



TNC-Pi9K6 9600/1200 Baud Assembly Instructions & Operating Tips

July 25, 2018

John Hansen W2FS
Coastal ChipWorks

Special Thanks to: John Wiseman, G8BPQ
Mark Griffith, KD0QYN
Ed Slingland, N2WD

Email: john@hansen.net

Website: <http://tnc-x.com/TNCPi9K.htm>

Contents

| | |
|---|----|
| Introduction | 3 |
| Assembly Instructions | 3 |
| Parts List | 3 |
| Assembly | 5 |
| Schematic Diagram | 8 |
| Parts Layout..... | 9 |
| Connecting the Radio | 10 |
| Configuring the Raspberry Pi | 11 |
| Configuring the TNC | 11 |
| Reprogramming the Teensy's Firmware | 13 |
| Support | 15 |

Introduction

Congratulations on your purchase of TNC-Pi9K. This board will allow you to transmit and receive either 1200 baud or 9600 baud packet radio transmissions when mounted on a Raspberry Pi. This manual is very much a work in progress and substantial additions and changes will occur. At this point, the manual is designed to guide you through the assembly process only.

Assembly Instructions

Parts List

Your kit should come with the parts listed in the table below. You can use the two columns of checkboxes to track your progress. As you inventory your parts, put a checkmark in the first column. Once you've installed a part, put a checkmark in the second column.

| Inventoried | Installed | Designation | Description | Notes |
|--------------------------|--------------------------|--|---------------------------------------|------------------------|
| <input type="checkbox"/> | <input type="checkbox"/> | C1, C2, C4, C5, C6, C11, C14 | 100 μ f electrolytic | Polarized |
| <input type="checkbox"/> | <input type="checkbox"/> | C2 | 0.1 μ f monicap | |
| <input type="checkbox"/> | <input type="checkbox"/> | C5 | Install jumper wire, not capacitor | |
| <input type="checkbox"/> | <input type="checkbox"/> | C8 | 1 uf cap (blue) | ***See Note Below |
| <input type="checkbox"/> | <input type="checkbox"/> | R1, R15 | 47K resistor | Orange, purple, yellow |
| <input type="checkbox"/> | <input type="checkbox"/> | R4, R5, R6, R12, R13 | 620 ohm resistor | Blue, Red, Brown |
| <input type="checkbox"/> | <input type="checkbox"/> | R2, R3, R10 | 10K resistor | Brown, Black, Orange |
| | | | | |

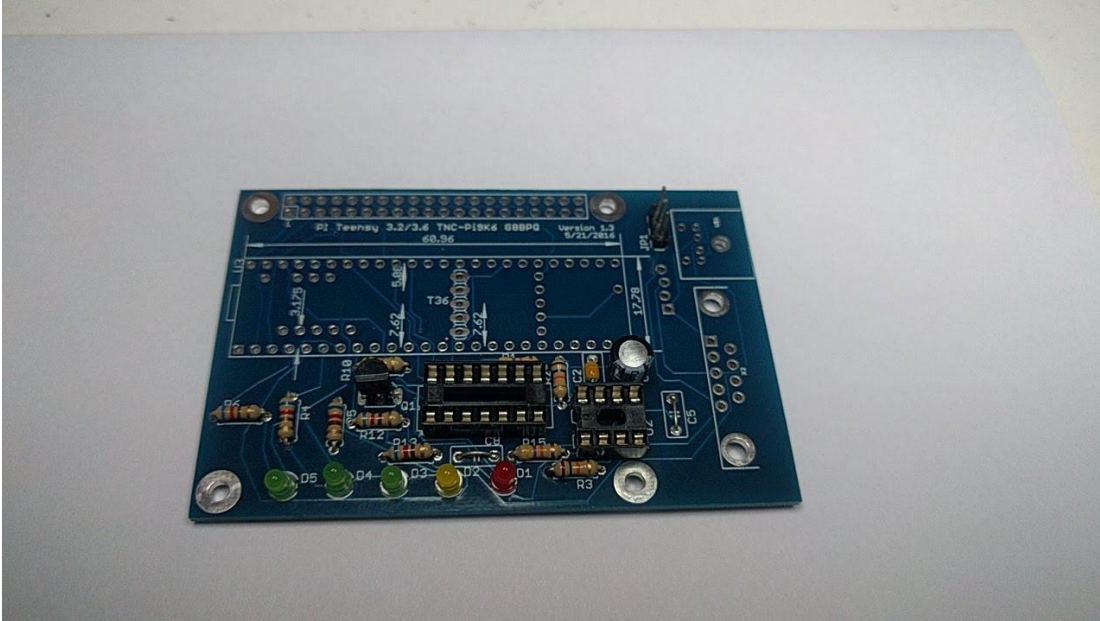
| | | | | |
|--------------------------|--------------------------|-----|--------------------------|--|
| <input type="checkbox"/> | <input type="checkbox"/> | D1 | 3mm Red LED | Polarized. Short lead toward R3. |
| <input type="checkbox"/> | <input type="checkbox"/> | D2 | 3mm Yellow LED | Polarized. Short lead toward D3. |
| <input type="checkbox"/> | <input type="checkbox"/> | D3 | 3mm Green LED | Polarized. Short lead toward D4. |
| <input type="checkbox"/> | <input type="checkbox"/> | D4 | 3mm Green LED | Polarized. Short lead toward D5. |
| <input type="checkbox"/> | <input type="checkbox"/> | D5 | 3mm Green LED | Polarized. Short lead toward corner of board. |
| | | | | |
| <input type="checkbox"/> | <input type="checkbox"/> | Q1 | PN2222 transistor | 3 pin, flat side |
| | | | | |
| <input type="checkbox"/> | <input type="checkbox"/> | U1 | MCP 4261 | 14 pin IC with socket. Notch toward Q1 |
| <input type="checkbox"/> | <input type="checkbox"/> | U2 | MCP6023 Op Amp | 8 pin IC with socket Notch toward U1. |
| | | | | |
| <input type="checkbox"/> | <input type="checkbox"/> | JP1 | 3-pin header | With shorting jumper |
| | | | | |
| <input type="checkbox"/> | <input type="checkbox"/> | | 6 Pin MiniDin Jack | Clip off extra pins on side of part. |
| <input type="checkbox"/> | <input type="checkbox"/> | | 9 Pin D-Sub Jack | |
| <input type="checkbox"/> | <input type="checkbox"/> | | 40 Pin Header | Install with pins upward through the board. Leave a small space between the board and the header |
| <input type="checkbox"/> | <input type="checkbox"/> | U3 | Teensy Board version 3.2 | |
| | | | | |

| | | | | |
|--------------------------|--------------------------|--|-----------------------|---|
| <input type="checkbox"/> | <input type="checkbox"/> | | Printed Circuit Board | |
| <input type="checkbox"/> | <input type="checkbox"/> | | Spacer and Screws | For providing support for TNC-Pi9K board on Raspberry Pi. |

***Most radios do not require C8 and for those you can simply install a wire jumper here (like C5). However occasionally radios have receive audio that is DC coupled (no capacitor in series with the audio output). For those, you should install this capacitor. So far the only radio that has been reported to have this issue is a Kenwood TS-V71.

Assembly

- First install all of the parts that lie flat on the board. Note that all of the components are installed on the silkscreen side of the board. Install
 - a. Capacitor C2.
 - b. A jumper wire (you can use the clipped leads from C2) at C5 and C8.
 - c. All of the resistors.
- Next install the IC sockets. The notches on the sockets should go toward the left.
Do not plug the chips into the sockets at this point.
- Install the LEDs. The short leads on the green and yellow LEDs should go to the left, while the short lead on the red LED should go to the right.
- Install the transistor, making sure that the flat side of the transistor lines up with the flat side of the part outline on the board.
- Install capacitor C1. It is polarized. Make certain that the stripe on the side of the capacitor goes in the hole that is closest to the 8 pin IC socket.
- Install jumper JP1. This jumper determines which pin on the mini-DIN connector is receive audio. It is different for 9600 baud than it is for 1200 baud. Place the included jumper on the side closest to the edge of the board for 9600 and the side further away for 1200 baud. Below is a shot of the board as it will look at this point.



- Install the Teensy board at U3. The board should be installed so that the USB connector is farthest away from JP1. You can clip the pins after you have completed soldering but be careful doing this because they will have a tendency to fly quite a ways when they are clipped.
- Install the mini-Din jack. Before you do this you will have to clip off the two metal fins on the outside edges of the jack. There are no holes on the board for them to go into.
- Now install the 9 pin D-Sub connector. Ensure that you push it all the way in so that it is flush against the board. In addition to soldering the pins, you'll gain mechanical stability by soldering the pins that go into the large round holes on the sides of the connector.
- Solder in the 2 x 40 pinheader.. It is the only part that is installed through the bottom of the board.
 - a. Start by installing this jumper on your Raspberry Pi and then
 - b. Lower the TNC-Pi9K6 board onto the connector so that the body of the connector is on the BOTTOM of the TNC-Pi9K6 board.
 - c. Now you'll need to solder the board so it is about a half millimeter from all the way down to the base of the connector in order to prevent the board from bumping into the USB connector on the Pi. It is not necessary to solder all of the pins. You should at least solder the 4 pins in the corners (for stability) and the first six pins on each row (pins 1 – 12).

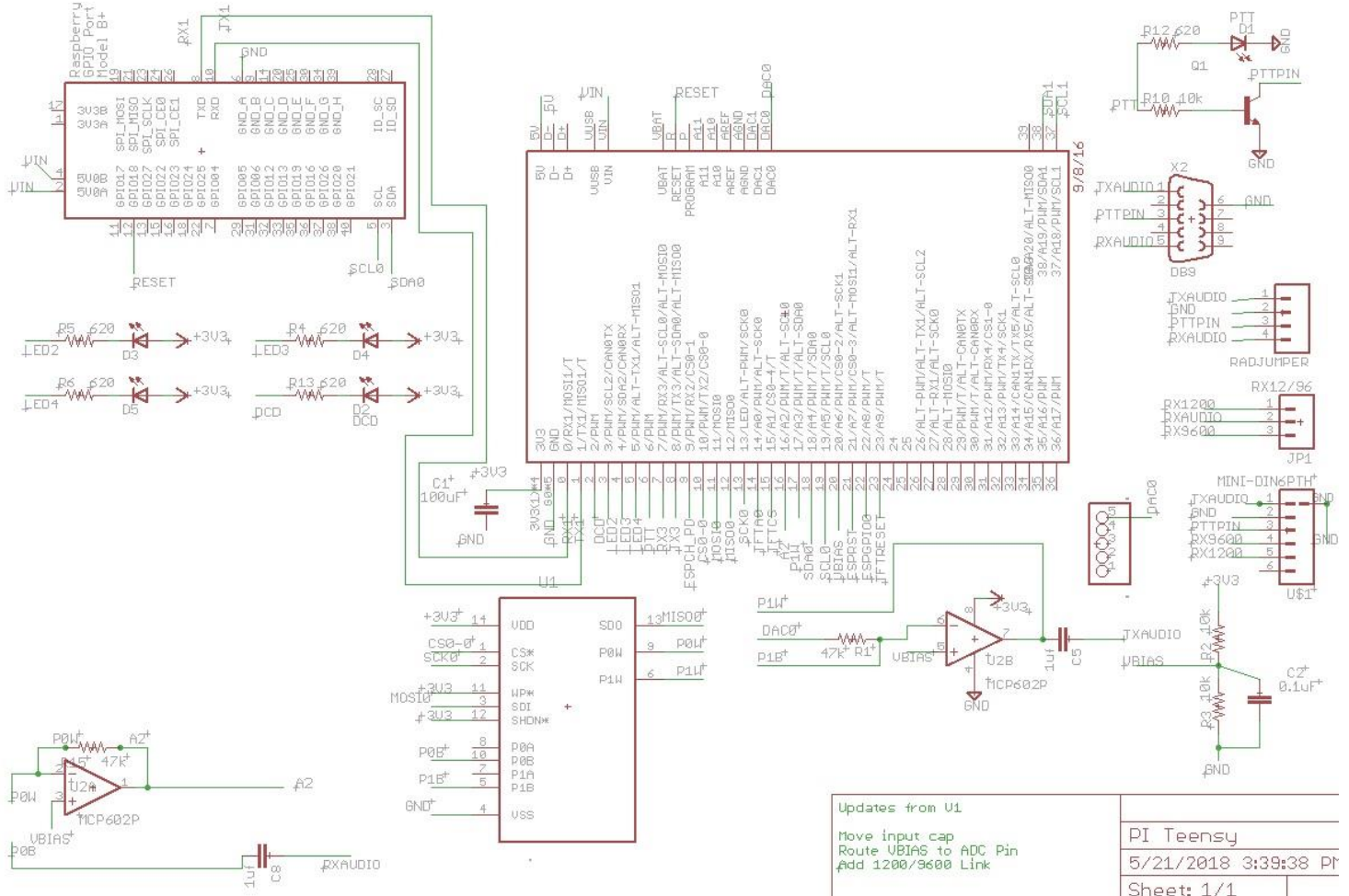
Your best bet is to carefully trim the solder leads on the part of the TNC-Pi9K6 board that will be on the mini-Din connector side of the Pi to get as much clearance as possible. You might also want to put a piece of insulating tape on top of the Pi's USB connector just to be on the safe side.

The pins on the 40 pin header will allow you to stack a second TNC-Pi9K6 on top of the first one if you choose to do so.

- Install the ICs in their sockets making sure that the notch on the IC matches the notch on the socket. The notches on the ICs should be on the left side of each of the chips.
- I've included 2 4-40 screws and a 21/32" spacer. You can use this to provide some additional mechanical stability by putting it between the hole in the Pi and the hole in the TNC-Pi9K. Use one of the mounting holes near the row of diodes for this purpose. If you are stacking 2 or more TNC's you might also find it preferable to use a spacer that is a male to female, rather than female to female. These male to female spacers are available on the TNC-Pi9K6 website.

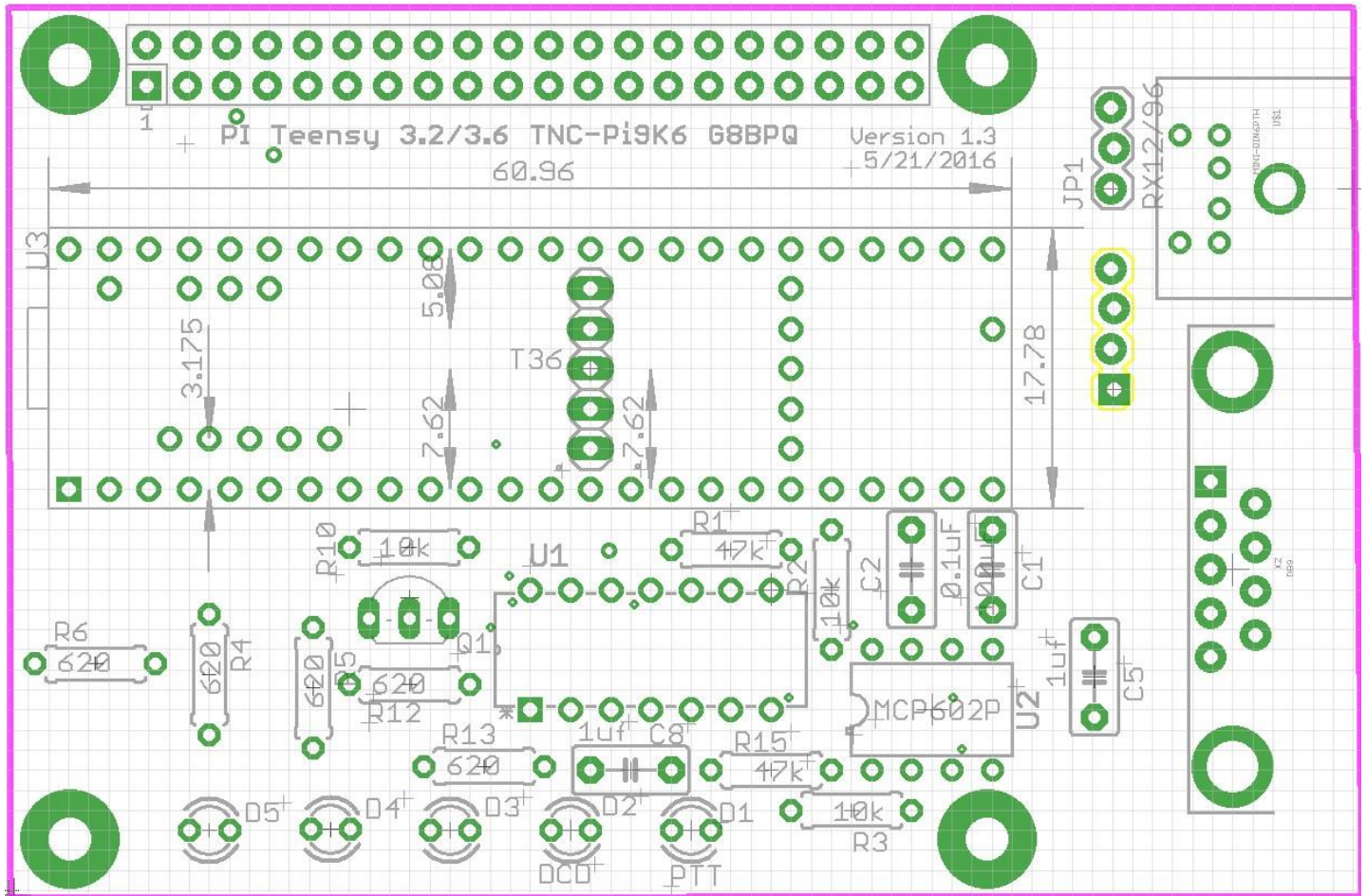
Congratulations, you're done assembling the TNC-Pi9k.

Schematic Diagram (for TNC-Pi9K)



(see next page for parts layout diagram)

Parts Layout (for TNC-Pi9K6)



Connecting the Radio

There are 3 ways to connect the radio to the TNC: You can use the mini-DIN connector, the 9 Pin D-Sub connector, or connect directly to the four holes that are just below jumper JP1. The mini-Din connector is perfect for the many transceivers that have a 6 pin packet connector. This includes nearly all radios that support a 9600 baud connection. In that case, you just need to get a 6 pin mini-Din to 6 pin mini-Din cable with plugs on both ends. These are widely available and are sometimes referred to as PS/2 cables. Supplies include Newegg and Amazon or you can order one from the Coastal ChipWorks website. Be careful when choosing one of these cables. They were generally used for mouse or keyboard connections in some cases not all six connections were wired (because mice and keyboards didn't need all of them). Make sure you get one that has all six connections wired.

The 6 pin mini-DIN packet radio connector had a separate connection for 1200 baud packet receive and 9600 baud packet receive. JP1 is used to select which of these wires the TNC is connected to. Place the jumper near the edge of the board for 9600 baud and near the center of the board for 1200 baud.

A second way to connect the TNC to your radio is to use the 9 pin D-Sub connector. The wiring here is the same as for Kantronics TNC's. It is as follows:

Pin 1 (the square pad): TX Audio
Pin 3: Push to Talk (PTT)
Pin 5: Receive Audio
Pin 6: Ground

The third way to connect the TNC to your radio is using the four holes that are below JP1. If you choose this method configure them as follows:

Pin 1: (the square pad): Transmit Audio
Pin 2: Ground
Pin 3: Push to Talk (PTT)
Pin 4: Receive Audio

Please note that you cannot use just any radio to transmit data at 9600 baud. Unlike 1200 baud packet, at 9600 baud data is NOT transmitted by sending tones over the air. A 9600 baud signal requires that receive audio be connected directly to the radio's discriminator. This will require that you either purchase a radio that is designed this way, or you modify your radio to make this connection. Modification of radio's that are not designed to support this connection is beyond the scope of this manual.

Many radios use an industry standard connector (6 pin mini-Din) with standard pin-outs to make this connection. These radios include the Kenwood TM-71A and TM-D710, the Yaesu FT-7900, FT-817, FT-857. Unfortunately, most manufacturers have only added this feature to their higher end radios. This was not always the case. I have a Yaesu FT-1500 2-meter radio that I bought some years ago for \$125 new that works great at 9600 baud. While I have not tested it, the Alinco DR-135 (~\$160) appears to support 9600, but with a non-standard connector.

Configuring the Raspberry Pi

(NOTE: If you are using Mark Griffith's PiGate Software SD image, ignore this section entirely).

The following instructions were based on a downloaded image of the "Stretch" version of the operating system. I did this by downloading Raspbian, not NOOBs.

It might be a good idea to make sure your Pi operating system is up to date. You can do this from the command line by entering the following (this assumes you have either hard wired or wireless network connectivity):

```
sudo apt-get update  
sudo apt-get upgrade
```

With previous version of the operating system, it was necessary to edit some of the configuration files in order to get the serial part to work. With the Stretch version of Raspbian, this is no longer necessary. Go to the command prompt (by clicking on the black screen icon at the top of the GUI desktop. Run the following: `sudo raspi-config` A menu will come up. Scroll down to item 5 using the arrow keys and select: "Interfacing Options" Scroll down to item P6 Serial and select that. It will ask you if you want to be able to get a log in shell via the serial port. Say No. It will ask you if you want the serial port enabled. Say Yes. You'll go back to the main menu. Tab over to and you'll have the opportunity to reboot. Do that, and you are done. There are no files to edit. Nothing! Ok, there is one catch. When you address the serial port you have to do it as `/dev/serial0`. Using the older version of the operating system it was called `/dev/ttyAMA0`. Note that these configuration instructions work with any version of the Pi.

Configuring the TNC

Mount the TNC-Pi9K6 on your Raspberry Pi and apply power. If all is well you should see all of the LEDs on the board flash on except for the red PTT LED. Also, place

your finger lightly on the Teensy board and make sure it does not get really hot. If it does, remove power immediately and debug the problem. As shipped, the TNC-Pi9K6 is set for 9600 baud over the air data transmission/reception. You can change this, and other parameters using the `pitnc_getparams` and `pitnc_setparams` programs. The programs can be downloaded in a .zip file at:

<http://www.tnc-x.com/pitnc9K6params.zip>

You may need to change the permissions on these files in order to be able to access them. Use `chmod` to do this. To view the menu of parameters use the `pitnc_getparams` program:

```
./pitnc_getparams /dev/serial0 0
```

(if you are in the same directory as the program.) The baud rate is set using parameter 8. To change it to 1200 baud use the `pitnc_setparams` program:

```
./pitnc_setparams /dev/serial0 0 8 12
```

To change it back to 9600 baud:

```
./pitnc_setparams /dev/serial0 0 8 96
```

To make these changes effective you need to reboot the TNC. You can do this without removing power with the following command:

```
./pitnc_setparams /dev/serial0 0 15 2
```

If you have a radio that has one of the 6 pin mini-Din jacks on it the receive audio at 1200 baud is sent out a different pin on the jack than is receive audio at 9600 baud. So if you change the baud rate you must do two things. First change the baud rate setting in the menu on your radio to the proper baud rate. Second, change the jumper JP1 so that it is set for the proper baud rate. If 1200 baud, set it toward the center of the board; if 9600 set it toward the outer edge of the board.

The `pitnc_setparams` program also allows you to set the RX and TX level on the TNC. The optimal RX level at 9600 baud is almost always 0. This value allows the TNC to set its own RX level. The RX level is set with parameter 9. We generally find that an RX level of 50 is best for most radios when using 1200 baud. Here is a list of some actual voltages for both RX and TX for various parameter values:

RX/TX levels and line voltage:

255 = 3000mV P-P

192 = 2250mV P-P
160 = 2000mV P-P
128 = 1500mV P-P
64 = 750mV P-P
32 = 375mV P-P
16 = 188mV P-P
8 = 94mV P-P
4 = 47mV P-P
2 = 24mV P-P
0 = 0mV (auto-set level for RX only)

When I first ran the TNC at 9600 baud, I did so without changing either the TX or RX level. At that point the RX level was set to 0 and the TX level was set to 255. I was rather surprised to find that the TNC worked well with these values. It is likely that the optimal TX level is going to be somewhat lower than this however. The TX level is set using parameter 10:

```
./pitnc_setparams /dev/serial0 0 10 X
```

where X is the designed value. I have noticed that some radio's manuals (my FT-817, for example) have a recommended voltage level for signals to be transmitted at 9600 bps. My FT-817 suggests a level of 1.0 volts P-P maximum. That would correspond to a value of 96 for parameter 10. This might be a good starting point.

Do not change the I2C address value unless you are sure you know what you are doing. It should remain at 0 for serial port operation. If you set it to something else, it means you are going to communicate via the I2C bus and when you do this you will no longer be able to access via the serial port.

Reprogramming the Teeny's Firmware

In order to switch between the packet firmware and the ARDOP firmware, you will need to reprogram the Teensy board. Fortunately, this is not very difficult. You will need a USB cable with a micro USB connector on one end and a standard PC USB connector on the other end. This is the type of cable that comes with many cell phones (at least until type C USB connectors became common). However, not all of these cables will work. Some cables of this type only have wires that deliver power to the device, they don't actually transmit data. You will need a cable that transmits data as well. You will know you have the wrong cable if the Teensy won't program. This is what happened to me when I tried the first cable of this type I could get my hands on. I swapped this out for a different cable and that time it worked fine.

You will need to download two pieces of software in order to reprogram your Teensy. First, you need the Arduino IDE. You can find this at the following site:

<https://www.arduino.cc/en/Main/Software?>

Do not download the Windows app that appears on this page. Instead download either:

Windows Installer, for Windows XP and up OR

Windows ZIP file for non admin install

If you are using Linux or Mac OS X there is a version of this that you can download instead.

Download one or the other of these files and install them as you could any other program.

Then you will need the Teensyduino add on to allow the IDE that you just downloaded to work with the Teensy. You can do this here:

https://www.pjrc.com/teensy/td_download.html

This page contains links to the software and also has instructions on how to install it.

Next, download the TeensyProjects.zip file from www.tnc-x.com:

www.tnc-x.com/TeensyProjects.zip

Unzip this file and you will find the TeensyProjects folder. Move this folder so it is contained within your Arduino folder. If you accepted the defaults on installation, with Windows 10 it will be within your Documents folder.

Inside the TeensyProjects folder you will find the following two files:

boards.txt
platform.txt

Move these files to the avr folder that is in your Arduino program directory. If you didn't change the defaults and you installed this in Windows that will be in:

Program Files (x86)/Arduino/hardware/Arduino/avr

Now you can run the Arduino IDE program. Click on File/Preferences on the menu. This will open the Preferences dialog box and the first item will be "Sketchbook Location". Set this to the location where your TeensyProjects folder is. For example, mine was:

```
C:\Users\John\Documents\Arduino\TeensyProjects
```

Press OK to close the box. Now go to the Tools Menu and change the Board item to "Teensy 3.6".

Go to File/Open... and pick the .ino file you want to load into the Teensy. For example if you are loading the 1200/2400/9600 baud packet code, click on the SM-Teensy directory and select SM_Teensy.ino. If you are loading the ARDOP code go to the ARDOP_Teensy folder and select ARDOP_Teensy.ino

Plug your TNCPi9K6 board into a USB port on your computer using the cable described above. You should see lights flash on your TNCPi9K6 board. Then select from the Arduino IDE menu the Sketch item and select Verify/Compile. It will take a while for the firmware to compile, but then another small program will appear on your screen with directions for programming your Teensy. All you have to do is push the white button on the Teensy board. This is where you will find out if you have the right cable or not. If you see the progress bar indicating that the Teensy is programming you have the correct cable.

Support

If you have any questions about your TNC-Pi9K6 or are having hardware issues with it, please contact John Hansen, W2FS at john@coastalchip.com. Software issues, particularly with regard to the Linux version of BPQ are best addressed to the author, John Wiseman, G8BPQ.

There is a support group for the Raspberry Pi users in amateur radio that focuses mainly on using TNC's with the Pi. You can find it on groups.io:

<https://groups.io/search?q=RaspberryPi-4-HamRadio>