FT-891 Project Build

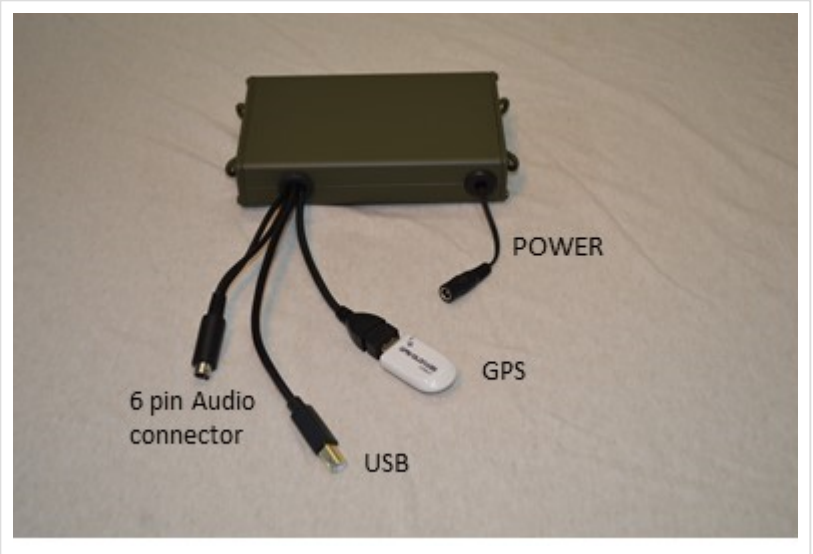
Posted on **April 29, 2020**

Write-up by: Bob Nazro Amateur Call W1RPQ

Building an FT-891 Field Rig with Raspberry PI computer.

I wanted to build a “Go Kit” for field activation using both HF voice and digital modes. I also wanted it to be compact and fit in a backpack with battery power or a small case. For the HF rig I selected the YAESU FT-891 radio for its size and quality. After watching some videos on field radio operations by Julian, OH8STN, I decided to add the Portable Zero rails and installed the Raspberry Pi4 inside the Portable Zero battery box. The first big step in building the Go Kit was to select the Raspberry PI board that I wanted as the computer. I chose a Raspberry PI 4 with 4Gig of Ram and picked a 32G microSD card to hold the programs. I set up a test station on the bench and began building the Pi software. You need to download the latest version of Raspbian software

and using a program called “balenaEtcher” burn the software image to the microSD card. This is kind of like loading Windows 10 on a computer. Once that is complete you insert the microSD card in the Pi 4 board and boot the system up. You need a monitor and keyboard plugged into the Pi for the initial steps. The software takes you thru steps to setup the screen, create a WIFI connection and configure interconnection protocols. Once the initial setup is complete you can then us VPN software to remotely connect to the Pi computer from your home computer. After running two scripts called “*sudo apt-get update*” and “*sudo apt-get upgrade*” the Pi should be at the very latest version and ready to load additional software. You will need to make 2 additional changes to the software. The first is to set the Pi to “auto log in” and the second is to set the screen resolution. In the Terminal window Run “*sudo raspi-config*” and select option 3 “Boot Options” then B4 “Desktop Autologin Desktop GUI, automatically logged in as ‘pi’ user” select <ok> This brings you back to the main page. Next select “7 Advanced Options Configure advanced settings” then select “A5 Resolution Set a specific screen resolution” scroll down and select “DMT Mode 82 1920×1080 60Hz 16:9”. Click <ok> click <ok> again and reboot. Backup the MicroSD card now. Any mistakes you make later, you can start at this point again, instead of building the microSD card and reconfiguring the Pi all over. I had seen several videos by Jason, KM4ACK, he has a Youtube channel full of Ham Radio related information. I spent several days building the basic computer, one program at a time and troubleshooting issues with the LINUX code. I then found a new video that Jason had put together called “Pi Build V2”. This was a single script that you ran on the Pi after building it, “*git clone https://github.com/km4ack/pi-build.git $HOME/pi-build && bash $HOME/pi-build/build-a-pi*” This downloaded and installed all of the programs that I wanted on the Pi. After a few emails back and forth with Jason, I was able to get everything configured correctly. The PI interfaces to the FT-891 with a USB cable for Rig control and a 6 pin DIN cable for the audio sound card feed. I chose a SABRENT USB Audio Sound Adapter for the audio feed between the PI and the radio. This sound card does not require any drivers. I also added a USB GPS dongle for position information and a constant clock signal.

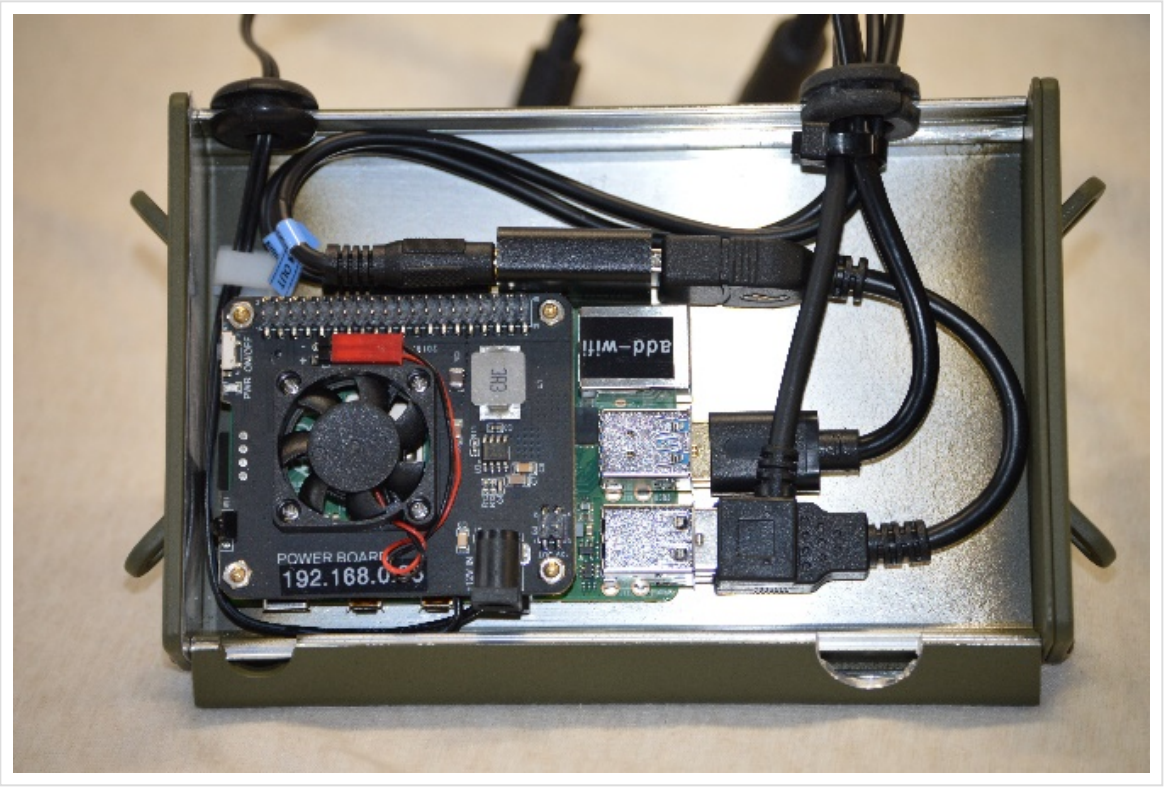


After testing the functionality on the bench, I started on packaging everything into the battery box. I laid out and drilled the mounting holes for the raspberry PI board and added a second hole to the side of the case for the data cabled to exit from. I used a step drill and a drill press for this effort.

The Pi board mounted inside the case.

I then ordered some 6” USB extension cables and a 1’ USB cable to reach the back of the radio. I wanted the rig to run off of 12Vdc and not the normal 5Vdc that the Pi board required, and I found

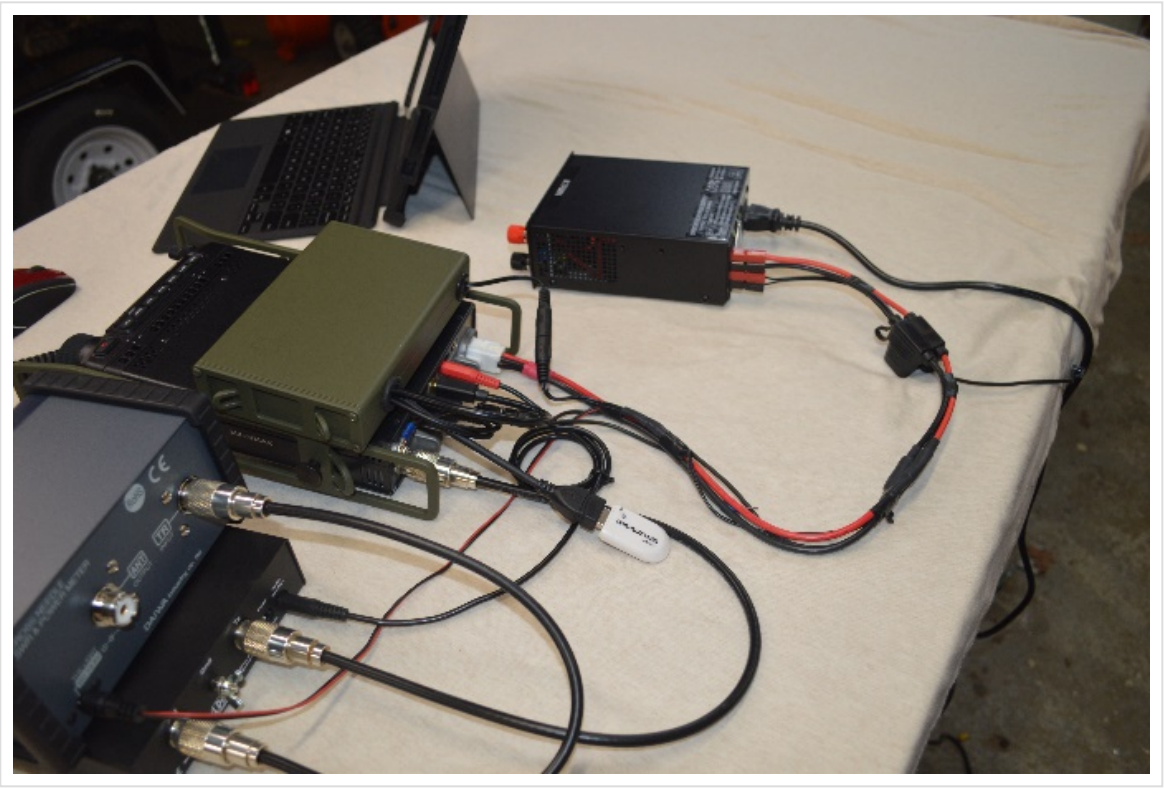
a “DockerPi” Power Board that ran off of 12Vdc and provided a fan, 5 Vdc and a remote on/off switch. All this fit compactly into the battery case.



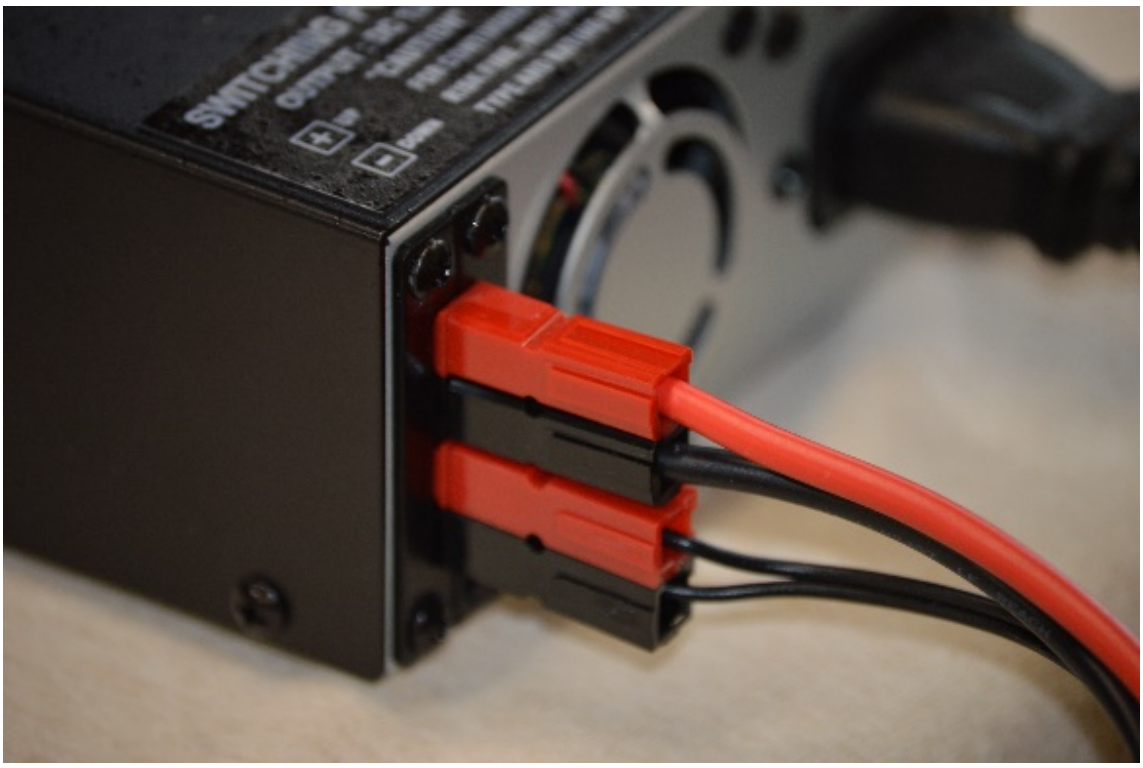
The case is made of aluminum and I needed to replace one of the end-covers with a clear plate to be able to see the LEDs on the PI board and be able to use the IR remote control. I used a plastic sheet and traced out the metal panel and cut it with some scissors.

The box sits on top of the FT-891 and the connections to the radio are simple with the short cables.

I built a power connection cable to meet the radio, PI board and SWR meter requirements. I also included a ground wire that can connect to each component and then to a separate ground rod.



I selected an MFJ 4230MVP 20-amp power supply due to the compact size and the power pole connectors on the back for use with a generator.



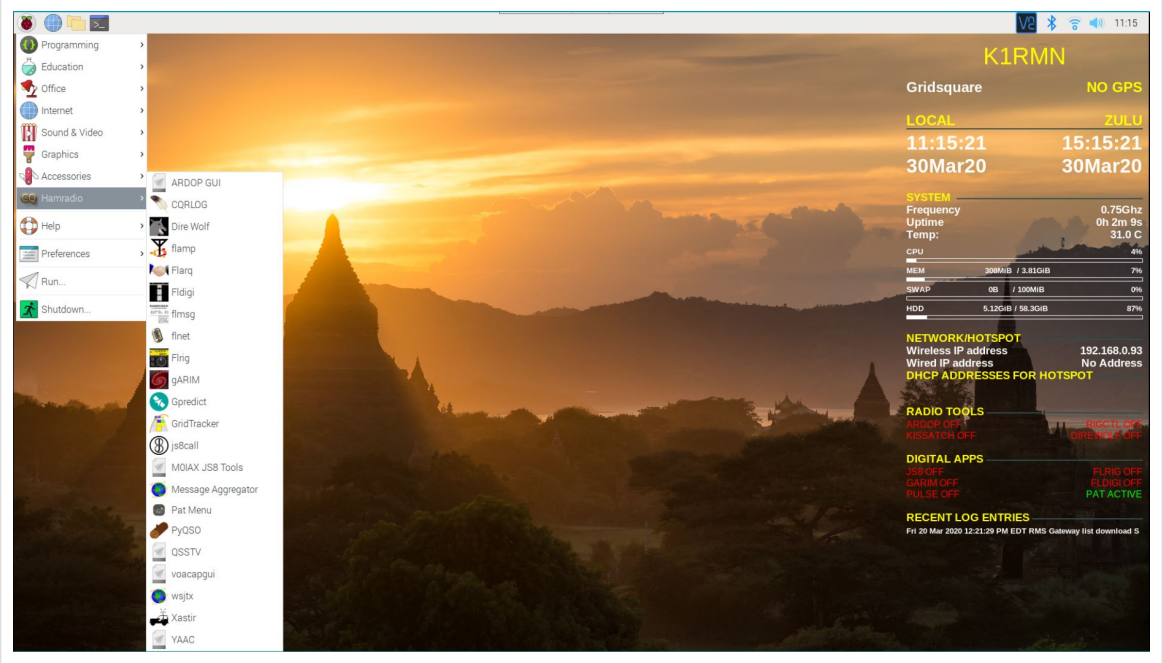
The face of the FT-891 is easy to read and the Portable Zero rails protect the front knobs, both in the case and in the field. The Raspberry Pi case nests on top of the radio and keeps all the connections simple, short and out of the way.

The final setup is compact and fits in an APACHE 4800 case.

The final setup connected and ready to operate.

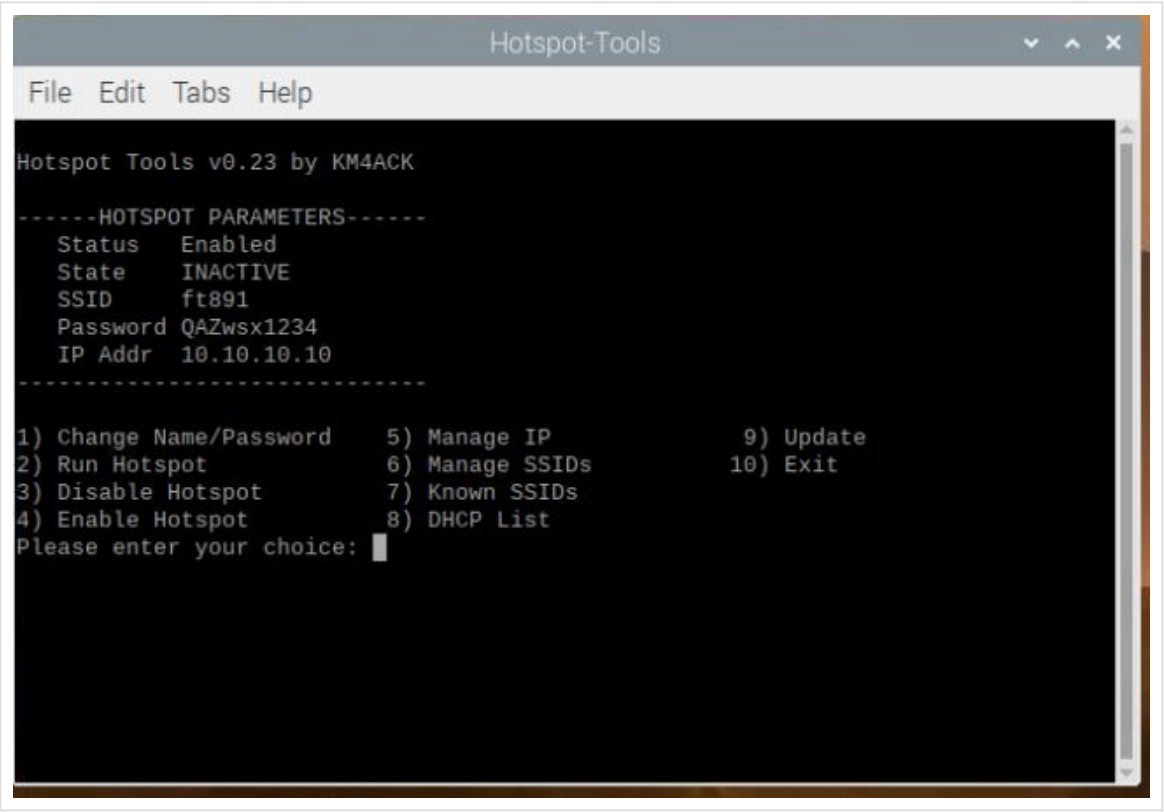
I will be using a BuddyPole antenna system, in the field and chose the LDG Z-100Plus Auto tuner to help with the SWR matching. To control the software, I am using a VPN connection on a DELL Tablet to control the Raspberry Pi untethered. You could use a laptop, CELL phone or IPAD, as well.

Let me cover some information on how to operate the PI in the field and a bit about the Amateur Radio software loaded on the Pi.

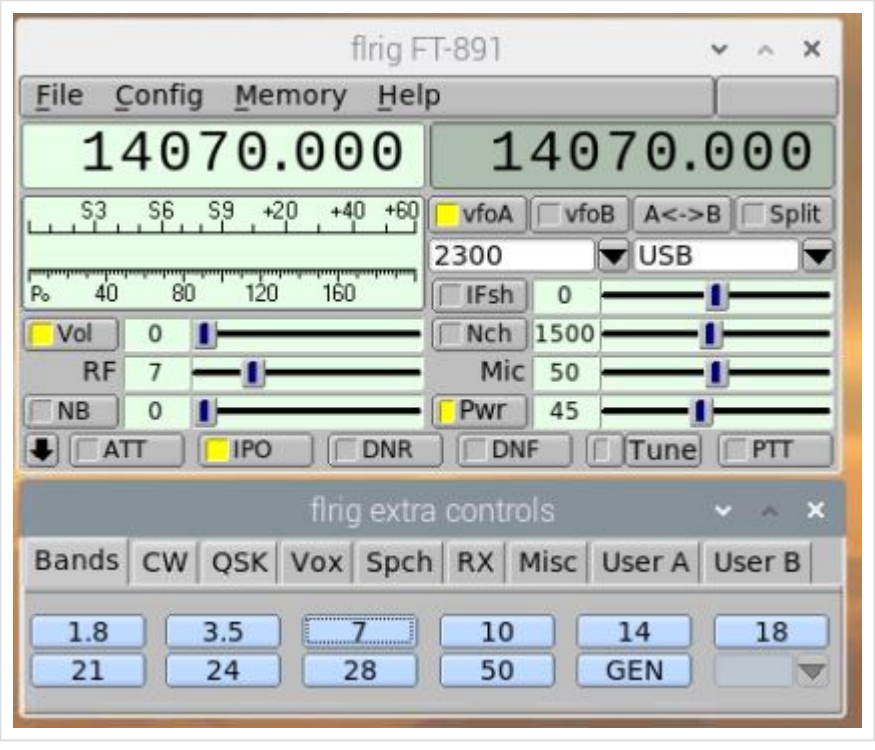


Once you have everything connected and have a 12Vdc battery or generator running, turn on the FT-891 rig and start the raspberry PI computer with the IR remote. Once the blue light on the DockerPi power board is soliod blue, open the VNC connection to the raspberry Pi and you are now ready to operate the portable station. On the right side of the main menu is a dashboard that shows Date/Time, System information, Network connections and what radio tools are running. The left side is the menu drop down list. The Raspberry Pi 4 is a very powerful computer with dual monitor capability, sound output and, with WIFI connected, the Chrome browser will work as well as any other computer. LYNIX is a very different software language, but there are several videos and books on the market that can help you figure out how to use it.

WIFI Connections. When you build the Pi with Jason's software the first part of the setup is about the WIFI connections. The PI will connect to a known WIFI source or start its own hotspot, if it cannot locate a source. This is how you can connect a laptop or tablet in the field to the Pi computer. The internal hotspot IP address is 10.10.10.10. On your tablet or laptop device open WIFI connections and look for the Pi Hotspot, click on it and enter the password you chose when building the Pi. You can now use VNC to connect to the Pi and have a dashboard in the field without internet connection. I installed Jason's "Hotspot Tools" and this allows you to control the WIFI portion of the Raspberry Pi. This tool will also allow you to connect to new WIFI hotspots on the go.



I use FLRIG to control the radio. All the other software programs can connect via FLRIG to get to the radio. This is a simple way to connect your radio to all the software apps you intend to use.



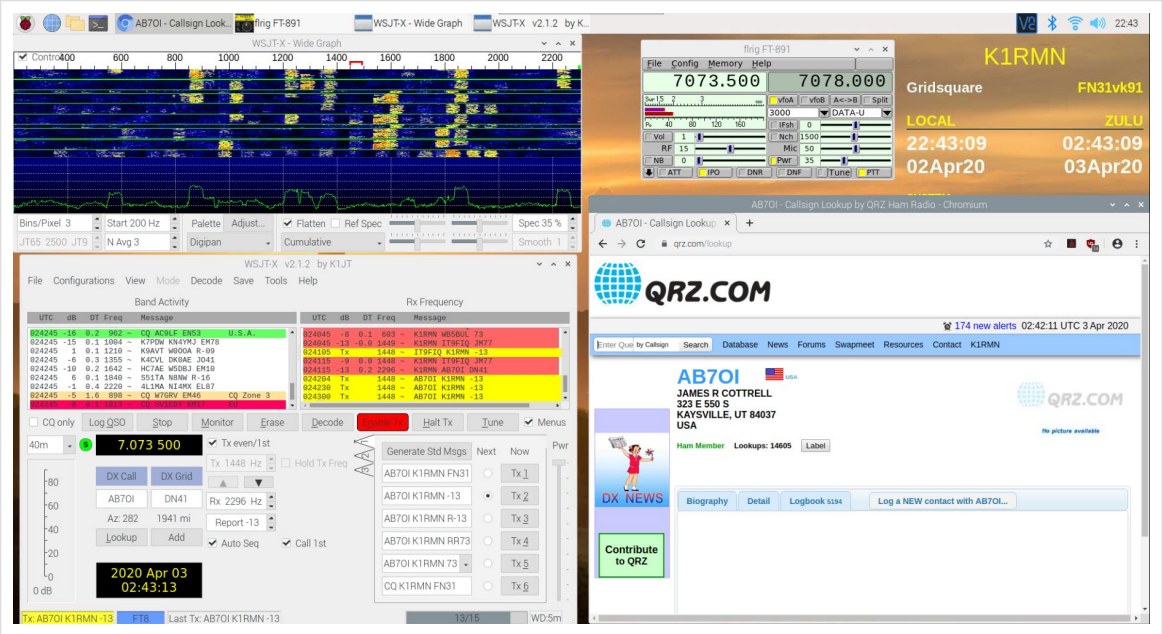
I will mostly be using FLDIGI, WSJT-X and js8call, along with one of the logging programs for field work. It is easy to transfer an ADIF file from the Pi to my home computer and upload to my main logging software. Several popular Amateur Radio Applications have been converted to LYNIX and

are available for use on a Raspberry Pi. Updating individual programs is similar to a Windows system program update. Select the software update, download it to the Pi and run the executable file. There is also a simple script that you can run to update all of the programs on the Pi. Open a Terminal window abd enter "*Cd Downloads/"* to change to the Downloads directory and then enter:

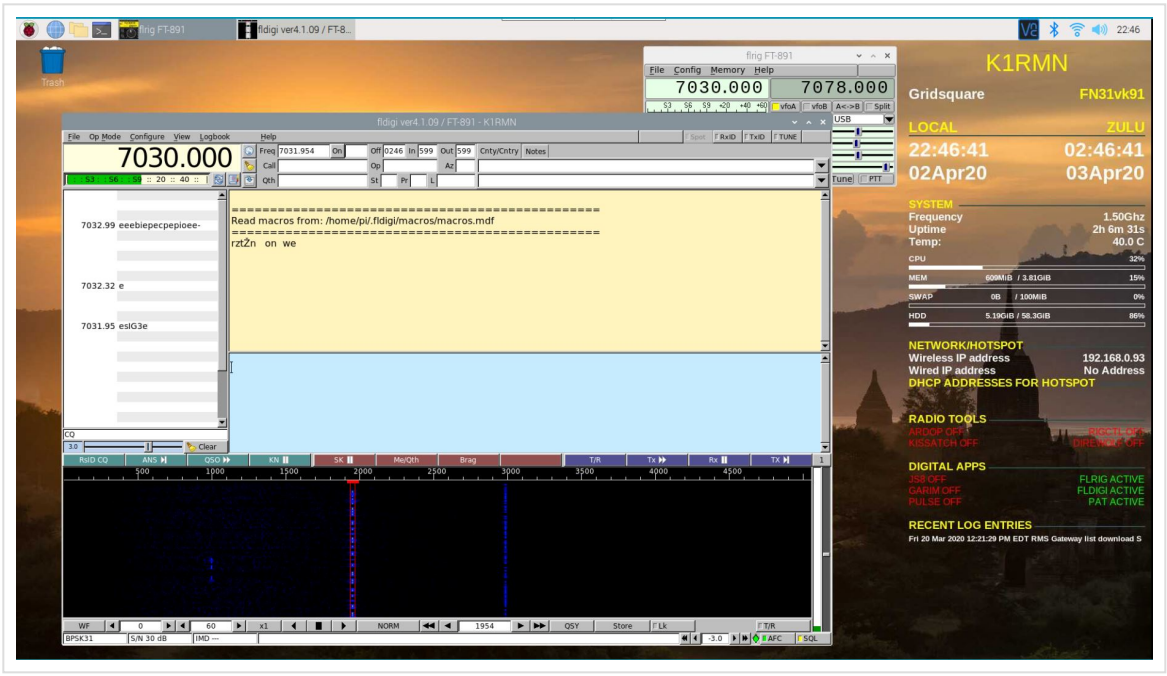
"wget https://raw/githubusercontent.com/km4ack/pi-build/dev/update-bap".

A key point with the setup for these digital programs is ensuring that the Audio inputs and outputs are set to "USB Audio Device. Click on the speaker in the upper right-hand corner and verify that is the selection. Under input and output device settings click the options tab and insure that the "Auto Gain Control" box is unchecked. On the Capture Tab check the enable box and set the volume control to the midpoint. After checking these settings Left click on the speaker and insure the volume control is up about 80%. If you are transmitting and see a low power out on your SWR meter, check this volume setting first. I have had this too low and thought there was something wrong with the radio and software only to find the volume was just too low on the Raspberry Pi.

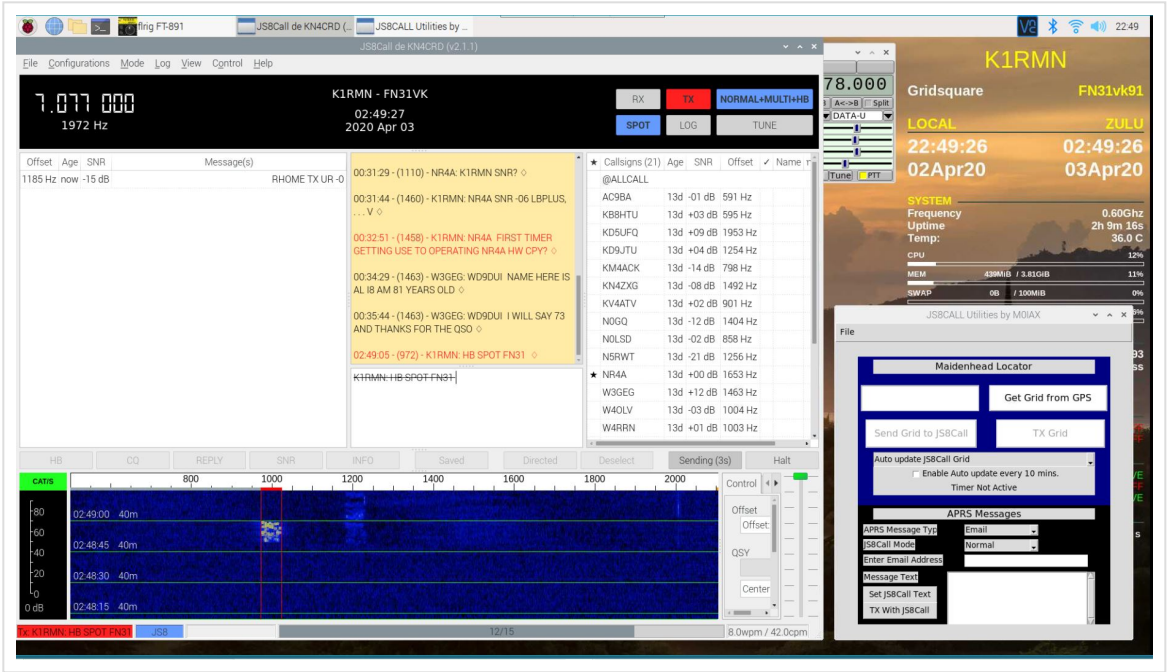
I use WSJT-X for FT-4 and FT-8 contacts. I also open a browser to allow me to use QRX.com, if I have internet access. For its size and no additional software required, the Sabrent USB Sound Adapter works well and the WSJT-X waterfall is clear and easy to adjust.



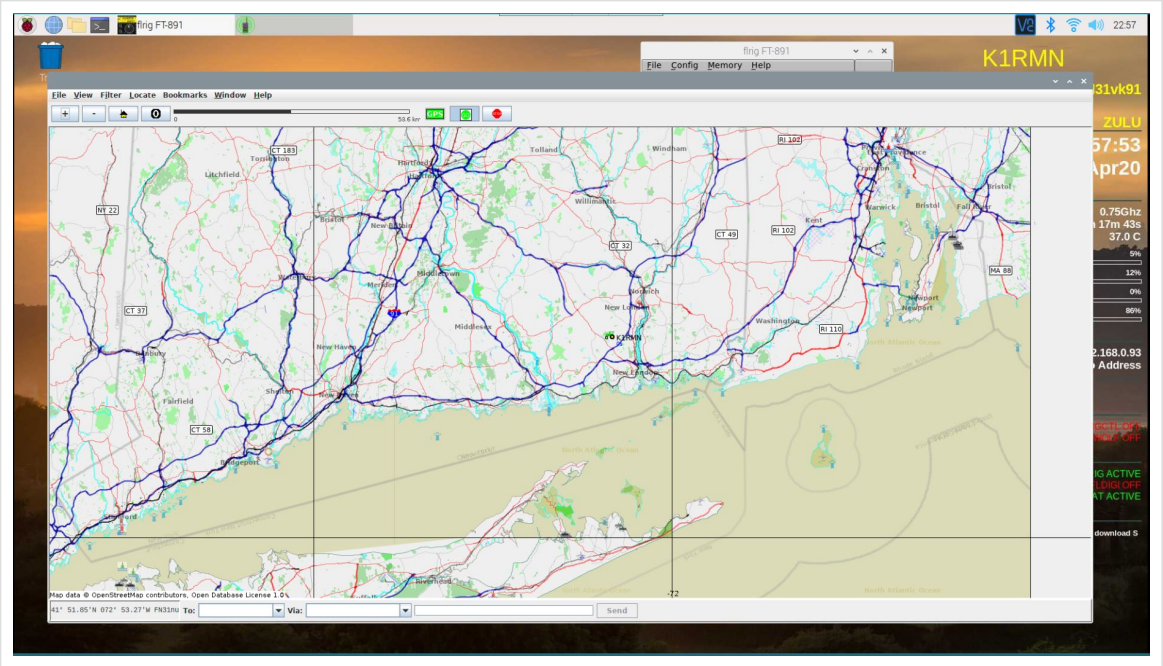
I use FLDIGI for PSK-31, providing a text method of Digital communications. The same audio volume issue should be the first thing to check if you are experiencing any issues with your signal.



Js8call is a new form of digital text communications. I don't have a lot of experience with this mode, but as I am learning and making contacts right out of the box.



YACC is one of the GPS programs that I loaded onto the Pi and it seems to provide good maps, with a simple setup procedure. GPS is an area that I am just getting into and I have a lot to learn.



The end result is a portable Ham Shack with lots of Digital modes that can be run in the field at a SOTA, POTA, Field Day or any remote location, complete with logging programs and GPS data to track satellites and maintain that constant clock required for digital communications. Logging software is a personal preference and there are a few available for use on the Raspberry Pi. Transferring ADIF logs from the Pi to your home computer is quite easy with the use of a free program called “WinSTP”. This program uses the SSH communications protocol to transfer files back and forth between the Pi and your computer. Then it’s a simple process of uploading the ADIF file into your primary Logging program.

All the parts to build the portable Raspberry PI were available online and were delivered in a few days. The only major tool required was the Drill Press, which made drilling the case much easier. My next addition to the Go Kit will be portable battery power. The Appendix contains a list of all the parts and software that I used to build the kit, along with some helpful LYNIX commands to setup the PI, check the GPS and update the Pi software.

Testing the FT-891 Field Kit outdoors on 40 Meters with a BuddyPole antenna. I made several 40 Meter contacts in about an hour while adjusting the antenna for optimal performance. Now I’m ready to go out and activate some parks and lighthouses or operate remotely for Field Day or just take the kit with me as I travel around the country.

Bob Nazro, K1RMN

Acknowledgements: I want to thank Jason, KM4MAC and Julian, OH8STN for their ideas and assistance during this project. Jason has a YouTube site full of information of all sorts of Ham Radio projects and Jilian is an expert in Off Grid Amateur Radio setups. The best video I found for building the basic Raspberry Pi 4 system was from Crosstalk Solutions; Chris does a great job of walking you thru all the steps to download the Raspbian image, burn it to the microSim card and configure the Pi software for the basic system.

Jason's YouTube channel: https://www.youtube.com/channel/UCSQhXfGo_68Ta8-2wStAWkw

Julian's YouTube channel: https://www.youtube.com/user/SurvivalTechEU

Chris's YouTube channel: https://www.youtube.com/watch?v=BpJCAafw2qE

APENDIX:

Equipment List:

Yaesu FT-891 HR Radio

LDG Z-100Plus Autotuner

MFJ 4230MVP 20 amp power supply

4GB-9004	Raspberry Pi 4 Model B/4GB

Portable Zero 891-EG 891 Escort Rails

Portable Zero PFZ-FP3-B Battery Case

MakerFocus Raspberry Pi 4 Power Expansion Board DockerPi Power Board 20W (5V 4A)

Oxsubor SuperSpeed USB 3.0 Male to Female Extension Data Cable Right Angle (20CM,8IN)

1FT – FT8/FT4/JT65/PSK Audio Interface 6-Pin Mini DIN Cable DC-42F for YAESU Printer Cable 1Ft, USB Printer Cable USB 2.0 Type A Male to B Male

Sabrent Aluminum USB External Stereo Sound Adapter for Windows and Mac. Plug and Play No Drivers Needed [Black] (AU-EMCB)

Gmouse G-Mouse USB GPS Dongle

Apache 4800 Weatherproof Protective Case

Software Programs in the Pi build:

HOTSPOT: Hotspot is used to generate a *WIFI* hotspot that you can connect to with other wireless devices. This is useful when in the field so you can connect to and cont*r*ol the pi from a wireless device.

GPS: This will install the needed utilities to get a GPS device configured as a time source on the pi. Helpful when you are not connected to the internet since the pi doesn't have a real time clock and therefore will NOT keep accurate time when powered off. Confirmed to work with this GPS https://amzn.to/2R9Muup Other GPS units may work but have not been tested.

FLRIG: Rig *C*ont*r*ol graphical interface. http://www.w1hkj.com/

FLDIGI: Digital Mode Software http://www.w1hkj.com/

FLMSG: Forms manager for FLDIGI http://www.w1hkj.com/

FLAMP: Amateur Multicast Protocol – file transfer program http://www.w1hkj.com/

PAT: Winlink client for Raspberry Pi https://getpat.io/

ARDOPC: HF modem for Pat. Recommended if installing Pat https://www.cantab.net/users/john.wiseman/Documents/ARDOPC.html

ARDOPGUI: GUI interface for ARDOPC. Recommended if installing Pat

PATMENU: Menu for configuring Pat. Recommended if installing Pat. https://github.com/km4ack/patmenu

DIREWOLF: Software TNC. In this setup, direwolf is used for 2M packet connection with Pat and can be used for APRS connection with Xastir. *Recommended* if installing P*AT* or Xastir https://github.com/wb2osz/direwolf/tree/master/doc

AX25: AX25 tools for Direwolf & Pat. Recommended if installing Pat.

HAMLIB: Rig *Cont*rol software. https://sourceforge.net/projects/hamlib/

PULSE: Pulse audio. Provides a way to configure virtual sound cards. REQUIRED for AMRRON ops.

JS8: JS8Call digital software https://js8call.com

M0IAX: Tools for working with JS8Call. Recommended if installing JS8Call https://github.com/m0iax/

WSJTX: FT8 & WISPR software suite. https://sourceforge.net/projects/wsjt/

XASTIR: GUI interface useful when configuring APRS nodes. https://sourceforge.net/projects/xastir/

YAAC: Yet Another APRS Client GUI interface useful when configuring APRS nodes. https://www.ka2ddo.org/ka2ddo/YAAC.html

PYQSO: Logging software https://github.com/ctjacobs/pyqso

GPREDICT: Satellite Tracking http://gpredict.oz9aec.net/

CQRLOG: Loggin*g* Software https://www.cqrlog.com/

QSSTV: Slow scan tv http://users.telenet.be/on4qz/qsstv/index.html

Gridtracker https://tagloomis.com/

Propagation (VOACAP) Propagation Prediction Software https://www.qsl.net/hz1jw/voacapl/index.html

PI scripts that can be helpful: These are all run in the terminal window

sudo raspi-config

Build a Pi script:

git clone https://github.com/km4ack/pi-build.git $HOME/pi-build && bash $HOME/pi-build/build-a-pi

Update Scripts:

Cd Downloads/

wget https://raw.githubusercontent.com/km4ack/pi-build/dev/update-bap

Standard update and upgrade commands

sudo apt-get update

sudo apt-get upgrade

To check GPS:

systemctl is-active gpsd

systemctl is-active chronyd

To see GPS data

gpsmon -n

Other programs on the PI for GPS:

cgps -s

gpsmon -n

xgps

POWER HAT setup:

Wget -qO-http://git.io/fj3b9 | sudo bash

To check the status of the wireless connection you are using (to see if wlan0 has acquired an IP address) : *ifconfig*

To check which network the wireless adapter is using: *iwconfig*

Lists USB hardware connected to your Raspberry Pi: *lsusb*

This entry was posted in **Projects** by **Rob Sawyer**. Bookmark the **permalink [https://tricityarc.net/ft-891-project-build/]** .