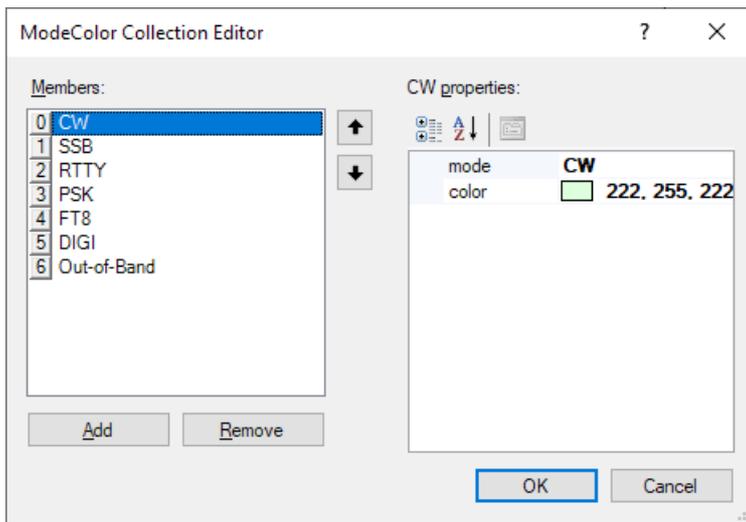
Add or remove the entries on the left panel, and set the mode name and color on the right panel. Click on the *OK* button to save the changes.

Band Plan Files

Band Plan information is stored in the JSON files located in this folder:

C:\ProgramData\Afreet\HamCockpit\VE3NEA\BandPlans

The users may create their own band plan files with a text editor and switch between them using a drop-down list on the Toolbar
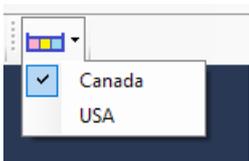
It is recommended to have at least two band plans, one for the regular operation and another for contesting, since frequency allocation by mode changes significantly during the contests.

Band Plan File Format

The band plan file contains a list of band segments. Each segment has these parameters:

- *Band* - the band name, preferably as defined in the ADIF format.

- *StartFrequency*, *EndFrequency* - the boundaries of the band, in Hertz.

- *DefaultFrequency* - when changing the bands, the radio will be initially tuned to this frequency. Assumed to be equal to *StartFrequency* if not specified.

- *PrimaryMode* - the main mode in the given frequency segment. Used, in particular, to infer the mode of a cluster spot from its frequency.

- *ValidModes* - all modes allowed in the segment. Used, in particular, by the skimmer plugins to determine where to decode a particular mode. If not specified, assumed to include only the primary mode.

- *MainSegment* - On some bands, there is more than one segment used for a particular mode. For example, the 14074-14078 kHz and 14095-14099 kHz segments are used for FT8 on the 20m band. One of these segments must be marked as main.

- *Remark* - optional text that will be appended to the segment label when information about the segment is displayed. For example, *"F/H"* is a remark in the 14095-14099 kHz segment because this segment is used for the Fox-and-Hound style QSO. The mouse tooltip on the band map is a combination of the band, mode and remark: *"20M FT8 (F/H)"*

Here is an example of segment definition:

```
{
  "Band": "40M",
  "StartFrequency": 7000000,
  "EndFrequency": 7025000,
  "PrimaryMode": "CW",
  "ValidModes": ["CW","RTTY"],
  "MainSegment": true,
  "Remark": "Only Extra class in the USA"
}
```

The json file is build of segment devinitions like this:

```
{
{...segment...},
{...segment...},
...
...
{...segment...}
}
```

Clock Demo Plugin

Purpose

The Clock Demo plugin shows a UTC or Local Time clock on the toolbar. Click on the *UTC* indicator to switch between the UTC and local time.

 Note

This plugin comes with source code to help plugin developers learn how to add their own controls to the toolbar.

Settings

- **Blink**

  enable or disable blinking of the ":" time separator

- **Dock to Right**

  move the clock to the right side of the toolbar:

Conventional Receiver Plugin

Purpose

This plugin provides an interface to a conventional receiver that outputs demodulated audio and has CAT control via an RS-232 port. Most radios used by the Hams fall in this category, including the ICOM, Yaesu, Kenwood and Elecraft transceivers.

 Note

This plugin makes use of the OmniRig engine. Make sure that you have the latest version of OmniRig installed on your system.

Settings

- **OmniRig Radio**

  Select to use Rig1 or Rig2 in OmniRig for CAT control.

- **Soundcard**

  Select the soundcard that will be used to input the audio signals from the radio.

- **Soundcard Channel**

  Specify which channel of the stereo soundcard to use.

- **CW, SSB and DIG Settings**

  Each of these settings has three parameters:

  - **Passband High, Passband Low** - the upper and lower cutoff frequencies of the filter in the radio.

  - **Dial Offset** - the audio tone that corresponds to the dial frequency of the radio.

- **Exlcusive, Latency**

  Experimental, do not change.

Explanation

The *Conventional Receiver* plugin inputs audio signals in the range of 0 to 6000 Hz, but most radios produce the audio that occupies only a part of this range. For example, the radios in the CW mode typically have a 500-Hz filter that suppresses all signals outside of this range. The *CW Settings*, *SSB Settings* and *RTTY Settings* parameters tell the program what part of the input spectrum is usable in each mode.

Other plugins use this information to show only that part of the spectrum on the waterfall, to decode signals only in that frequency range, etc.

Examples

1. In the CW mode, the CW Pitch setting in the radio is 800 Hz, and the filter bandwidth is 500 Hz.

2. The settings should be:

   - Dial Offset: 800 Hz
   - Passband Low: 800 - 250 = 550 Hz
   - Passband High: 800 + 250 = 1050 hz

3. In the SSB mode, the filter passband is 300 - 2700 Hz, and the radio displays the frequency of the suppressed carrier (0 Hz

audio). The settings should be:

- Dial Offset: 0 Hz
- Passband Low: 300 Hz
- Passband High: 2700Hz

Status Bar

The *Conventional Receiver* plugin shows an icon on the status bar that indicates the current status of the CAT control interface. Place the mouse cursor over the icon to see information on the tooltip. Click on the icon to open the OmniRig setup dialog.

Conventional Transmitter Plugin

Purpose

This plugin allows HamCockpit to use a conventional CAT-controlled transceiver as a transmitter. Most radios used by the Hams fall in this category, including the ICOM, Yaesu, Kenwood and Elecraft transceivers.

Any combination of the receiver and transmitter is allowed, for example you could use a wideband Afedri SDR as a receiver and your regular transceiver as a transmitter, or use the same transceiver as both receiver and transmitter. In either case, as long as the Split mode is disabled, the frequencies of the receiver and transmitter are synchronized both ways. In particular, this allows you to tune your SDR receiver by spinning the dial knob of your transceiver.

When Split is enabled, the receiver and transmitter frequencies are changed independently. If you are using the Waterfall Plugin, you can set the RX and TX frequencies by clicking on the waterfall display or on the band map.

 Note

This plugin makes use of the OmniRig engine. Make sure that you have the latest version of OmniRig installed on your system.

Settings

- **OmniRig Radio**

    Select to use Rig1 or Rig2 in OmniRig for CAT control.

Status Bar

The *Conventional Transmitter* plugin shows an icon on the status bar that indicates the current status of the CAT control interface. Place the mouse cursor over the icon to see information on the tooltip. Click on the icon to open the OmniRig setup dialog.

CW Demodulator Plugin

SSB Demodulator Plugin

RTTY Demodulator Plugin

Purpose

The demodulator plugins are used to convert the I/Q signals received from an SDR radio into the audio signals for listening. These plugins are not used when the input to the program is already demodulated in the receiver.

Settings

- **Pitch**

  The desired pitch of the demodulated audio, in Hertz. Recommended values:

    - CW: 600 Hz
    - RTTY: 2125 Hz
    - SSB: must be 0 Hz.

IC-7610 I/Q Plugin

Purpose

This plugin interfaces the program with the IC-7610 radio via its high speed USB3 port, receives wideband I/Q data and controls the radio settings via the same port. *OmniRig* is not used by this plugin.

Settings

- **Use Sub-Receiver**

  select to receive I/Q data from the main receiver or from the sub-receiver.

- **Blocks per Second**

  Experimental, use the default value.

> ☐ **Tip**
>
> Press the *DUAL-W* button on the radio to enable streaming of the sub-receiver I/Q data.

IC-7610 Transmitter Plugin

Purpose

This plugin allows HamCockpit to use the IC-7610 radio as a transmitter and control it via its high speed USB3 port. *OmniRig* is not used by this plugin.

Any combination of the receiver and transmitter is supported, but for best results select the IC-7610 I/Q Plugin as a receiver when using this plugin for transmitter.

Settings

This plugin does not have any user-configurable settings.

Waterfall Display Plugin

Purpose

This is a visual plugin that shows a waterfall display and a band map, and provides a number of commands to control the RX and TX frequencies and filter settings. The primary purpose of this plugin is to provide the means for visual navigation in the Ham bands.

Settings

- **Scale Background Color, Scale Foreground Color**

  Use these settings to change the colors of the frequency scale. The colors of the band map may be configured in the Band Plan plugin settings.

Controls
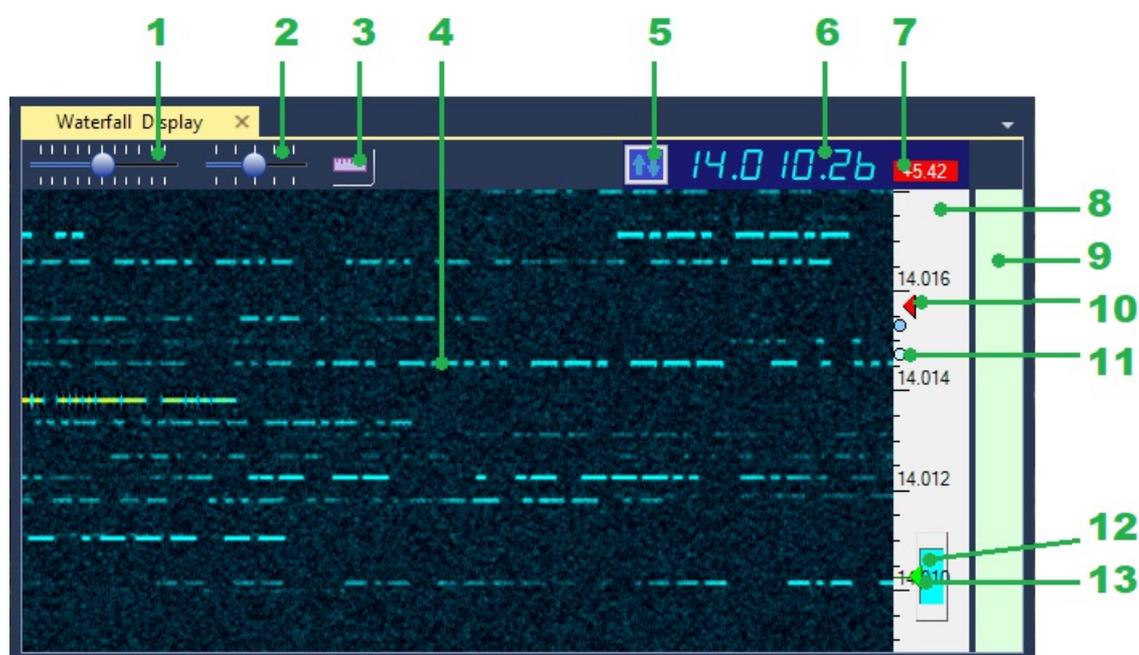
1. **Zoom slider**. Controls the vertical zoom of the waterfall.
2. **Speed slider**. Controls the horizontal speed of the waterfall.
3. **Relative Scale button**. Switches between the absolute frequencies and frequency offsets on the frequency scale **8**.
4. **Signal trace**.
   - Left-click to tune the receiver at the signal.
   - Right-click to enable the Split mode and set the transmit frequency.
   - Ctrl-click to set a frequency mark (**11**).

5. **Tune Button**. Used for paging the waterfall display up and down,
   switching between the band plan segments and changing the bands.
   - Left-click - tune page up
   - Right-click - tune page down
   - Alt-Left-click - next band segment
   - Alt-Right-click - previous band segment
   - Ctrl-Left-click - band up
   - Ctrl-Right-click - band down.

6. **Frequency Display**. Shows the current dial frequency.

- Left-click - jump to the band start (mode dependent).
- Right-click - enter the new dial frequency manually.

7. **Split offset display**. Shows the transmitter offset in the Split mode. Red color indicates that Split is enabled.
   - Left-click to toggle Split On/Off
   - Right-click to reset the Split offset to zero.

8. **Frequency scale**. Shows either the absolute frequencies or frequency offsets, depending on the state of the button **3**.
   - Drag up and down to pan the waterfall vertically without changing the dial frequency.
   - Left-click to set the dial frequency
   - Right-click to enable the Split mode and set the transmit frequency.
   - Ctrl-click to set a frequency mark (**11**).

9. **Band Map**. Shows the band segments using the data provided by the Band Plan plugin. Move the mouse cursor to see band plan information on the mouse tooltip.
10. **Transmit frequency indicator**. Visible only when the Split mode is enabled.
11. **Frequency mark**. Ctrl-click on the waterfall (**4**) or band map (**10**) to place a frequency mark. Up to 4 marks are available. Useful for tracking the listening pattern of a DX station.
12. **Bandpass filter**. Represents the bandpass filter in the Bandpass Filter plugin. The blue rectangle shows the extent of the filter passband. Drag with the mouse to change the filter bandwidth. In the SSB mode the upper and lower cutoff frequencies are adjusted independently, in CW only the filter bandwidth is adjustable.
13. **Dial frequency indicator**. Left-click on the waterfall (**4**) or band map (**10**) to change the frequency.

Example

The video below was recorded with the IC-7610 I/Q plugin as signal source. It shows how to zoom and pan the waterfall, adjust filter passband and navigate the band using the mouse and the dial knob.

AGC Plugin

Purpose

This plugin performs automatic gain control, keeping the output audio within a range of comfortable listening.

Settings

This plugin does not have any configurable settings.

It is important to ensure that the plugin appears at the right place in the DSP Pipeline. Using drag-and-drop in the Plugin Settings dialog, put it after the Bandpass Filter but before Volume Control

 Note

If AGC is placed before *Bandpass Filter* plugin, it reacts to the signals outside of the filter passband, causing so-called AGC pumping.

If it is placed after the *Volume Control* plugin, audio level changes made with the Volume slider are effectively removed by AGC.

Enable / Disable

To enable or disable AGC, click on the AGC button on the toolbar:

Volume Control Plugin

Purpose

Allows the operator to change the level of the output audio sent to the soundcard.

Settings

This control does not have any settings in the Plugin Settings Dialog.

Be sure to put it after the AGC plugin in the DSP Pipeline, or AGC will undo the manual volume changes. Use drag-and-drop in the *Plugin Settings Dialog* to change the order of plugins.

# Frequency Display Plugin

Purpose

This is a stand-alone version of the frequency display built into the Waterfall Display Plugin. See the description of that plugin for more information.

S-Meter Plugin

Purpose

Shows the level of the audio sent to the soundcard for listening. Expressed in the dBFS units (dB Over Full Scale), relative to the maximum amplitude that the soundcard can reproduce.
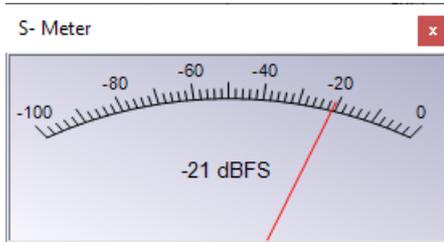
Plugin List Plugin

Shows the list of all plugins and their exported interfaces. This plugin is more useful for the developers than for the end users, but still could be handy for checking if the program finds the plugins that you have installed.

Third Party Plugins

- [Australian band plan](#) by Graham VK3GA
- [Callsign Lookup plugin](#) by Peter W6OP
- [QRZ Plugin](#) by Claus OE6CLD
- [Space Weather Plugin](#) by Peter W6OP
- [SDRPlay Plugin](#) by Dmitry UN7ZO
- [AirSpyHFPlug plugin](#) by Dmitry UN7ZO
- [Mouse Wheel Tuning Plugin](#) by Dmitry UN7ZO

OmniRig

OmniRig is a freeware CAT control engine used in over 50 applications. All programs that support *OmniRig* can access the radio simultaneously.

Some *Ham Cockpit* plugins, such as Conventional Receiver, use this engine. If you install one of those plugins, make sure that you have the latest version of *OmniRig*.

To install or update *OmniRig*, download it from the link below and run the Setup program.

OmniRig is extensible, one can add support of a new radio by creating an INI file with the description of its CAT commands. The INI files are located in this folder:

C:\Program Files (x86)\Afreet\OmniRig\Rigs

To get the latest set of INI files, download the INI Files for OmniRig package and unzip all files into the Rigs folder.

Most of the INI files have been contributed by the users. Open the file in Notepad to see who created it and how to contact the author. If your radio is currently not supported, you can create your own INI for it. The documentation and tools required for INI file creation are available on the OmniRig web page.

> **Tip**
>
> Delete the unused ini files from the Rigs folder to make *OmniRig* start faster.

Resources

 OmniRig web page

 Download OmniRig

 Download INI Files for OmniRig

# How to Add Sidetone to the Audio Processed in the Software

Digital signal processing of RX audio in the PC has many advantages, but it also presents a problem: the output does not include the sidetone. Sending Morse code without hearing the sidetone is virtually impossible, but fortunately, there is an easy solution: processed audio and sidetone can be mixed together in the soundcard.

1. Use a cheap soundcard dongle (a $1 one is OK!), connect the headphones to its output, and the sidetone to the Mic input:


The sidetone may come from the Morse keyer or, if the keyer is built into the radio, from the Phones connector of the radio (but not from the ACC connector). Make sure that only the sidetone is present and receiver audio is muted. How to achieve this is radio-dependent: for example, in TS-570 you can set the IF gain to the minimum, and in IC-7610 you just press the Main AF knob.

2. Configure the software to use the dongle for audio output:
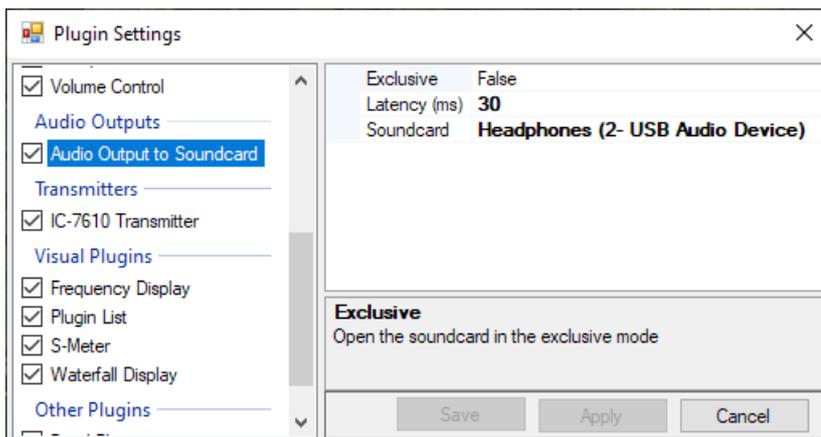
3. Click on the Windows Start button, type "control panel" and click on *Control Panel*, then click on *Soundcard*. Select your soundcard and click on *Properties*. Click on the *Levels* tab in the *Soundcard Properties* dialog, enable both *Speakers* and *Microphone*, and adjust their levels for comfortable listening:
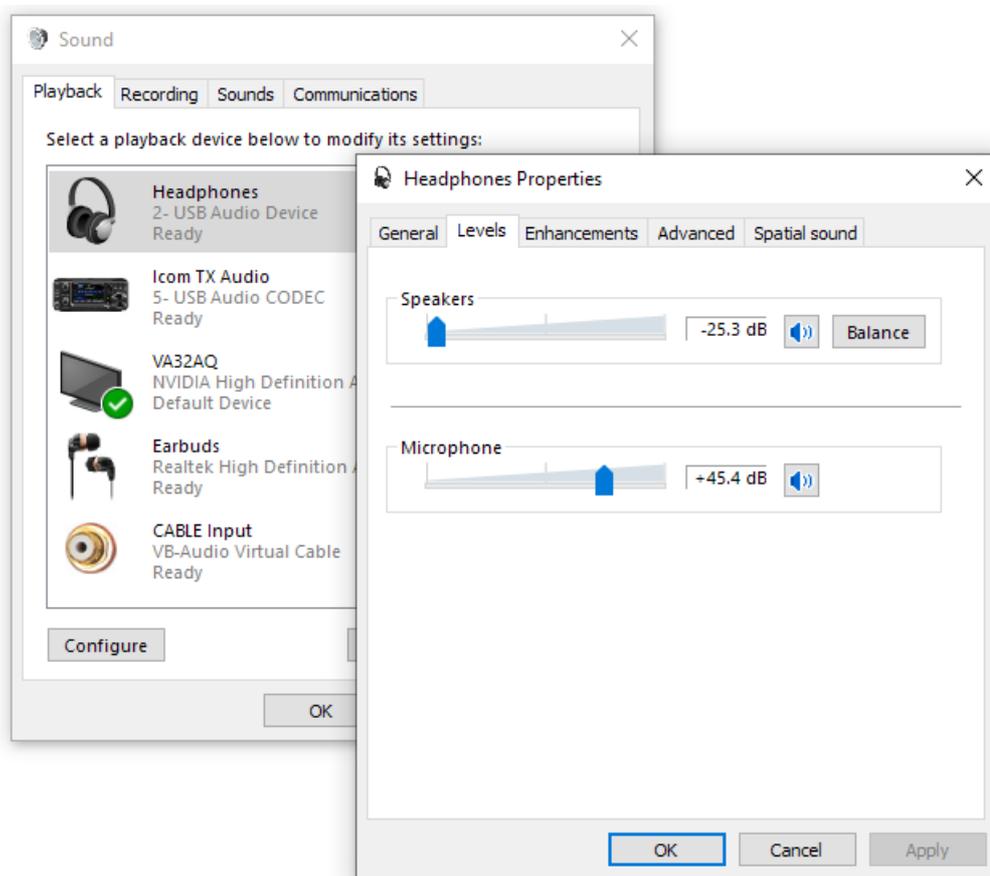
That's it! Now you can hear both audio from the PC and sidetone from your radio or Morse keyer.

Frequently Asked Questions

Q: *How can I ask questions about Ham Cockpit?*

A: You can post your questions to the [Ham Cockpit forum](#), or click on the *Help / Email the Author* command in the program menu and send a direct email to the author.

Q: *I downloaded the plugins and unzipped them to the HamCockpit folder, but they did not appear in the program.*

A: You have to unblock the zip file before you extract the plugins: right-click on the zip, click on Properties in the popup menu, tick the Unblock checkbox, then click on OK.

Q *Where are the data files used by Ham Cockpit located?*

A There are two folders that contain the Ham Cockpit data files:

- *User Data Folder* contains the data specific to the current user, including:
    - *HamCockpit.json*, the file that stores all settings of the program
    - *HamCockpit.log*, the error log file.

- *Reference Data Folder* contains read-only reference data, such as band plans.

Both folders can be opened via the Help menu or by typing these paths in File Explorer:

`%appdata%\Afreet\Products\HamCockpit`

`%programdata%\Afreet\HamCockpit\`

Q *How can I make the waterfall window visible if its command in the View menu is grayed, and the window is not anywhere on the screen?*

A An item in the *View* menu is grayed when the corresponding plugin cannot produce more windows. The *Waterfall Display* plugin can show only one window, and, since its window is already open, its menu command is disabled. Where the window is located is another question. It looks like it is somewhere outside of your screen boundaries; sometimes this happens, especially if you have a multi-monitor setup or you have played with screen resolutions.

To fix the problem, simply delete the settings file, `Settings.json`, where all program settings are stored. The file is located in the `User Data Folder` (see the previous question). Close Ham Cockpit before deleting the file, then start it again and configure the program from scratch.

Version History

V.1.2.0 Conventional Transmitter Plugin

- added the **Conventional Transmitter** plugin;
- the source code of a few more plugins is released.

V.1.1.0 BandScope Plugin

- added the **Band Scope** plugin;
- added error handling for the case when there is no default audio device.

V.1.0.4 Small Fixes

- added exception logging
- improved error handling

V.1.0.3 Small Fixes

- improved error handling

V.1.0.2 Small Fixes

- fixed the User's Guide menu command
- fixed application icon

V.1.0.1 Initial release

The program includes these plugins:

- **Afedri-822x SDR**, an interface to the Afedri SDR radio
- **IC-7610 I/Q**, an interface to the USB3 I/Q stream of the Icom 7610 radio
- **IC-7610 Transmitter**, controls T/R, Split and TX frequency via USB3
- **CW, SSB and RTTY demodulators**, demodulate I/Q signals to audio
- **AGC**, automatic gain control
- **Bandpass Filter**, a variable bandwidth filter for CW, SSB and RTTY
- **Volume Control**, a slider to control the level of the output audio
- **Audio Output to Soundcard**, sends processed audio to an audio device
- **Waterfall Display**, a feature-rich, CW-grade waterfall display
- **Band Plan**, a non-visual plugin that provides band plan data to other plugins

The pack of additional plugins, a separate download, includes:

- **Clock Demo**, adds a UTC/Local time clock to the toolbar
- **Conventional Receiver**, an interface to a receiver that outputs demodulated audio
- **Frequency Display**, displays and controls the operating frequency and TX Split offset
- **S-Meter**, an analog-type meter that shows the audio output level
- **Plugin List**, a panel that shows the list of all installed plugins and interfaces they export

Roadmap

These plugins are in my immediate plans

- *DONE!* **Conventional Transmitter**, a plugin to control the transmitter in a conventional radio
- **IF Adapter**, support of the analog I/Q output from a conventional radio fed to a soundcard
- **Decoded Text Display**, a panel that displays color-highlighted text produced by the future CW/RTTY/Digi decoder plugins

These plugins will come later

- **Adapters** for other SDR radios;
- **Spot Provider** that downloads spots from the Telnet DX clusters, RBN and other sources;
- **Spot Manager** that collects spots from all spot providers, archives them and serves spot data to other plugins;
- **Cluster Monitor**, a visual plugin that shows the spots provided by *Spot manager*;
- **CW Decoder** a single-channel CW decoder based on the decoding algorithms used in CW Skimmer;
- **CW Decoder** based on the Deep Learning algorithm;
- **Logger Interface**, a non-visual plugin that talks to the logger database and allows other plugins to add QSO, check the Needed status of the callsigns, etc.;
- **QSO Entry Box**, a visual plugin to enter QSO in the logger database;
- **CW, RTTY, PSK, SSTV and FT8 skimmers** for mass decoding and spotting of the signals in the corresponding modes;

Ideas considered

There are ideas of many other plugins that I would like to develop, such as a world atlas plugin, a combination of my DX Atlas and HamCap programs with modern graphics and more information displayed, and a plugin for the Vector Sensing antenna for polarization matching, direction finding and many other applications. Currently I do not have time for these projects but may consider them in the future.

Thanks

This project uses quite a few free resources. Many thanks to their authors!

- [Visual Studio 2019](#), Community Edition
- [Git](#), version control system
- [GitHub](#), code sharing service
- [GitHub Pages](#), project publishing service
- [IPP](#), Intel Integrated Performance Primitives
- [CSCore](#), audio library for C#
- [DockPanelSuite](#), docking library for C#
- [SharpGL](#), OpenGL wrapper for C#
- [Json.NET](#), JSON serializer for C#
- [CSIntelPerfLibs](#), IPP wrapper for C#
- [ScintillaNET](#), Scintilla text editor for C#
- [MathNet.Numerics](#), math library for C#
- [FTD3xx_Net](#), FTDI control library for C#
- [ColorSlider](#), slider control for C#
- [Win32Interop](#), Win32 Interop library for C#
- [DocFX](#), documentation generator
- [GhostDoc](#),code documenting tool.
- Icon 🎩 by [Freepik](#) from [www.flaticon.com](#)