PgTgBridge – Installation Guide

PgTgBridge is software that translates the TCP network protocols used by FlexRadio® FLEX6000/FLEX8000 series radios into the CAT protocol used by Elecraft® amplifiers and tuners.

Key Features:

- Creates seemless integration between FlexRadio® and Elecraft® hardware.

- Automatic radio discovery. Connects to first radio heard on network or can be locked to a specific radio serial number on multi-radio networks.

- Plugin system for amplifiers and tuners. Plugins for Elecraft® KPA1500, KPA500 and KAT500 are built-in. Provides API for other amplifiers and tuners to be supported.

- Full remoting of Elecraft KPA1500 via TCP over network including PTT! No RCA or USB cables required for full integration. Put the amp in another room! KPA500 and KAT500 can be configured for TCP as well.

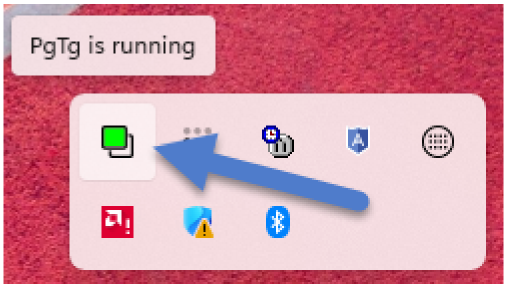
- Provides configurable power control settings per band when switching amplifier mode from Operate to Standby. (Reduces radio power in Operate, full power in Standby)

- Sends transmit slice frequency to amplifier and tuner for automatic band tracking and tuning solution selection.

- RF Output Power, SWR and Amplifier Temperature metering is displayed in the FlexRadio® client software and on Maestro. Operate/Standby/Tuner controls appear as well. Metering is also available in a floating window on the PC.

Requirements:

PgTgBridge runs on 64-bit installations of Windows that support the .NET 10 Desktop runtime. Windows 10 and 11 are currently supported. PgTgBridge is comprised of two parts: a Windows service and a Task Tray application. The service contains the primary logic of PgTg. It runs in the background so the user does not need to be logged in for normal operation. The second part is PgTgController that runs when the user logs in and stays hidden in the task tray. It is used to configure and monitor the service. The icon looks like this:



PgTg is running

PgTgBridge (PgTg) communicates with the radio and amplifier over Ethernet. The KPA1500 amplifier natively has an Ethernet port making the wiring requirements quite simple. PgTg handles sending TX slice frequency data to the amplifier so no USB connection is required. With KPA1500 firmware version v3.07 (releasing soon), PTT signaling is also handled over Ethernet. This means you can locate the KPA1500 away from the radio, even in another room or building. Transmit turnaround is under 2 milliseconds.

PgTg also supports the Elecraft® KPA500 amplifier and KAT500 tuner. You will need to use the standard serial data cables with these units connected to the PC running PgTg. PgTg supports connecting to the KPA500 / KAT500 via serial (USB) or TCP. Therefore you could add TCP to Serial redirection and convert access to TCP. This is easily done with a Raspberry Pi. Another reason this approach is appealing is the Raspberry Pi can also signal PTT to the KPA500 using a relay hat. That would then allow placing the KPA500 and KAT500 in a different location than the operating position,

PgTg provides TX slice frequency data to both the KPA500 and KAT500. With a future firmware update, PTT over Ethernet could be achieved with the KPA500. For now, the Raspberry Pi solution is the best go-to plan. Refer to the last section of this document if you want to use TCP to connect to the KPA500 or KAT500.

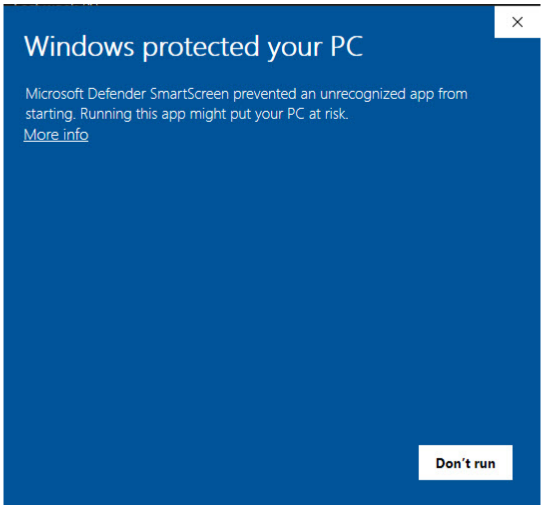
Before beginning the installation of PgTg, you should use the Import/Export Profiles feature in SmartSDR to create a backup of your current radio settings, memories and profiles.

Install Steps:

Obtain the installer program from kd4z.com

The installer program is deployed in Windows MSI form. Just double click the MSI file to launch the installation process. Currently the installer is not digitally signed so Windows may pop up a warning about preventing an unrecognized app from starting. This is expected.

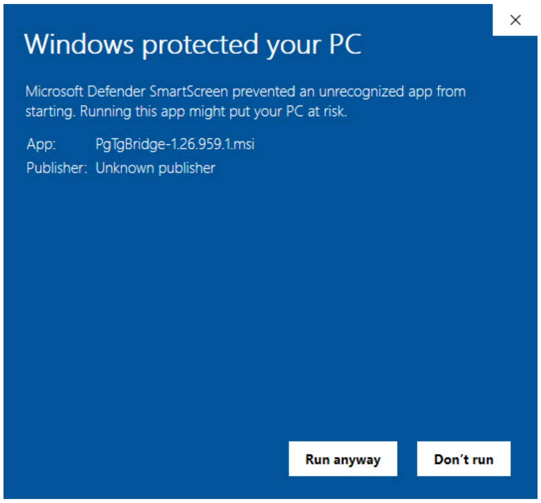
Click on the words "**more info**" and then "**Run anyway**" to proceed.



Windows protected your PC

Microsoft Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

More info



Windows protected your PC

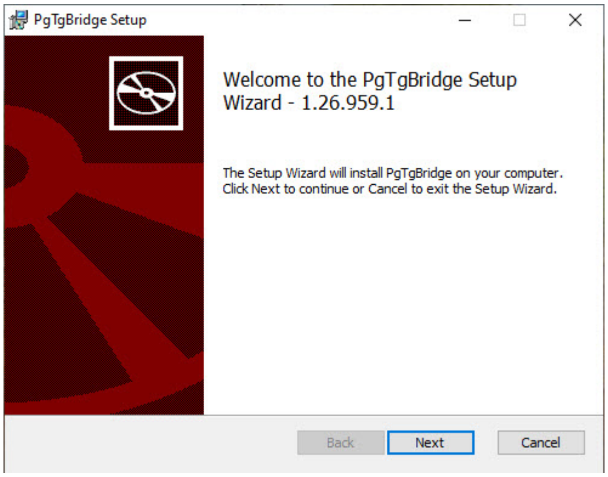
Microsoft Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

App: PgTgBridge-1.26.959.1.msi
Publisher: Unknown publisher

You will be presented with a standard installation wizard to gather the information needed to install PgTg. If the Microsoft .NET 10 isn't detected, you will be given the opportunity to install it directly from the PgTgBridge installer. No download is needed. You must choose "**Install Runtime**" if you see the Runtime Required page. The Microsoft Windows Desktop Runtime installer will launch. You might be prompted from User Account Control (UAC) to allow it to proceed. You must accept it!

When the runtime installer has finished, you will be brought back to the PgTg installer window. Click "Finish". This will actually end the installer application. But if you had to install the .Net Runtime, PgTg is NOT INSTALLED YET!

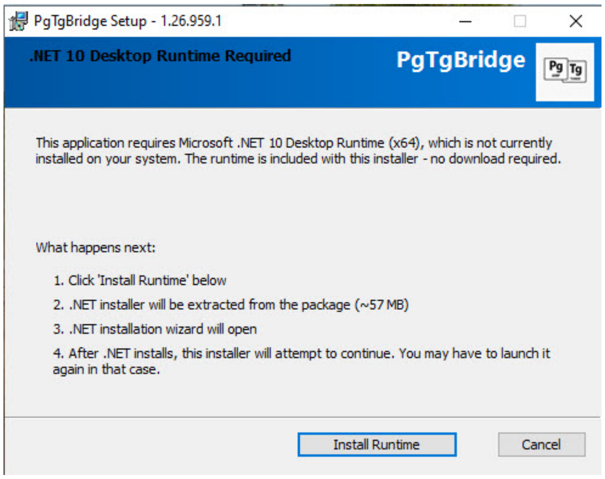
Launch the PgTgBridge MSI installer AGAIN. This time the installer will detect the .NET Runtime is installed and proceed with the actual installation of PgTgBridge.



PgTgBridge Setup

Welcome to the PgTgBridge Setup Wizard - 1.26.959.1

The Setup Wizard will install PgTgBridge on your computer. Click Next to continue or Cancel to exit the Setup Wizard.



PgTgBridge Setup - 1.26.959.1

.NET 10 Desktop Runtime Required

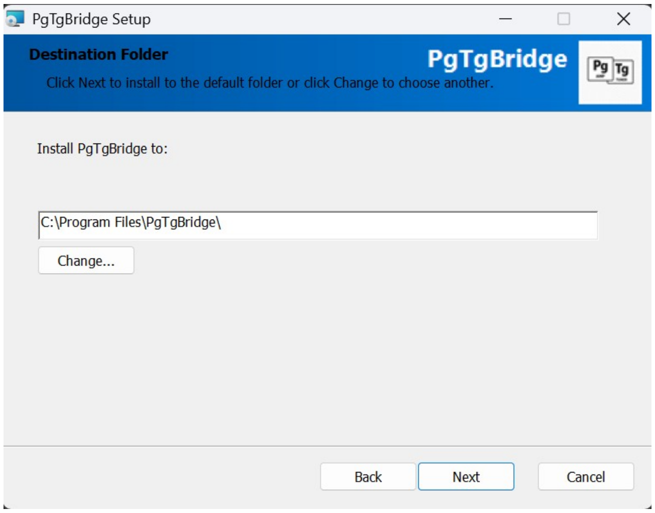
This application requires Microsoft .NET 10 Desktop Runtime (x64), which is not currently installed on your system. The runtime is included with this installer - no download required.

What happens next:

1. Click 'Install Runtime' below
2. .NET installer will be extracted from the package (~57 MB)
3. .NET installation wizard will open
4. After .NET installs, this installer will attempt to continue. You may have to launch it again in that case.

The next wizard pages will ask where and how you want PgTgBridge installed. Unless you want to install it to a different drive, I suggest you accept the default location. Click **Next.**

The Service Account page allows configuring what user the Windows Service runs as. Local System is the default which does not require a password to be associated with the service. PgTg does not require this level to operate, and can be configured to run as a normal non-privileged user. Be advised that if the password is changed for that user, the service will no longer run. You will have to configure the service from the "Services" applet in Control Panel in that event. Click **Next** to proceed.



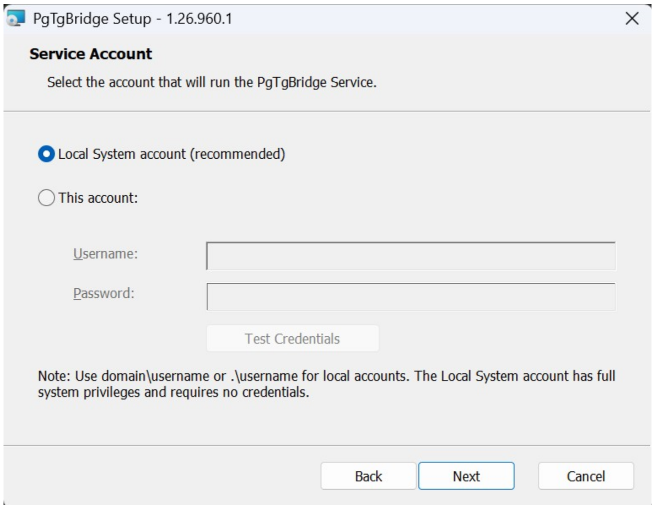
PgTgBridge Setup

Destination Folder

Click Next to install to the default folder or click Change to choose another.

Install PgTgBridge to:

C:\Program Files\PgTgBridge\



PgTgBridge Setup - 1.26.960.1

Service Account

Select the account that will run the PgTgBridge Service.

● Local System account (recommended)

○ This account:

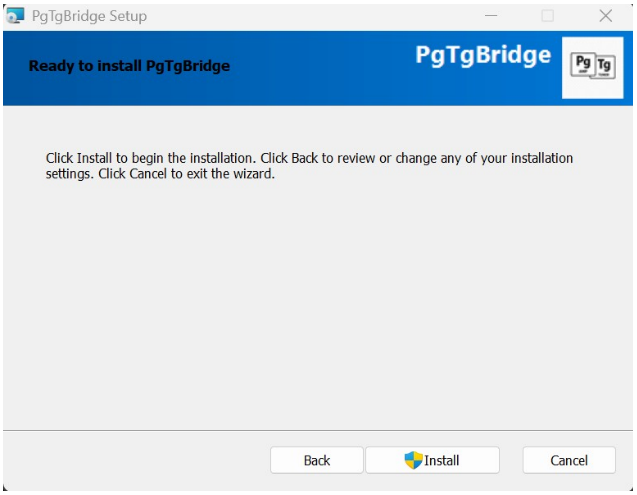
Username:

Password:

Note: Use domain\username or .\username for local accounts. The Local System account has full system privileges and requires no credentials.

At this point the installer can proceed with deploying the PgTgBridge service and PgTgController. This involves setting up firewall rules, creating the service registration and shortcu to launch PgTgController.

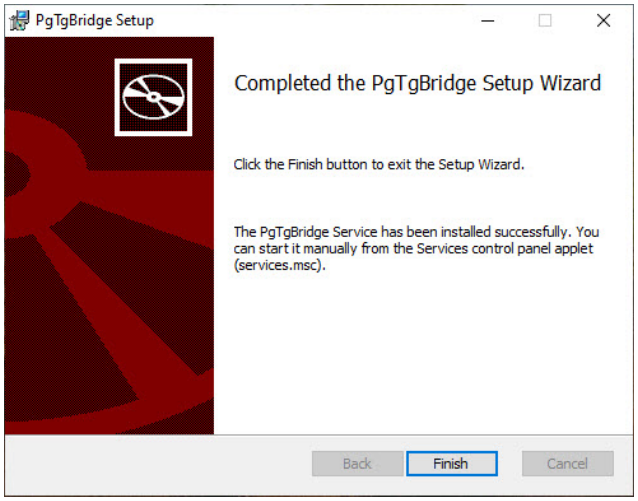
Click Install. You may see the PgTgController pop up a message to say "Hi, I'm running here in your Task Tray" If the PgTgService hasn't finished starting, you may see PgTgController complain about that...you can ignore that this time! Click **Finish** to dismiss the installer wizard.



PgTgBridge Setup

Ready to install PgTgBridge

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.



PgTgBridge Setup

Completed the PgTgBridge Setup Wizard

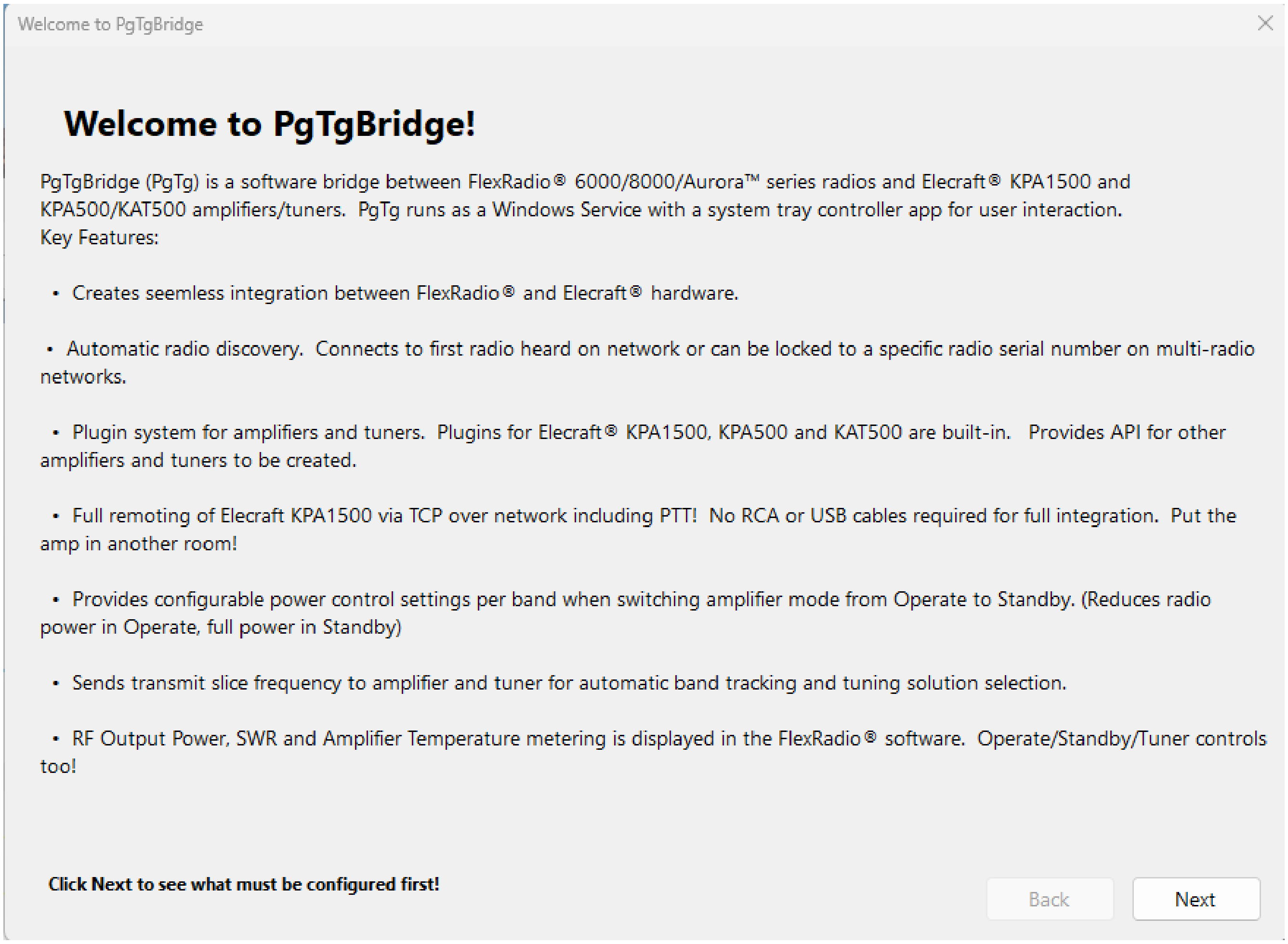
Click the Finish button to exit the Setup Wizard.

The PgTgBridge Service has been installed successfully. You can start it manually from the Services control panel applet (services.msc).

The icon for PgTgController should show up in the Task Tray area with a green square. Look for it in the lower right corner of your desktop. The green color indicates it is communicating with the PgTgBridge Service and it has started successfully. It also may be yellow (PgTg has been paused or red, indicating that the PgTgBridge Windows service isn't running .

Welcome!

If this is the first time the PgTg installer is executed on your PC (or if you removed the SettingsConfig.json file), you will be presented with a short introduction wizard. It will provide a quick summary of the configuration settings that will be needed for PgTg to find your hardware and operate correctly.



Welcome to PgTgBridge!

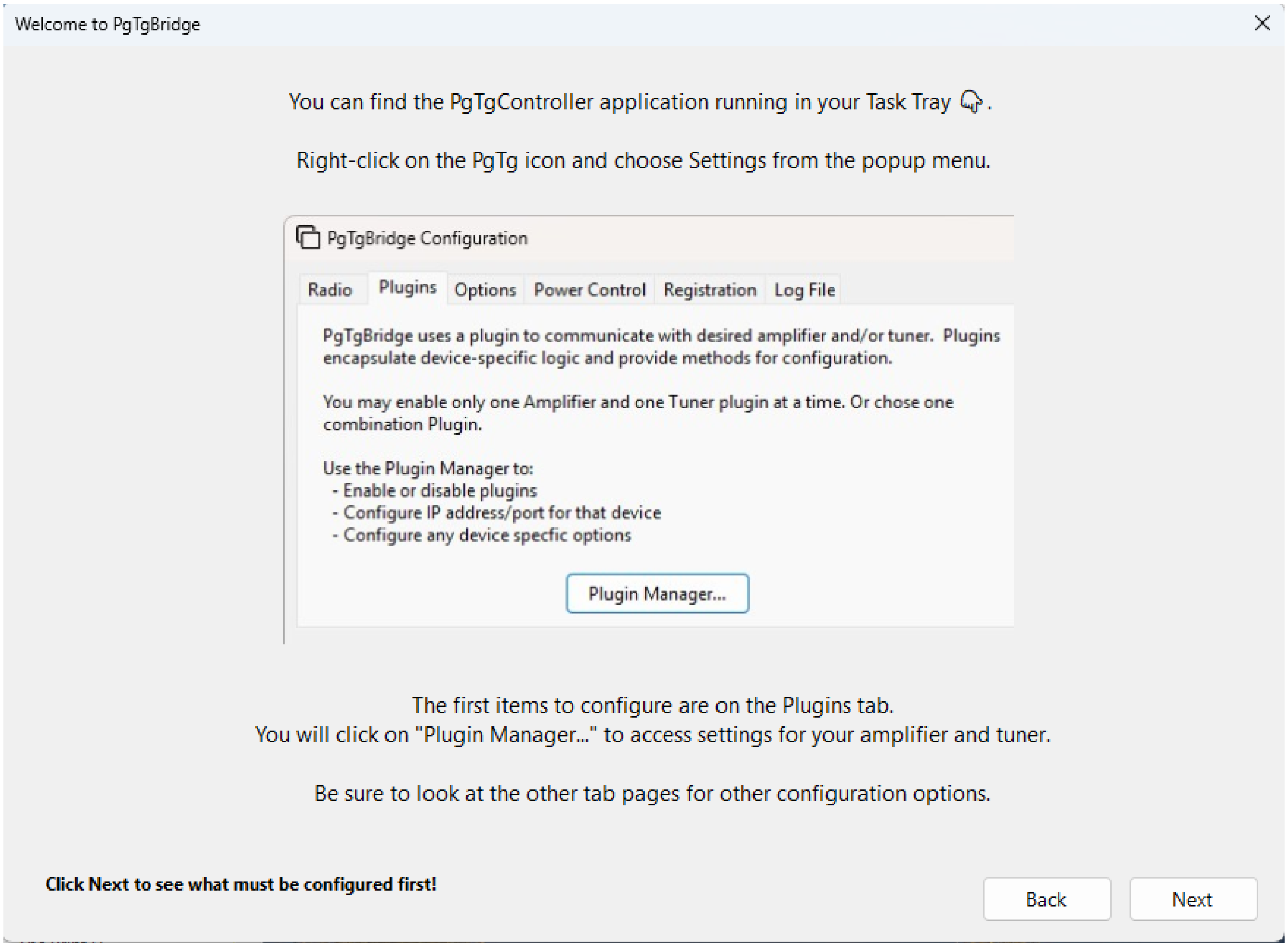
PgTgBridge (PgTg) is a software bridge between FlexRadio® 6000/8000/Aurora™ series radios and Elecraft® KPA1500 and KPA500/KAT500 amplifiers/tuners. PgTg runs as a Windows Service with a system tray controller app for user interaction.
Key Features:

- Creates seemless integration between FlexRadio® and Elecraft® hardware.
- Automatic radio discovery. Connects to first radio heard on network or can be locked to a specific radio serial number on multi-radio networks.
- Plugin system for amplifiers and tuners. Plugins for Elecraft® KPA1500, KPA500 and KAT500 are built-in.   Provides API for other amplifiers and tuners to be created.
- Full remoting of Elecraft KPA1500 via TCP over network including PTT! No RCA or USB cables required for full integration. Put the amp in another room!
- Provides configurable power control settings per band when switching amplifier mode from Operate to Standby. (Reduces radio power in Operate, full power in Standby)
- Sends transmit slice frequency to amplifier and tuner for automatic band tracking and tuning solution selection.
- RF Output Power, SWR and Amplifier Temperature metering is displayed in the FlexRadio® software. Operate/Standby/Tuner controls too!

Click Next to see what must be configured first!

Click **Next** to proceed to the next page.

The next page gives you a peek at the PgTgBridge Configuration window. Note there are many tabs that organize the settings. The Plugins tab is displayed here. Note, you will need to click on the Plugin Manager button when selecting the amplifier/tuner combinations.



Welcome to PgTgBridge

You can find the PgTgController application running in your Task Tray.

Right-click on the PgTg icon and choose Settings from the popup menu.

PgTgBridge Configuration

PgTgBridge uses a plugin to communicate with desired amplifier and/or tuner. Plugins encapsulate device-specific logic and provide methods for configuration.

You may enable only one Amplifier and one Tuner plugin at a time. Or chose one combination Plugin.

Use the Plugin Manager to:
- Enable or disable plugins
- Configure IP address/port for that device
- Configure any device specfic options

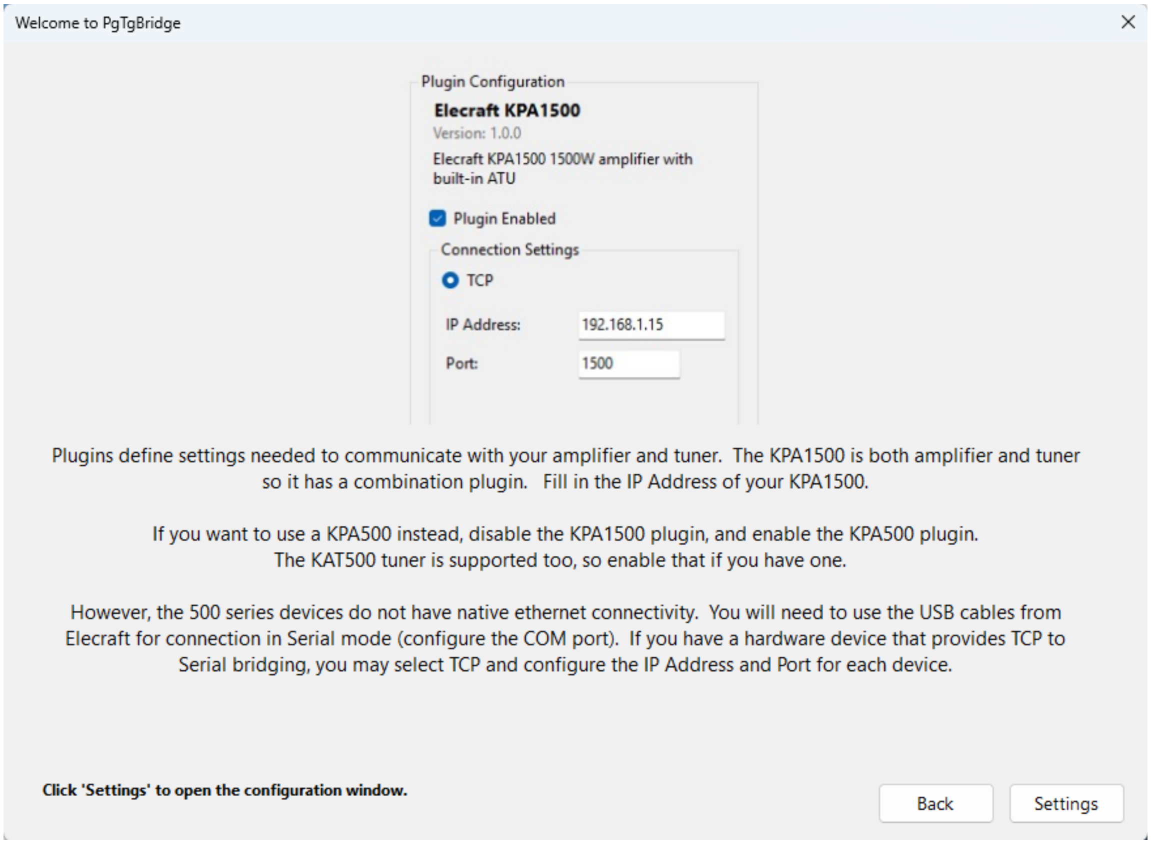
The first items to configure are on the Plugins tab.
You will click on "Plugin Manager..." to access settings for your amplifier and tuner.

Be sure to look at the other tab pages for other configuration options.

Click Next to see what must be configured first!

Click **Next** to continue.

The last introduction page shows the detail area for the KPA1500 Plugin. You will need to fill in the IP address of your amplifier. Note: you must configure your router/firewall DHCP settings to reserve that IP address for your amplifier. You should not need to change the port from the default of 1500. That settings is available if you have to forward through a router on a different port to get to the amplifier.



Welcome to PgTgBridge

Plugin Configuration

Elecraft KPA1500

Version: 1.0.0

Elecraft KPA1500 1500W amplifier with built-in ATU

☑ Plugin Enabled

Connection Settings

● TCP

IP Address: 192.168.1.15

Port: 1500

Plugins define settings needed to communicate with your amplifier and tuner. The KPA1500 is both amplifier and tuner so it has a combination plugin. Fill in the IP Address of your KPA1500.

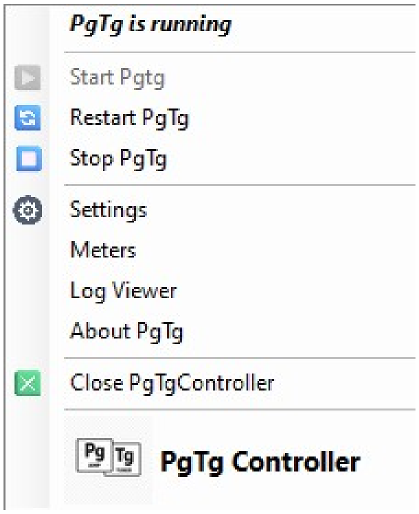
If you want to use a KPA500 instead, disable the KPA1500 plugin, and enable the KPA500 plugin. The KAT500 tuner is supported too, so enable that if you have one.

However, the 500 series devices do not have native ethernet connectivity. You will need to use the USB cables from Elecraft for connection in Serial mode (configure the COM port). If you have a hardware device that provides TCP to Serial bridging, you may select TCP and configure the IP Address and Port for each device.

Click 'Settings' to open the configuration window.

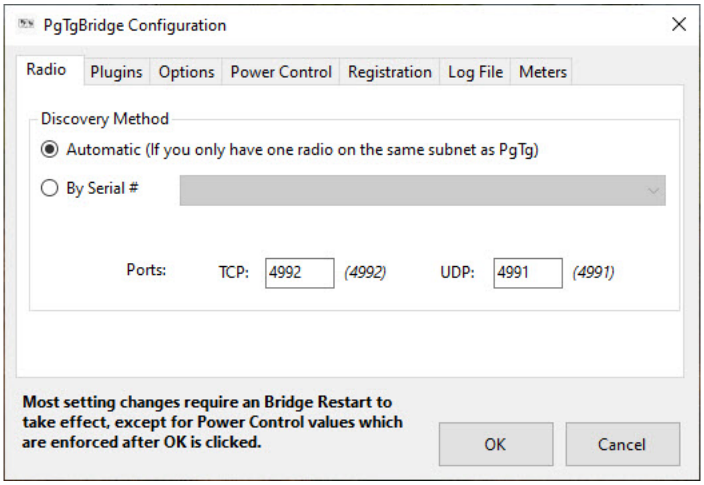
Click **Settings** and you will be navigated to the actual settings window to configure everything. Note that most of the settings will require you to use PgTgController to “Stop then Start” or “Restart” PgTg.

When you have completed making the setting assignments, you can click on the PgTgController icon in the Task Tray to pop up the menu. This is what it looks like:



PgTg Settings Window:

Radio Tab: Choose how you want PgTg to select your radio. If you only have one, leave it on Automatic.
If you have more than one Flexradio running, PgTg will select the first one heard on the network. In this case, select “By Serial #” and select the desired radio from the drop-down list. Again, leave the Port values at the default unless you have special requirements.



Discovery Method

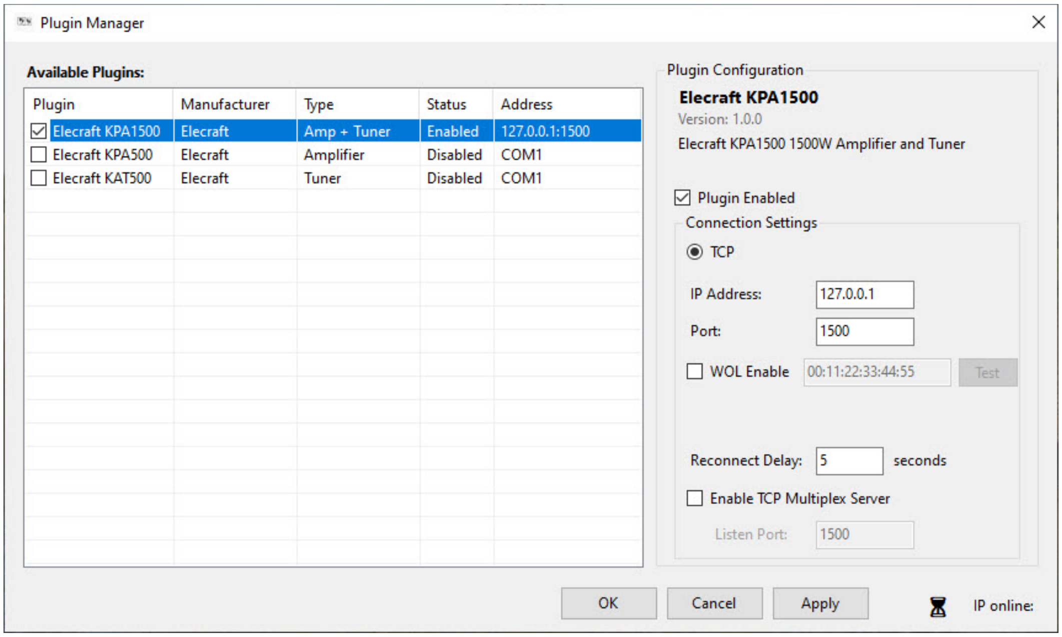
● Automatic (If you only have one radio on the same subnet as PgTg)

○ By Serial #

Ports: TCP: 4992 *(4992)* UDP: 4991 *(4991)*

Most setting changes require an Bridge Restart to take effect, except for Power Control values which are enforced after OK is clicked.

Plugin Tab: Click the Plugin Manager button. You will need to select your desired plugin. The KPA1500 has a built-in tuner, and the plugin knows how to handle both roles. However, if you want to use the KPA500 with or without the KAT500, disable the KPA1500 plugin, and enable what you want to use. Note: You can only enable one amplifier and one tuner plugin at a time. There is an API exposed for third-party software developers to write plugins for PgTg.



Available Plugins:

Plugin	Manufacturer	Type	Status	Address
☑ Elecraft KPA1500	Elecraft	Amp + Tuner	Enabled	127.0.0.1:1500
□ Elecraft KPA500	Elecraft	Amplifier	Disabled	COM1
□ Elecraft KAT500	Elecraft	Tuner	Disabled	COM1

Plugin Configuration

Elecraft KPA1500
Version: 1.0.0
Elecraft KPA1500 1500W Amplifier and Tuner

☑ Plugin Enabled

Connection Settings

● TCP

IP Address: 127.0.0.1

Port: 1500

□ WOL Enable 00:11:22:33:44:55

Reconnect Delay: 5 seconds

□ Enable TCP Multiplex Server

Listen Port: 1500

IP online:

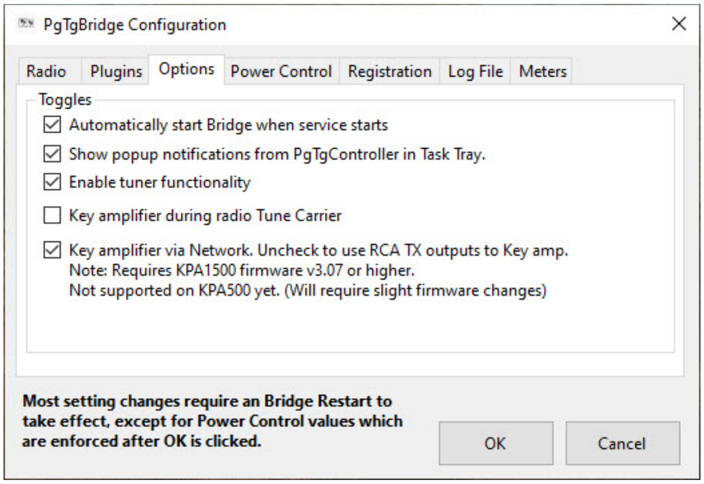
The KPA1500 detail area has configuration items for IP address, port and Wake-on-LAN (WOL) support. You will need to fill in your amplifier’s IP address. You should also obtain the MAC address of your KPA1500 (from the front panel menu),
and fill it in to enable WOL support. PgTg will power on your amplifer for you when it starts. You can click the **Test** button to validate that you have the MAC address entered correctly. Your amp should power on indicating success.

PgTg Settings Window (continued):

Options tab: The options tab has checkbox toggles for specific needs. You likely want PgTg to launch when your PC is powered on. If you get tired of the toast messages from PgTgcontroller, you can silence them. If you don't use the tuner feature (mainly in the KPA1500, but it will affect all tuner selections), you can disable the tuner functionality of PgTg. This will remove the TUNER tab in SSDR and hide the TUN button on the Maestro.

One cool, first to market feature that PgTg brings to the table is Keying the amp via the network. As the note mentions, this is a new feature of the latest KPA1500 firmware. (v3.07) If you don't have that version in your KPA1500, you MUST uncheck the "Key amplifier via Network" checkbox! You will get "Interlock" errors from the Flexradio otherwise.

If your KPA1500 does not have the correct firmware version, uncheck the box and make the usual connection to the KEY IN input of the amplifer from the radio's TX1, TX2 or TX3 outputs with an RCA plug type cable. This settting is primarilly use with the KPA500 or other amplifiers that do not support PTT over Ethernet.


PgTgBridge Configuration

Toggles

☑ Automatically start Bridge when service starts

☑ Show popup notifications from PgTgController in Task Tray.

☑ Enable tuner functionality

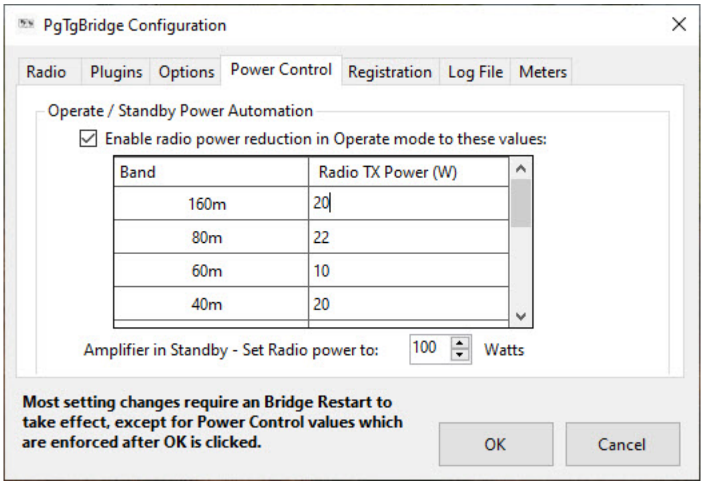
☐ Key amplifier during radio Tune Carrier

☑ Key amplifier via Network. Uncheck to use RCA TX outputs to Key amp.
Note: Requires KPA1500 firmware v3.07 or higher.
Not supported on KPA500 yet. (Will require slight firmware changes)

Most setting changes require an Bridge Restart to take effect, except for Power Control values which are enforced after OK is clicked.

Power Control tab: One feature that Elecraft users enjoy is the exciter power level integration of the K3/K4 radio with the KPA amplifers. (Operate to standby settings). There is no such feature in the Flexradio world. (without 3rd party software)

PgTg adds the ability to configure what power level you want the radio to be forced to when the amplifier is toggled between operate and standby. The operate settings are granular per band. The standby setting is a singular value which is not band specific.


PgTgBridge Configuration

Operate / Standby Power Automation

☑ Enable radio power reduction in Operate mode to these values:

Band	Radio TX Power (W)
160m	20
80m	22
60m	10
40m	20

Amplifier in Standby - Set Radio power to: 100 Watts

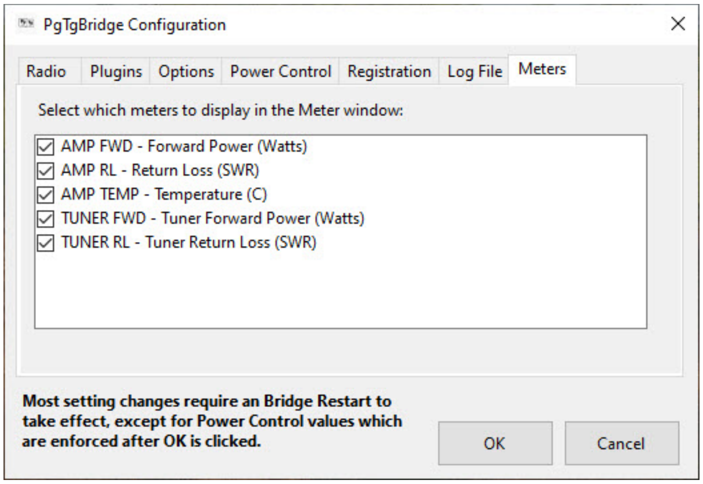
Most setting changes require an Bridge Restart to take effect, except for Power Control values which are enforced after OK is clicked.

PgTg Settings Window (continued):

Meters tab: PgTg is in constant communication with the amplifier and radio. It seemed useful to display meter data on the PC. This tab simple allows you to select which of the five possible values of data to display. The TX slice frequency is also displayed, however it only updates when you change the TX slice frequency.

In the case of the KPA1500, the values for AMP FWD and TUNER FWD are identical so you can disable the TUNER bargraph. Likewise, the AMP SWR and TUNER SWR meters are fed to the radio with the same value. So, you may want to hide the TUNER SWR bargraph as well.

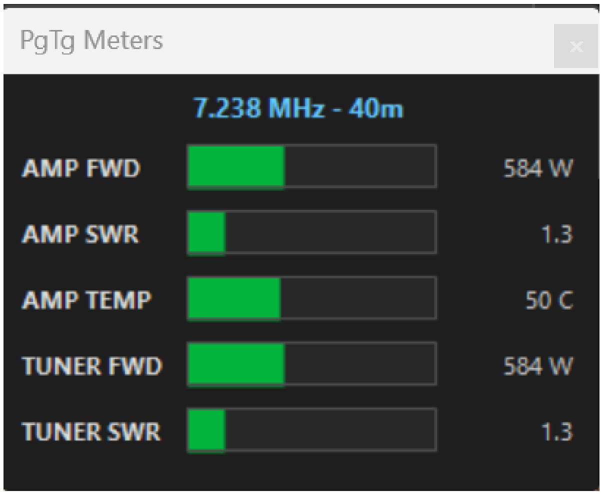
However, in the case of the KPA500 amplifier and KAT500 tuner, you can have different values. So, leave them all on! Also, since the KAT500 does not expose a CAT command that gives TUNER FWD, but does expose the raw ADC value from the coupler, PgTg does a straight line approximation of output power based on measured data points from an LP-100A



Select which meters to display in the Meter window:

☑ AMP FWD - Forward Power (Watts)
☑ AMP RL - Return Loss (SWR)
☑ AMP TEMP - Temperature (C)
☑ TUNER FWD - Tuner Forward Power (Watts)
☑ TUNER RL - Tuner Return Loss (SWR)

Most setting changes require an Bridge Restart to take effect, except for Power Control values which are enforced after OK is clicked.



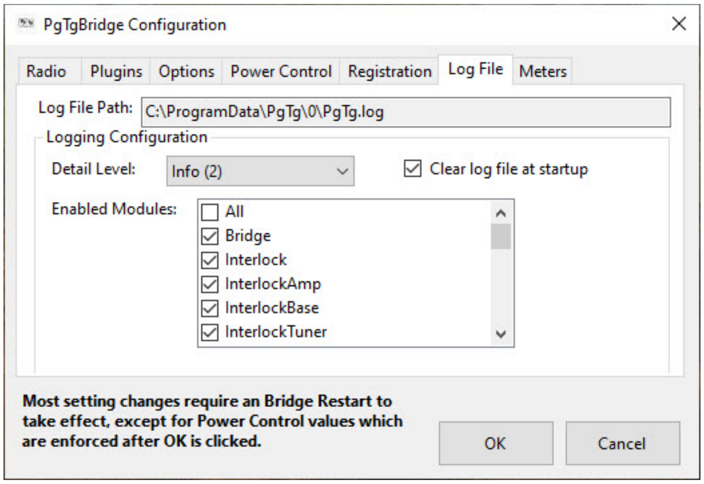
7.238 MHz - 40m

AMP FWD	584 W
AMP SWR	1.3
AMP TEMP	50 C
TUNER FWD	584 W
TUNER SWR	1.3

A note about Return Loss (RL)

Flexradio sends the measured values of "SWR" in terms of Return Loss, not as a Standing Wave Ratio. They named those meters "RL", so that is how PgTg has to refer to them. PgTg converts the SWR value read from the KPA1500, KPA500, and KAT500 into RL values in terms of dBm. The values displayed in SSDR and on the Maestro correspond well with the polled values from the amplifier and tuner.

Log File tab: PgTg creates a tracelog that will be helpful when diagnosing issues. You can set the detail level as desired. The log will automatically limit to a certain size, and will rotate around 5 different filenames.   By default, the log will be cleared each time the Windows Service starts, not to be confused with when PgTgBridge is started from the Controller app.



Log File Path: C:\ProgramData\PgTg\0\PgTg.log

Logging Configuration

Detail Level: Info (2)

☑ Clear log file at startup

Enabled Modules:
☐ All
☑ Bridge
☑ Interlock
☑ InterlockAmp
☑ InterlockBase
☑ InterlockTuner

Most setting changes require an Bridge Restart to take effect, except for Power Control values which are enforced after OK is clicked.

Checklist:

1. Read this guide! Get up to speed on what PgTg actually is doing for you.
2. Gather information on your hardware before installing PgTg. Amplifier IP address and MAC address are must-haves. If using serial ports, identify which COM port goes to what device! (KPA500 / KAT500) Make a note of the exciter drive levels you want in each band.
3. Pick a Windows computer for PgTg that runs all the time. It doesn't have to be powerful, as PgTg is written to be multi-threaded and runs all processes asynchronously. You should never see more than 5% CPU utilization in use.
4. Check the firmware versions on the KPA1500 and KPA500. Update to latest available.
5. Turn on your Flexradio, Amp and Tuner. In SSDR, make a fresh Profile Backup, saved somewhere safe.
6. Install PgTgBridge. Go through all of the tabpages in the Configuration screen. Plugins must be configured!
7. Save the settings then, use PgTgController to stop or restart PgTgBridge. You shouldn't have to restart the PC, but if you need to, just stop/start the PgTgService in the Control Panel/Services applet. PgTgController does not have to run for the integration to work. However, it is nice to see what is going on with it. If you accidentally close PgTgController in the Task Tray, just launch it again from the desktop shortcut again.

Odd Behavior:

PgTgBridge has been in Beta Test for months, but you might run into weird behavior from the radio or PgTg.. It has been in production use on my station, and the users never knew it. However... “There may be bugs here”

Currently PgTg is released as Trial version. But, there will be updates! Installing updates will be easy. You can just run the new installer MSI without uninstalling your currently installed version. Your settings are retained. The services do not have be stopped.

Throughout the development of PgTg, I have run into some very odd behavior that seems related to the Flexradio itself, namely related to the Profiles in the radio. I recommend making a backup of your FlexRadio configuraion from the Import/Export Profiles menu in SmartSDR.

If PgTg doesn’t seem to sync up with the hardware, or the meters don’t show data, restart the PgTg Service. If you notice that the radio reboots on it’s own, it’s likely going to need a full shutdown, remove DC power, then restart it again to return to normal.

If still experiencing odd behavior, you might have to perform the factory reset procedure on the radio, then restore your profile backup. (Power up holding the button until it turns white-- this will erase your profiles so you should have your profile backup handy).

Optional configuration using ser2net

It is possible for PgTg to control both the KPA500 and KAT500 using TCP.

These models use serial connections natively, but it is easy to use a Serial to TCP converter

(hardware device) or a Raspberry Pi with ser2net redirector. This means the Windows PC running

PgTg does not need to be near the amp or tuner. You will still need the two USB serial cables appropriate for the KPA500 and KAT500, except they will connect to the converters or a single Raspberry Pi.

How to configure ser2net on a Raspberry Pi for this job.

First, you should reserve the IP address for the Raspberry Pi in your Router/Firewall. It needs to have the same address every time it is booted. You could use a Static IP configuration in the Raspberry Pi, but that may become a headache for you diagnosing network issues later.

Linux assigns USB devices based on the physical USB hardware port in use. This could cause the port name to change if the cable positions are moved around. The best way to avoid this situation is to use the serial number of the FTDI chip in the cable as an identifier.

Steps to configure ser2net

Determine serial number of the FTDI USB cables you will be using for the KPA500 and KAT500 by inserting one at a time into the Raspbery Pi. From a shell prompt run:

dmesg

This displays the last few lines of the system log. Scan the information and make note of the serial number of that device and which device it is for. In this case, it was for the KAT500

```
[  238.695187] ftdi_sio 1-1.4:1.0: device disconnected
[  241.533996] usb 1-1.4: new full-speed USB device number 5 using xhci_hcd
[  241.680187] usb 1-1.4: New USB device found, idVendor=0403, idProduct=6001, bcdDevice= 6.00
[  241.680205] usb 1-1.4: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[  241.680219] usb 1-1.4: Product: FT232R USB UART
[ 241..680232] usb 1-1.4: Manufacturer: FTDI
[  241.680245] usb 1-1.4: SerialNumber: AK05UOD5
[  241.686874] ftdi_sio 1-1.4:1.0: FTDI USB Serial Device converter detected
[  241.687006] usb 1-1.4: Detected FT232RL
[  241.691455] usb 1-1.4: FTDI USB Serial Device converter now attached to ttyUSB1
```

Likewise, for the KPA500

```
[  346.400570] usb 1-1.1: Product: FT232R USB UART
[  346.400583] usb 1-1.1: Manufacturer: FTDI
[  346.400595] usb 1-1.1: SerialNumber: A10M20IX
[  346.409337] ftdi_sio 1-1.1:1.0: FTDI USB Serial Device converter detected
[  346.409477] usb 1-1.1: Detected FT232RL
[  346.413988] usb 1-1.1: FTDI USB Serial Device converter now attached to ttyUSB0
```

Now create a udev rule file with the following lines. Change the serial numbers to the ones you obtained above.

sudo nano /etc/udev/rules.d/10-elecraft.rules

```
KERNEL=="hidraw*", SUBSYSTEM=="hidraw", MODE:="777", GROUP="elecraft"
SUBSYSTEM=="tty", ATTRS{serial}=="A10M20IX", SYMLINK+="ttyKPA500"
SUBSYSTEM=="tty", ATTRS{serial}=="AK05UOD5", SYMLINK+="ttyKAT500"
```

(Ctrl+X , Y to save this new file.)

At this point, be sure to have both cables plugged into the Raspberry Pi.

To Reload the udev rules including the ones you just created, run this:

sudo udevadm control --reload-rules

Trigger the new rules to be applied by running this:

sudo udevadm trigger

As a sanity check, you should now have both devices showing up with shortcut links that will not change no matter which USB port you plug them into. Your ttyUSB0 / ttyUSB1 names may be different, depending on other serial devices you may have in use. It won't matter now. You will be using the symbolic link names for them instead. The udev rules will create the appropriate symbolic links based on the serial number.

Now get a list all of the tty devices mentioning USB. Look for your two USB devices on the standard USB0 / USB1 name convention. You should also see the links pointing to them.

```
ls -l /dev/tty* | grep "USB"
```

```
lrwxrwxrwx 1 root root         7 Feb 20 10:06 /dev/ttyKAT500 -> ttyUSB1
lrwxrwxrwx 1 root root         7 Feb 20 10:07 /dev/ttyKPA500 -> ttyUSB0
crw-rw---- 1 root dialout 188,  0 Feb 20 10:07 /dev/ttyUSB0
crw-rw---- 1 root dialout 188,  1 Feb 20 10:06 /dev/ttyUSB1
```

Now you are ready to install or update ser2net.

```
sudo apt update
sudo apt install ser2net
```

The next step is to configure ser2net. It will depend on what version of ser2net you have installed. The older versions use a .conf file. The newer versions starting with 4.0 use a .yaml file. The syntax is totally different so determine which is right for you.

To determine version

```
ser2net -v
```

If version 3.x, edit the conf file:

```
sudo nano /etc/ser2net.conf
```

Add these two lines:

```
ipv4,6001:raw:600:/dev/ttyKPA500:38400 8DATABITS NONE 1STOPBIT
ipv4,6002:raw:600:/dev/ttyKAT500:38400 8DATABITS NONE 1STOPBIT
```

Else, if you have version 4.0 or higher, you need to edit the yaml file. Change the file to have these lines and save the file:

sudo nano /etc/ser2net.yaml

```
%YAML 1.1
default:
  name: local
  value: true
  class: serialdev

connection: &con6001
  accepter: tcp,0.0.0.0,6001
  enable: on
  connector: serialdev,/dev/ttyKPA500,38400n81,local
  timeout: 600
  options:
    max-connections: 3

connection: &con6002
  accepter: tcp,0.0.0.0,6002
  enable: on
  connector: serialdev,/dev/ttyKAT500,38400n81,local
  timeout: 600
  options:
    max-connections: 3
```

Save the file in nano (Ctrl+x, Y)

Now restart the ser2net service:

s**udo service ser2net stop**

You can watch the log for ser2net by runing this in another shell. Watch for errors and correct the yaml file if needed.

journalctl -f -u ser2net

Then restart ser2net for the changes to take effect.

sudo service ser2net start

If ser2net started normally, you will see these two log entries from journalctl:

```
 systemd[1]: Starting ser2net.service - Serial port to network proxy...
 systemd[1]: Started ser2net.service - Serial port to network proxy.
```

For another sanity check, verify ser2net is listening on the two ports defined above. The :600 portion is a search filter that looks for lines with those characters. Since we are only interested in port 6001 and 6002, this will filter down the list.

```
 netstat -an | grep ":600"
```

```
tcp        0      0 0.0.0.0:6001            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:6002            0.0.0.0:*               LISTEN
```

If netstat shows these two lines, ser2net is working.

Open up PgTgController from the task tray, and go into Settings / Plugins.

Click the Plugin Manager button. Change the PgTg Plugin configuration to use TCP instead of Serial.

Be sure to enter the ports defined above.

First select the KPA500 plugin. In the Connection Settings area, change the radio button to TCP from Serial.

You will need to fill in the IP address of your Raspberry Pi. Fill in the port number configured in ser2net for the KPA500. In this example, we used 6001.   Likewise, select the KAT500 plugin, and make the same changes. Use the same IP address for the Raspberry Pi as for the KPA500, but enter the port that you configured in ser2net for the tuner. In this example, that port is 6002.

Before clicking OK to save all the setting changes, be sure to disable the KPA1500 plugin if it is still configured to be enabled. After saving settings, restart PgTg. After about 30 seconds, go back into the Raspberry Pi shell, and run netstat again.

```
 netstat -an | grep ":600"
```

```
tcp        0      0 0.0.0.0:6001            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:6002            0.0.0.0:*               LISTEN
tcp        0     10 192.168.111.241:6002    192.168.111.250:54503   ESTABLISHED
tcp        0      7 192.168.111.241:6001    192.168.111.250:54500   ESTABLISHED
```

You will see two more lines if PgTg has is configured correctly and using TCP instead of Serial.

In this example, the Pi's IP is on 192.168.111.241 and PgTg is running on 192.168.111.250.

That completes the changes needed to use TCP for control of the Elecraft KPA500 / KAT500 on TCP instead of USB.

Reminder:

As of now, the KPA500 does not understand the PTT over TCP commands used by the KPA1500. You must make sure to ***UNCHECK the PgTg option for "Key amplifier via Network"*** until the KPA500 firmware supports the new commands. You will get an "Interlock Error" on the radio client when attempting PTT if you forget about this step!Warren Merkel, KD4Z

info@kd4z.com

Support:

A message group on groups.io has been created to discuss the use, features and issues regarding PgTgBridge. To join, navigate to: https://groups.io/g/PgTgBridge

For the latest news and installer visit https://kd4z.com

Revisions:

1 – In sync with PgTgBridge Installer version : 1.26.960.1 - February 10, 2026

2 – Added ser2net instructions. – February 23, 2026

3 – Corrected ser2net instructions. Changed syntax and limited to ipv4. – March 2, 2026

4 - Edits to setup procedure text.