The purpose of this design is to allow a functional and clean interface between an ICOM IC-705 transceiver and a Xiegu XPA125B amplifier. It is known that the ALC port on the IC-705 is susceptible to damage with 3rd party amplifiers. There is also no control between the IC-705 and XPA125B for automatic band switching. Lastly, driving too much power from the IC-705 to the XPA125B can cause damage to the transceiver. This design addresses all of these issues. First, the connection from this device to the IC-705 is over Bluetooth, removing the need for a physical connection to the transceiver thus eliminating the sensitive ALC port. Second, this device will receive band information from the IC-705 and pass that into the XPA125B over a 6-pin Mini Din cable. Also, over the Mini Din cable, PTT activation will be passed from the Bluetooth of the IC-705 to the XPA125B. Third, each band requires a different amount of RF Power to be sent from the IC-705 to the XPA125B to achieve maximum amplifier output without driving beyond the 100W of the amplifiers output capabilities or its 5W maximum input. This device monitors the band selected on the radio and auto adjusts the RF Power of the IC-705 to achieve these goals. The device and its design are outlined below.

Here is a link to a short video demonstration of it all working:

The device is a combination of a custom PCB board, a small number of electrical components and an Arduino ESP32 TTGO T-Display. Software is then loaded on the TTGO board to make this all work. You will need to be able to solder fairly small components to the PCB Board, strip some wires, heat shrink a cable and also load the software on the TTGO. I'll also provide step by step testing instructions to make sure everything is correct before plugging everything together. But I do want to caution that I am not responsible for any damage or destruction to your equipment. DO THIS PROJECT AT YOUR OWN RISK!

I need to acknowledge several folks for taking this project to 99% completion of what I wanted.

http://www.carnut.info/IC-705/ICOM_IC-705.html#BT
: It seems he did the original design

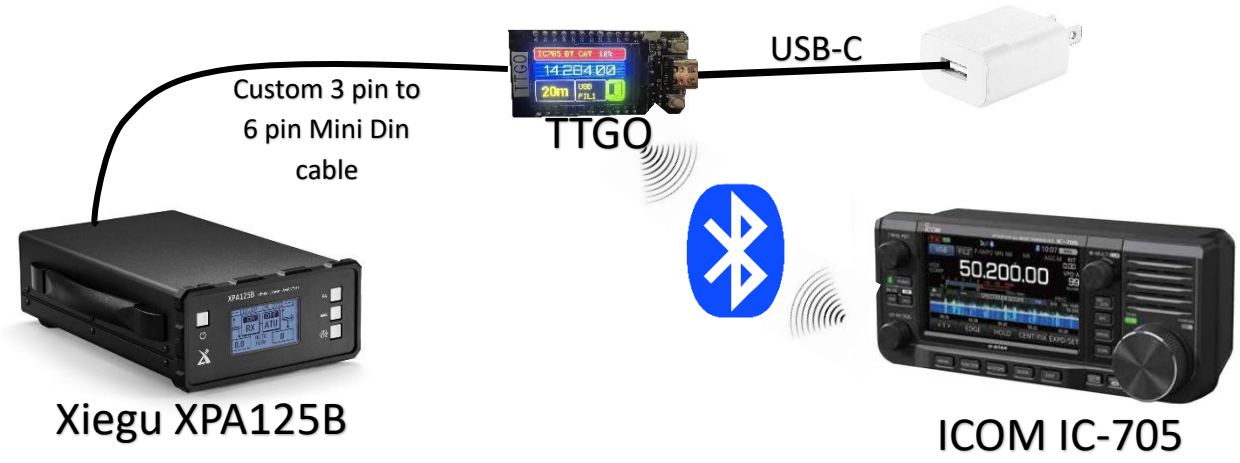
https://github.com/WillyIoBrok/CIVmasterLib
: This is the core. It allows CI-V commands to and from the IC-705

https://github.com/PE1OFO/IC-705-BlueTooth-Controller
: He designed the original PCB and made changes and updates to the software
: His PCB had some issues so I created a new one which works perfectly

https://github.com/dl1bz/ESP32-IC705BT-PA
: The software again updated by DL1BZ and I used this to fork to add more features

What did I do? I created a new PCB design to address the issues in the original design. I updated the software with several new features around RF Power settings and display and also how the frequency is displayed. I put this document together in hopes it would make it easier to build, test and deploy the solution.

Here is a visual representation of the connectivity:



Parts List:

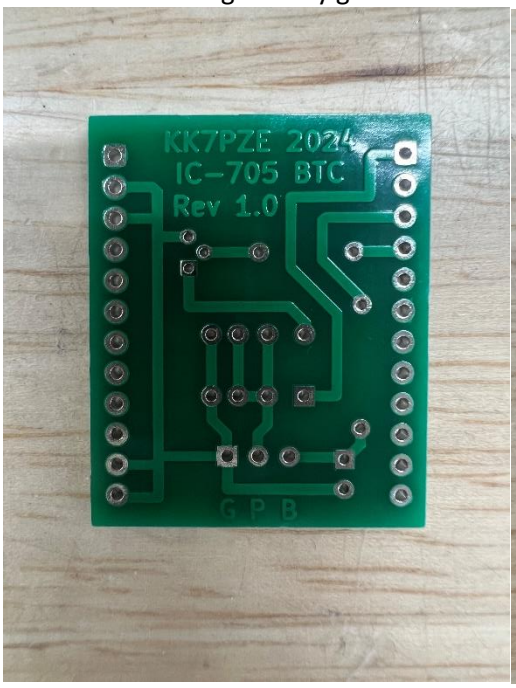
- Xiegu XPA125B Amplifier
  - Recommend updating to v2.0.0 Build 014
- ICOM IC-705 Transceiver
  - Recommend updating to v1.32
- PCB – See below for more info
- LilyGO TTGO T-Display V1.1 ESP32
  - https://www.amazon.com/dp/B099MPFJ9M
- Through hole angled pin header, 3 pins, 2.54mm pitch 6mm length
  - https://www.amazon.com/dp/B00VG9UDO2
- 2 x 12 pin/hole at 2.54mm
  - https://www.amazon.com/dp/B01MFBPH9N
- 2 x pins at 2.54mm
  - Part of above kit
- 3 pin sockets
  - https://www.amazon.com/dp/B096D849KN
- Transistor BC547 (will also receive BC557 in kit. Don't use those)
  - https://www.amazon.com/dp/B07Y27GNX1
- 1000-ohm resister 5%
  - https://www.amazon.com/gp/product/B07QXP4KVZ
- 4700-ohm resister 5%
  - Part of above kit
- 22uF 50V 5x11mm Electrolytic Capacitor
  - https://www.amazon.com/dp/B0CMQ9PSQW
- HFD4/5
  - https://www.ebay.com/itm/385922956466
- Mini DIN 6 Cable
  - https://www.amazon.com/gp/product/B07KVFY456
- USB Charger
  - Any will do. USB-A or USB-C
- USB Cable
  - End that goes into TTGO is USB-C. The other end should match your USB charger.

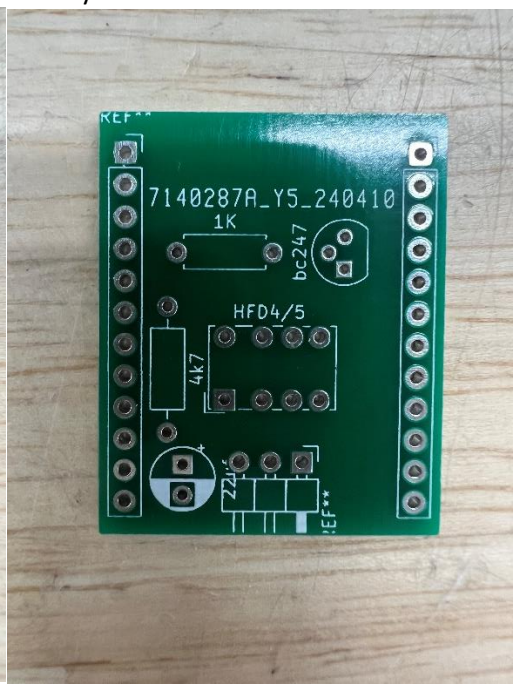
Tools:

Sharp side cutters
Wire Strippers
Soldering Iron with a fine tip
Rosin Core Solder
Multimeter
Magnifying glass if old like me
Colored tape…Red, Blue, White, something other than black
Glue or hot melt glue gun
Something to make a base to hold it upright when done. 3D case if you're cool. I'm not 🙁

The PCB:

A PCB is a generally green colored board you see on electronics. Mine looks like this:





Download the ZIP file which contains the files for this project.

IC705-TTGO-XPA125B.zip

Unzip the file somewhere useful on your computer. The zip will contain

Instructions for IC-705 to XPA125B build.pdf

This document

CIV_Template.zip

This is the Arduino software we will load on the TTGO board

PCB – Joe v1.0.zip

The KiCad and Gerber files needed to order the PCB.

Getting the PCB

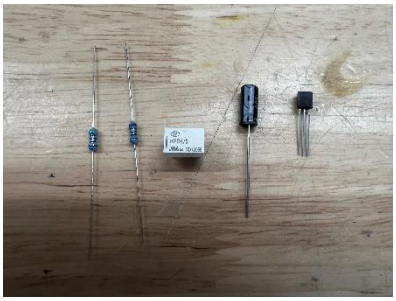
I don't sell them. I used JLCPCB.com to order mine. Simply setup an account, log in, select "My File" and pick "Upload Gerber of 3D files" Select the "Add gerber file" and upload the ZIP file. You can leave all the other options alone. Save to cart, check out and soon you'll receive 5 of these PCB Boards. The good news is that most of the components that you order to solder on come in packages of multiples so you can build several boards to have as a backup or to give to that special someone.

Building the hardware:

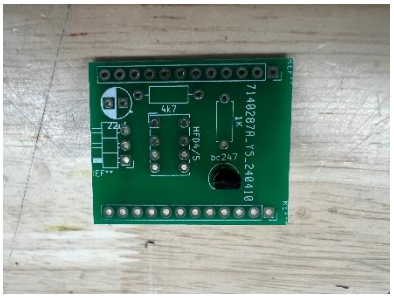
Time to fire up the soldering iron and get out the side cutters

1 – It's easier to solder the inner parts first like the resisters, transistor, switch first. Then do the capacitor, then the 3 pin socket. Finally solder on the long connectors.

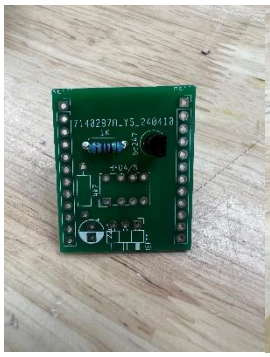
Trim these to the right size using your side cutters.



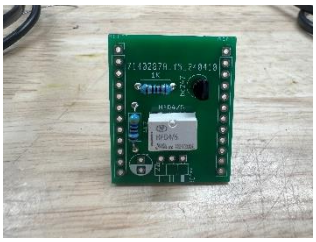
These are the electronic components to solder on

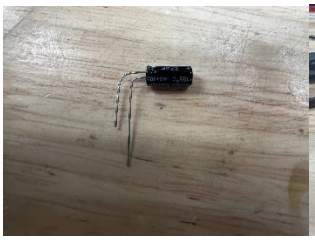


The nice thing is the board has outlines and labels for all the parts. I started with the transistor



Next, I did the resistors. Don't mix these up. Keep them in their bags until needed. You can also read the bands to find out which is which. Resistors are not polarity sensitive so the direction does not matter.


Then I did the switch.

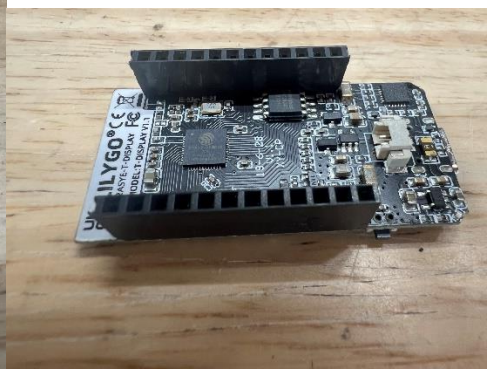


Last for the electronic components is the capacitor. Since this is an electrolytic capacitor it's important to get the polarity correct. There is a stripe filled with – on the capacitor and there is a + sign on the PCB. Make sure to NOT align those. The reason for the bending of the leads was so the capacitor would fit under the TTGO. If you install it straight up the TTGO board will not fit. Maybe you can find some smaller capacitors to work but these are what I used.

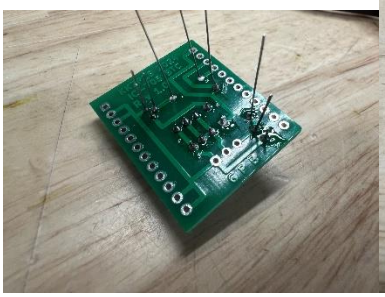
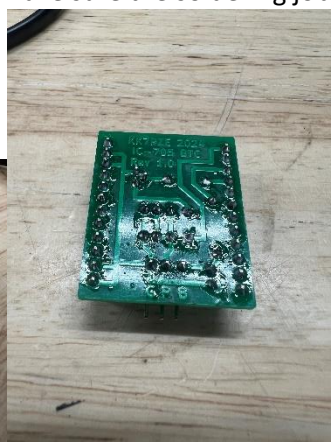
2 – Now lets put on the 3 pins


Make sure to keep these level with the board so it doesn't go on wonky making the plug go on funny.

3 – Set the pins in place on the TTGO and then plug it into the main board. This will keep everything aligned while you solder these to the TTGO board and PCB board.

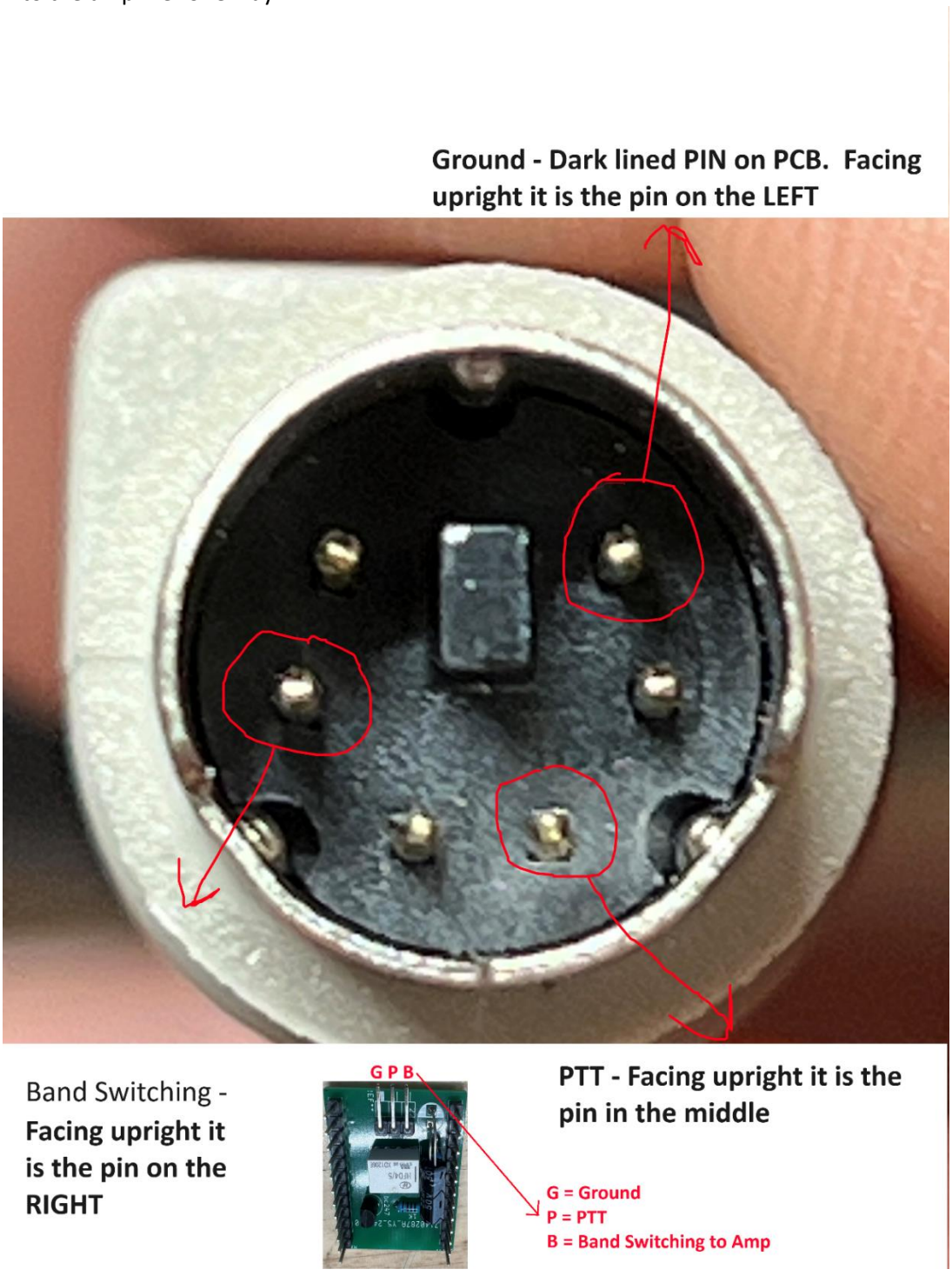
If you didn't for each part, make sure to trim the long lead strands on the back of the board where you soldered. It's also good practice to use your multimeters continuity tester to check the tip of each wire lead to the next component to make sure the soldering job is good.

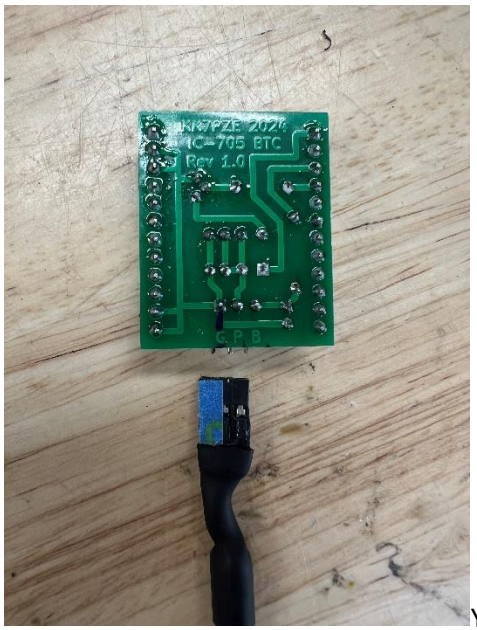



Don't judge my soldering abilities. It worked!

Good job with the board. Time to make the cable.

Take the cable you ordered and cut it in half. I did that so I could make two of them…You know, for a backup or a friend. Keep it longer if you need though.
Strip back the wires on the cut end and now it's time to figure out which ones go to what pins on the cable. Document this carefully. Doing this wrong could blow up your amplifier!!! We are only using 3 of the wires. Notice the orientation of the plug with a dimple at the top and dimples offset on the lower left and right. Also notice the offset rectangular piece of plastic. This is important as the plug only goes into the amplifier one way



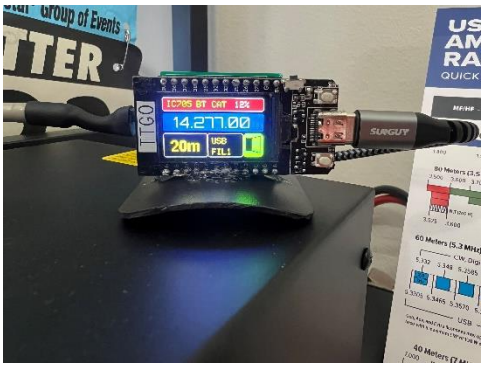


You'll crimp and solder the wires to the metal sockets and put them into the plastic 3-hole plug. Look at the metal pieces as they have a little tab on them that locks into the plastic piece. You'll notice that I put some blue tape on the GROUND pin of the connector and I highlighted in blue on the board above the "G" for Ground to make sure I didn't plug things in backwards. I also used heat shrink to cover the plug to cable area. You could also use tape. Make sure the extra wires from the cable are not shorted to anything or each other as that would get passed into the XPA125B and could cause it to fail.

I used a hot melt glue gun to glue the plug onto the board so it wouldn't fall off. The thing I like about hot melt glue is I can peel it off if I need to.

Now, lets mate the TTGO to the PCB correctly. The way I remember is the connectors for the cable to the amp and the USB cable face opposite directions.

Here they are sitting next to each other in the correct orientation.



Here you can see them mated together on my little plastic stand with the USB going out the right and the custom cable going left. Ignore the display you're seeing right now, next up we need to load the software so you can have this sexy beast working for you.

Working with the TTGO board to load the software:

1 - UnZip the CIV_template.zip

2 – Download the Arduino IDE
https://www.arduino.cc/en/software

3 – Download the library to control the TTGO display
https://github.com/Bodmer/TFT_eSPI
Select "<> Code" and pick Download ZIP

4 – Download the CIVmastLib
https://github.com/WillyIoBrok/CIVmasterLib
Select "<> Code" and pick Download ZIP

5 – Install the Arduino IDE and then run it

6 – Under Sketch, select Include Library and select add .zip library. Select the TFT_eSPI ZIP.

7 – Do the above again but add the CIVmasterLib ZIP.

8 – Under Tools, select Board and then Boards Manager. In the search engine, enter "esp32". Select the "esp32 by Espressif Systems"

9 – Under Tools, select Board and then click is "esp32" and in the list select "esp32 dev module"

10 – Exit the Arduino IDE

11 – Go to the Arduino libarary folder. For Windows this is under Documents/Arduino/libraries. Select that folder and then go into the TFT_eSPI folder. Right click on the "User_Setup_Select.h" file and select "Edit". Scroll down and you will see a line that says:
#include <User_Setup.h> // Default setup is root library folder
Put // in front of this so it looks like this:
//#include <User_Setup.h> // Default setup is root library folder

Now keep scrolling until you see this line:
//#include <User_Setups/Setup25_TTGO_T_Display.h> // Setup file for ESP32 and TTGO T-Display ST7789V SPI bus TFT
Remove the // at the beginning of this line so it looks like this:
#include <User_Setups/Setup25_TTGO_T_Display.h> // Setup file for ESP32 and TTGO T-Display ST7789V SPI bus TFT
Save the file and exit the editor. Now select edit again to make sure the changes are really saved in there!!!

12 – In the ZIP package you downloaded from me for this project t

13 – Launch the Arduino IDE

14 – Select File, Open and select the CIV_template" folder and open the "CIV_template.ino" file. This will open all the files in the project in a new Arduino IDE. You can close the blank sketch from the original install

15 – Select the Blue Check to verify the application. If at the end you see no errors you've done everything correctly so far.

16 – Open device manager on your computer. Look to see if you have a "Ports (COM & LPT)" section. If not, go to the next step. If so, expand it and note the current ports. Now go to the next step

17 – Connect the USB-C cable to the TTGO and to your computer. The TTGO display should boot up. Now look in Device Manager and note the new COM port.

18 – In the Arduino IDE select Tools, Ports, select the COM port.

19 – Now select the right arrow icon to verify and upload the program into the TTGO. If it works, you will see the program message to pair the device to your IC-705. Congrats, we are ready to start testing!

Testing:
1 – DO NOT PLUG THE CABLE FROM THE TTGO TO YOUR XPA125B YET

2 – While the TTGO is powered up, turn on your IC-705.

Select MENU, SET, Bluetooth Set
Bluetooth = ON
Auto Connect = ON
Pairing/Connect / Device Search / Search Data Device
When the ESP32-XPA125B shows up, wait till searching is complete

Select ESP32-XPA125B, Connect?, YES, CONFIRM
You should now see the display show something like this:

You are now connected from the TTGO to the IC-705.
In this example the RF Power of the IC-705 has been set to 12%
The IC-705 is on the 20 meter band
The IC-705 is tuned to 14.284.00
The IC-705 is using upper sideband
The IC-705 is using filter 1
The green box means the IC-705 is in receive mode. It will turn Red when transmitting

3 – Let's start by checking the display and its functionality. Change frequencies, change bands, change modes, change filters. Do you see them change on the display? The RF Power should also be changing to match the values per band:

```
160M =   0%
80M  =   5%
60M  =   0%
40M  =  50%
30M  =   0%
20M  =  12%
17M  =  11%
15M  =  10%
12M  =  11%
10M  =  50%
6M   =  50%
2M   = 100%
70CM =   0%
```

These are the RF Power values I'm using based on my DX Commander Class Plus 80 Antenna. You should set the values in the z_userprog.ino for BandPower1 and BandPower2 based on the table below it for the values that you want. The ICOM RF Power is based on two fields. BandPower1 and BandPower2. So, looking at these arrays if you wanted 50% power on the 30 meter band you'd put a 1 in BandPower1 in the 5th entry and put a 40 in BandPower2 in the 5th Entry. The BandPower1 Array are on lines 206-210 and the BandPower2 Array are on lines 212-216. The description of your selection are on lines 217-218. The big table that shows all of the BandPower1 and BandPower2 combinations for 0% to 100% are on lines 219-322.

4 – Now let's check that the hardware is working. You'll need your multimeter now. Go back up in this document and look at the pinout for the cable. You'll see the ground and PTT pins. Set your multimeter to continuity and look at it. It should show there is none. Now, press your PTT button and you should get continuity. Now, we need to make sure the Ground and Band pins are putting out the correct voltage

based on the band you select in the radio. Set your multimeter to DC voltage testing and if you need, select the 5 volt setting. Now, change the bands on your radio and note the voltages against the below chart. They should be pretty close to these values:

BAND	LEVEL(m V)	BAND	LEVEL(m V)	BAND	LEVEL(m V)
1.8 MHz	230	14.0 MHz	1380	50.0 MHz	2530
3.8 MHz	460	18.0 MHz	1610	----	----
5.0 MHz	690	21.0 MHz	1840	----	----
7.0 MHz	920	24.0 MHz	2070	----	----
10.0 MHz	1150	28.0 MHz	2300	----	----

Of course, the IC-705 doesn't do 5MHz so skip that.
Also, of note is when you select 144 or 430 the amp will switch to 160M band mode. The amp doesn't support these bands so make sure to turn off the amp before using them. Also, the software will not trigger the PTT of the XPA125B when these bands are selected.

My multimeters says:

Band		**vDC**
160m(1.8)	=	0.234
80m(3.5)	=	0.471
40m(7)	=	0.940
30m(10)	=	1.173
20m(14)	=	1.407
17m(18)	=	1.645
15m(21)	=	1.88
12m(24)	=	2.10
10m(28)	=	2.34
6m(50)	=	2.58
2m(144)	=	0.001
70cm(430)	=	0.001

Let's put this thing to work
1 - If all your voltages come in good and on the correct pins, power the amp, TTGO and IC-705 off.

2 - Plug the 6 pin Mini Din cable into your Amp

3 - Power on the TTGO

4 - Power on the XPA125B. Make sure the band setting is set to auto. To change from manual to auto press and hold the BAND button until it switches

5 - Now, turn on the IC-705 and it should automatically pair to the TTGO. You should see it connect and the display should work the same as our previous testing. Now, change bands and make sure your XPA125B also changes bands. Now, I'd also check that the RF Power displayed on the TTGO matches your IC-705. To do this, press the Multi-CLR button and look at the RF POWER. Does it match? If this has all worked, then you are done!

6 - Now, I'd suggest doing one more set of testing. Go to a free frequency on each band, set the mode on your radio to RTTY and key up the PTT. Look at the output on the XPA125B. You don't want it to go above 100 watts. Tune the RF Power as needed from the IC-705. Never go above 50%. Once you have it where you want it, note the band and RF Power. You can adjust the z_userprog.ino file as mentioned above, save the update and verify and upload again to the TTGO using the prior procedure for the Arduino IDE.

Good luck and have fun!

Joe
KK7PZE
https://www.qrz.com/db/KK7PZE