GhostBroker

CI-V Protocol Broker for ICOM Transceivers

User Guide

by YO3HEX

1. What is GhostBroker

GhostBroker is a CI-V protocol broker for ICOM transceivers. It sits between your radio and multiple software applications, letting them all share one physical serial connection without conflicts.

Without a broker, only one application can talk to the radio at a time. If you want OmniRig, WFVIEW, a logging program etc. running simultaneously on the same ICOM radio, you need something in the middle to manage the traffic. That is what GhostBroker does.

What it actually does

- Receives CI-V commands from each client application on its own serial port.
- Queues them and sends them to the radio one at a time, in fair order.
- Routes each response back to the client that asked for it.
- Broadcasts unsolicited radio updates (frequency changes, PTT etc.) to all clients that want them (spectrum broadcast is at the moment an experimental feature)
- Caches repeated read commands so the radio is not hammered with identical queries (cache functionality can be enabled or disabled based on your requirements)

What it does not do

- It does not modify radio behavior. It does not change frequencies, switch modes, or transmit on your behalf.
- It does not translate between CI-V versions or rewrite commands. It passes frames as-is.
- It is not a remote radio solution. All clients must run on the same machine where GhostBroker is installed or using a software package like VSPE that can expose a virtual serial port as a TCP client/server

2. How It Works

Architecture

GhostBroker uses one serial port for the radio and one serial port per client application. Every port runs at the same baud rate.

The broker has a single transaction loop. Only one command is in flight to the radio at any time. When the radio responds, the broker routes the response to the correct client and immediately picks up the next command from the queue. The protocol is strictly request/response on a shared bus.

Client Side	Broker	Radio Side
OmniRig → COM11		
WFVIEW → COM12	**GhostBroker** (queue → radio → route)	**ICOM Radio** COM17
DXLog → COM13		

Scheduling — Weighted Round Robin

Each client has a priority weight (1–5). The broker uses weighted round-robin scheduling: a client with weight 5 gets to send 5 commands before the next client gets its turn. If only one client has commands queued, it gets the radio immediately regardless of weight.

This means priority affects fairness under load, not absolute speed. A weight-1 client will never be starved — it simply gets fewer turns per cycle than a weight-5 client.

CI-V Echo Handling

Some ICOM radios echo back the exact command frame before sending the actual response. This is a radio setting ("CI-V Echo" on USB). GhostBroker handles this transparently.

When an echo is detected (the received frame is byte-for-byte identical to the command that was sent), the broker forwards it to the requesting client so that software like OmniRig which expects the echo receives it, but the transaction stays open. The broker continues waiting for the real response from the radio.

NOTE: You do not need to disable CI-V Echo on your radio. GhostBroker handles both cases. If you do disable it, that is fine too. (For example for ICOM 7600 there is no way to disable USB CIV echo) – also a large number of software packages do expect the echo reply to be part of the transaction.

Transceive Broadcasts

ICOM radios in transceive mode send unsolicited updates whenever the radio state changes — frequency, mode, split, PTT status. These are not responses to any command. GhostBroker detects them and broadcasts them to all connected clients.

Spectrum scope data (command 0x27) is also a transceive broadcast. Because spectrum frames arrive at high volume, spectrum forwarding is configured per client. Only clients with spectrum enabled receive that data. (this is an experimental feature that is going to be enabled in the next release of the GhostBorker)

Response Cache

Many applications poll the radio repeatedly for the same information — current frequency, current mode. The cache intercepts these repeated queries and returns the last known answer without touching the radio, as long as the cached entry has not expired.

The cache is address-agnostic. A response obtained for one client can be served to another client (with the destination address rewritten). Cache entries are invalidated automatically when a SET command changes the radio state, or when a transceive broadcast reports a change.

In typical usage the cache reduces actual radio traffic by 60–70%. This is the single most impactful performance feature in GhostBroker. Cache feature can be enabled or disabled from the configuration screen.

Queue Overflow

Each client has an internal queue with a maximum depth of 1000 frames. If the queue is full:

- For spectrum data: the oldest spectrum frame in the queue is dropped to make room. The stream continues without gaps in live data.
- For commands: the new command is dropped. Existing commands are preserved in order. This prevents a flood of spectrum data from pushing out important commands.

3. Requirements

Requirement	Details
Operating System	Windows 10 or Windows 11
Radio	ICOM transceiver with CI-V support via USB or RS-232
Serial Ports	One port for the radio + one port per client application. Use USB-to-serial adapters (e.g. CP2102, CH340) or virtual COM port software (e.g. VSPE, com0com) as needed.
Client Software	Any application that communicates via CI-V on a serial port. Tested with DXLog, N1MM, OmniRig, WFVIEW, HAMLib, Log4OM.
Baud Rate	All ports (radio and clients) must use the same baud rate. Match this to your radio's CI-V baud rate setting.

Client Limits by Baud Rate

Lower baud rates limit how many commands per second can pass through. GhostBroker enforces a maximum number of clients based on the baud rate you select:

Baud Rate	Max Clients
≤ 19200	3
≤ 38400	4
≤ 57600	5
115200	6

4. Configuration

All configuration is done in the GhostBroker GUI before starting the broker. Configuration is saved to a JSON file in your Windows AppData folder. The exact path is shown at the bottom of the log window when the application starts.

```
%APPDATA%\CivBroker\config.json
```

You can edit this file manually if needed, but the GUI covers all available settings.

Configuration Tab — Radio

Set the serial port and baud rate for your radio here. The port drop-down is populated automatically from ports detected on your system. The baud rate must match the CI-V baud rate setting in your radio's menu.

Field	What it sets	Default
Radio Port	Serial port connected to the radio	COM17
Baud Rate	Serial baud rate for all ports	19200

Configuration Tab — Client Endpoints

Each row in the client table is one software application. You must configure at least one client to start the broker.

Column	Function	Values	Notes
ENBL	Enable or disable this client	On / Off	Disabled clients are ignored completely.
CLIENT NAME	Identifier shown in telemetry and logs	Any unique text	Must be unique across all clients.
PORT	Serial port for this client	COM port	Cannot be the same as the radio port or another client.
PRIO	Scheduling weight	1–5	Higher = more turns per cycle. Default: 1.
SPECTRUM	Forward spectrum scope data	On / Off	Enable only for clients that use spectrum data. Reduces load on other clients.
NOTE: Each client port must have a COM-port-capable application on the other end, or a virtual COM port pair. The application must be open and listening before or shortly after starting GhostBroker.			

5. Advanced Settings

The Advanced Settings tab contains performance tuning, cache configuration, and debug options. These are saved along with the rest of the configuration and take effect the next time the broker is started.

Performance Settings

Setting	Default	Description
Response Timeout	200 ms	How long the broker waits for the radio to respond before giving up on a command. Range: 50–5000 ms. Increase if your radio is slow to respond (some ICOM radios need 300 ms or more).
Max Burst Per Endpoint	2	How many consecutive commands one client can send before the scheduler moves to the next client. Range: 1–20. Increase for clients that batch commands.

Cache Settings

The cache is enabled by default. Disabling it is not recommended unless you are troubleshooting a specific issue.

Setting	Default	Description
Enable Response Cache	On	Enables caching of read command responses. Repeated queries for the same data are answered from cache without querying the radio.
Cache TTL	500 ms	How long a cached response remains valid. After this time, the next query goes to the radio. Range: 10–10000 ms. Shorter = fresher data, more radio traffic.
Max Cache Entries	1000	Maximum number of cached responses. When exceeded, expired entries are removed first, then the least recently used entries. Range: 10–10000.

Debug Settings

Setting	Default	Description
Enable Debug Logging	Off	Outputs frame-level detail to the Visual Studio Debug Output window. This has a measurable performance impact. It is intended for development and troubleshooting only, not for normal operation – do not activate it as it would have no output and no use for normal user.

6. Telemetry

The Status tab is showing data while the broker is running. It updates every second and shows live broker health without any manual action.

Broker Metrics (top row)

Metric	What it shows
Throughput	Commands per second over a sliding 5-second window. The peak value tracks the highest rate seen since the broker started.
Latency	Average round-trip time from sending a command to receiving the radio response, in milliseconds. The peak value is the highest single latency recorded.
Cache	Cache hit rate as a percentage. The number in parentheses is the total count of commands served from cache (radio queries saved).
Total	Total commands processed since the broker started.
Uptime	Time elapsed since the broker was started.
Errors	Total error count (timeouts + write failures) and the error rate as a percentage of total commands.

7. Operating the Broker

Starting

1. Open GhostBroker.
2. On the Configuration tab, verify the radio port and baud rate match your radio.
3. Verify each client entry has the correct port, and that the client application is open on that port.
4. **SAVE CONFIGURATION AFTER EVERY MODIFICATION AND PRIOR STARTING THE BROKER using the SAVE button**
5. Click **Start Broker**.

If the configuration is valid, the broker opens all ports and begins processing. The Telemetry tab activates and the log window shows the startup summary. If there is a configuration error, a dialog describes the problem.

WARNING: Once the broker is running, the Configuration and Advanced Settings tabs are locked. You must stop the broker before making changes.

Stopping

Click **Stop**. The broker stops accepting new commands, clears all queues, and closes all serial ports. The configuration tabs unlock. Stopping and restarting is required any time you change a setting.

Saving Configuration

Click **Save** at any time while the broker is stopped. The current state of all fields is written to the config file. Configuration is not saved automatically — if you close the application without saving, unsaved changes are lost.

8. Troubleshooting

The broker starts but clients receive no data

- Verify that the client application is open and listening on the port assigned to it in GhostBroker.
- Verify the baud rate in GhostBroker matches the baud rate configured in the client application.
- Check that the client port is not already in use by another application.

The radio does not respond (timeouts in telemetry)

- Verify the radio port is correct and the radio is powered on.
- Check the baud rate. It must match the CI-V baud rate in the radio's menu.
- Try increasing the Response Timeout in Advanced Settings (some radios need 300–500 ms).
- If using a USB connection, try a different USB cable or port. USB-to-serial adapters can be unreliable.

OmniRig does not work correctly

- OmniRig expects to receive a CI-V echo before the actual response. GhostBroker handles this automatically. If OmniRig still misbehaves, verify that OmniRig is configured to use the correct COM port and baud rate.
- If the radio has CI-V Echo disabled, GhostBroker will still work correctly — it simply will not forward an echo because none is received.

High latency or slow response

- Check the Telemetry tab. If cache hit rate is low and throughput is high, the radio is being queried heavily. Consider increasing the Cache TTL.
- If many clients are competing, adjust priority weights so that the most time-sensitive application has a higher weight.
- Increase the baud rate if your radio supports it. Higher baud rate = more headroom for multiple clients.

Queue depth is growing (orange or red in telemetry)

- A growing queue means commands are arriving faster than the radio can process them. This is normal during bursts but should not persist.
- If it persists, reduce the number of active clients or increase the baud rate.
- If the queue is growing on a spectrum-enabled client, that client is receiving high-volume spectrum data. Disable spectrum on clients that do not need it. – **spectrum functionality not enabled in this version**

The broker will not start — configuration error

- The error dialog tells you exactly what is wrong. Common causes: duplicate port assignments, a client port matching the radio port, or too many clients for the selected baud rate.

9. Configuration Reference

All settable values, their valid ranges, and defaults in one place.

Parameter	Default	Min	Max
Radio Port	COM17	—	—
Baud Rate	19200	4800	115200
Client Weight (Priority)	1	1	5
Response Timeout (ms)	200	50	5000
Max Burst Per Endpoint	2	1	20
Cache TTL (ms)	500	10	10000
Max Cache Entries	1000	10	10000

Legend: Radio Client Performance Cache