

A Pi SDR Standalone Rig

Pi SDR IQ Plus with Quisk, WSJT-X, FLDigi for with Softrock/Ensemble or RS-HFIQ rigs

The new firmware for Pi SDR that gives standalone operation, is including already installed and configured the Quisk, WSJT-X, Fldigi, Hamlib applications and Virtual audio streams.

It can be downloaded in order to be flashed to the microSD from this link

-> <https://mega.nz/#!NHBCRAQA!IHf6jtemtVdGqQr9txQ-Raeerpu1I1b0eFOEF5RgSY>

The name of downloaded file is "pisdr_standalone_v1_1_lan.img.gz"

The cable connections are according to Hardware Setup Guides, yet without the PC.

Instead, there is need for a HDMI monitor and mouse/keyboard (I use a wireless dongle having both in combo a keyboard and a mouse from Logitech). The HDMI monitor is connected at the side of Pi SDR IQ Plus where is an HDMI out connector, also the keyboard and mouse can be connected to any of the front USB Host connectors.

The included applications, Quisk/WSJT-X/Fldigi are configured up to the point where they can operate together, eg their audio streams and CAT, though other configurations and calibrations should be performed by the operator, eg frequency/level/Rx image/Tx image/user callsign/locator setup for digimode apps etc.

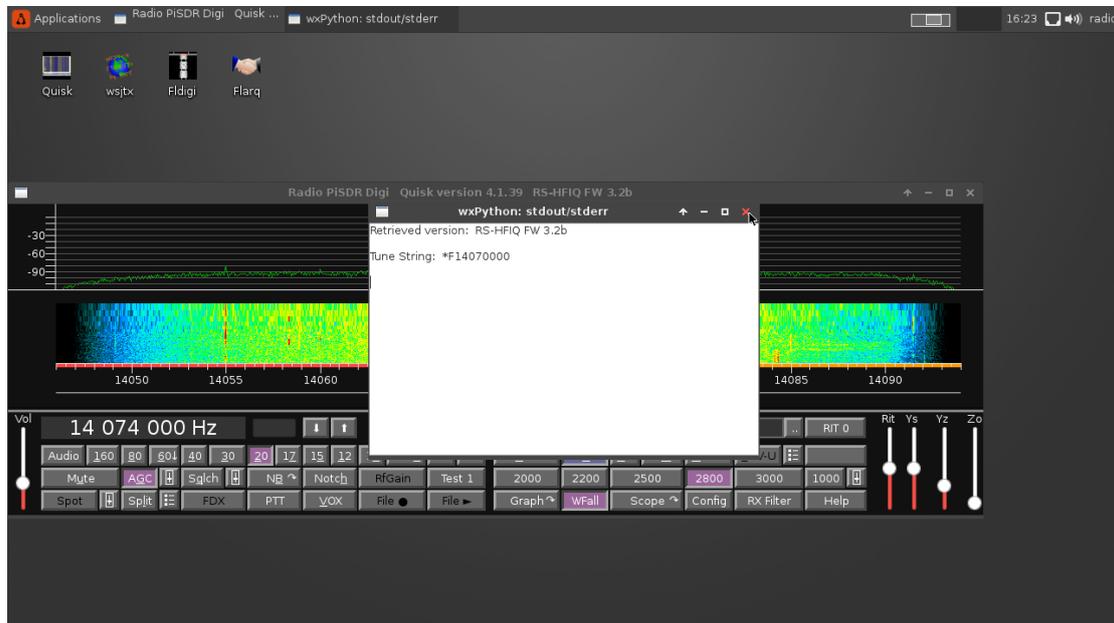
At first, upon startup, the firmware image is configured to have 'Blank' Quisk setup so user all that has to do is to select what radio is using by executing (double-click) the relevant desktop provided scripts, these scripts perform automatically the configuration required.

Each provided rig Quisk setup has a dual configuration installed, one for digimodes as default and one for phone operation, user can select ad-hoc each one that wishes to operate exclusively.

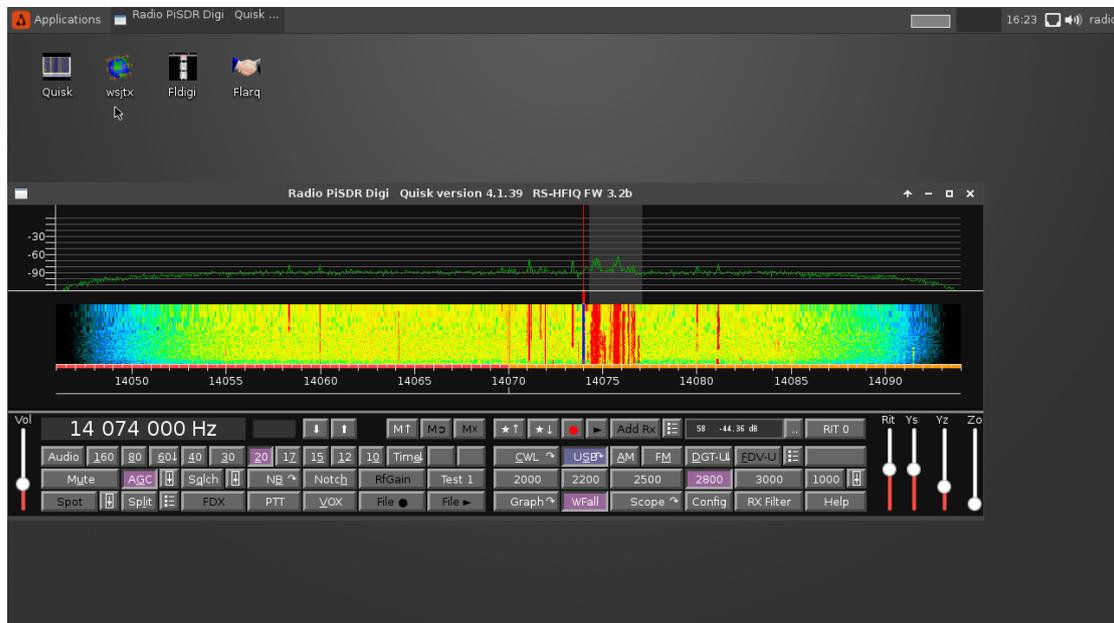


That is the first screen that should be shown after about 30sec from power on.

At that point, by double clicking on the Quisk desktop icon, the application will startup, as mentioned in digimode configuration so no sound will be heard.

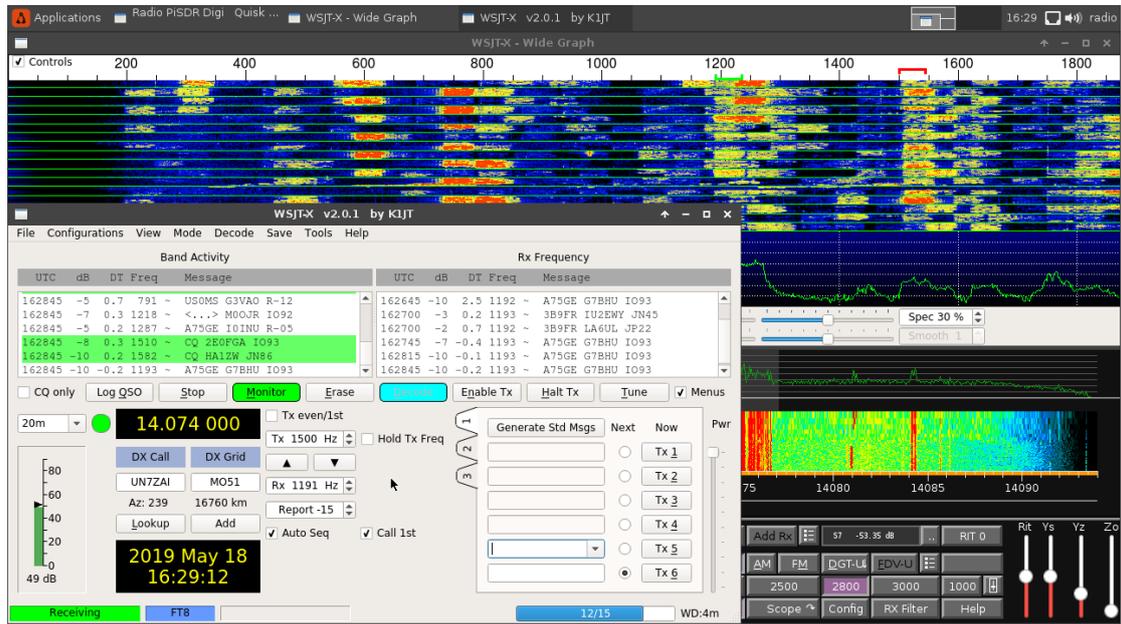


User should minimize or close the stdout/stderr python window upon popup. It is probably from the RS-HFIQ python code that is specific for rig comms.



At that point, user can double click the WSJT-X desktop item and in a few seconds the application will start.

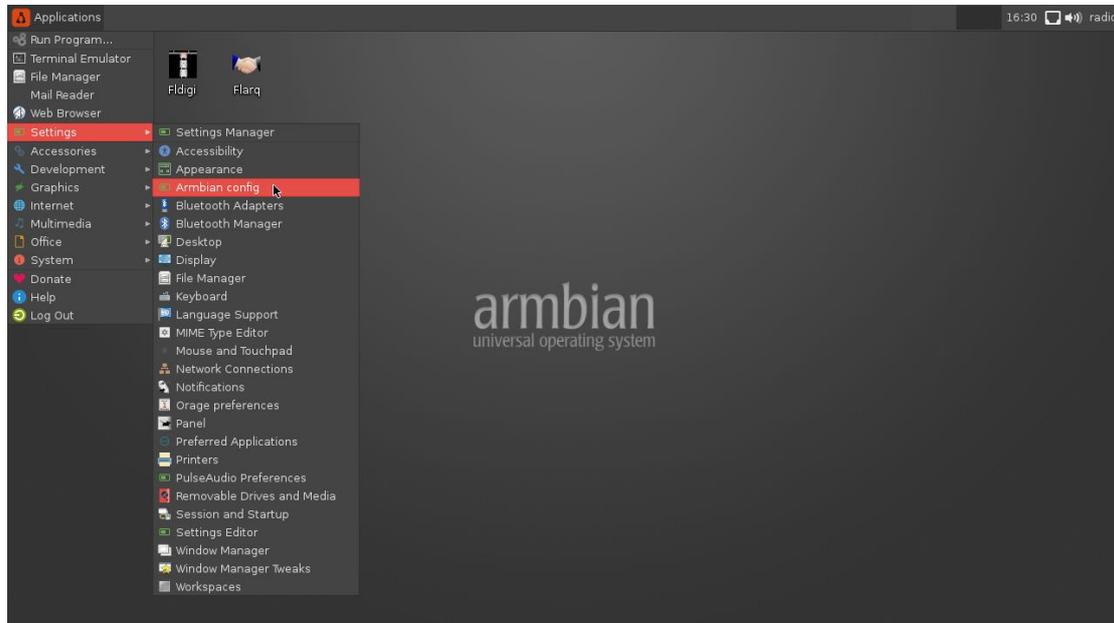
Since CAT and audio streams are already configured, after WSJT-X starts, user should select the band and decoded signals should show up after a while. Pay attention to the mode (LSB/USB) in Quisk and change it from there if required.



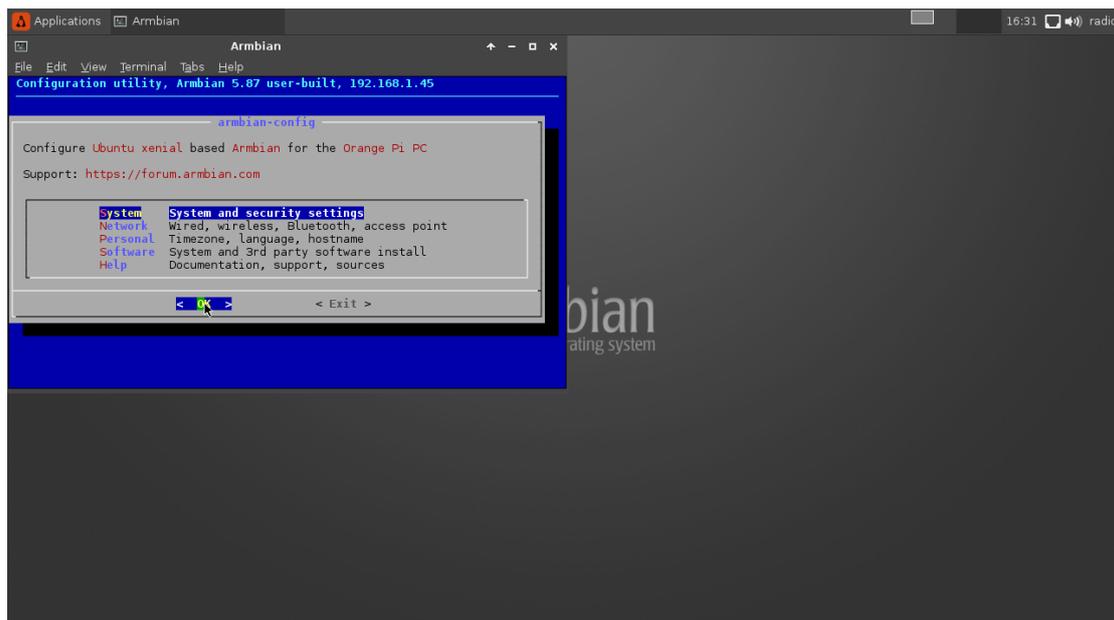
As it is firstly given, the firmware image has a relatively small screen size, a 720p resolution at 60Hz, “720p60”.

If using a larger monitor, user can change the screen resolution in the following way through the menu.

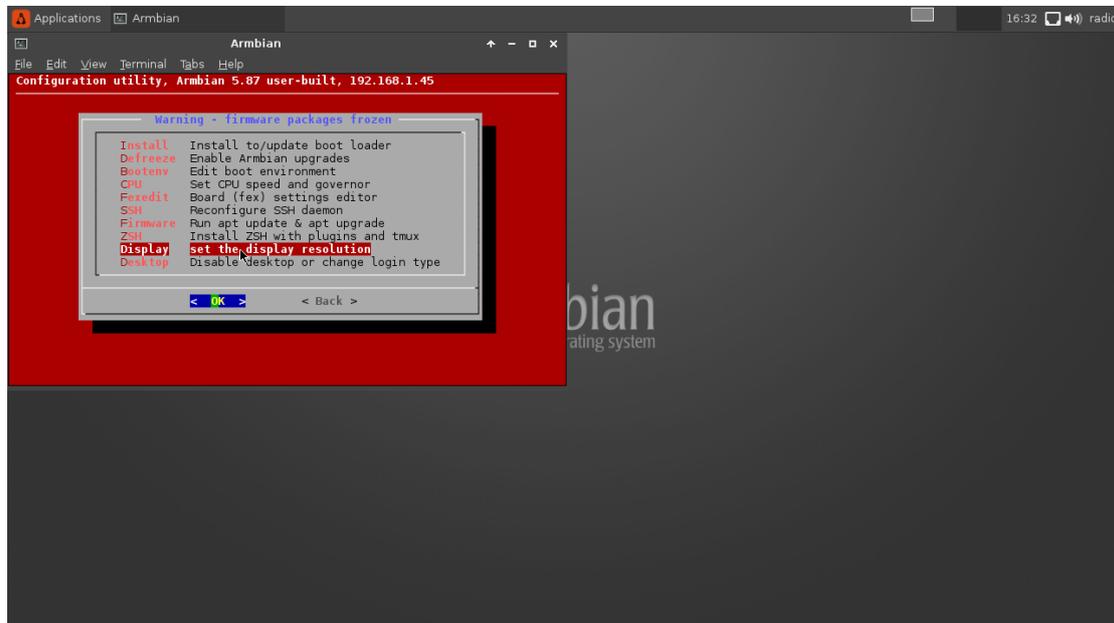
->Applications->Settings->Armbian config



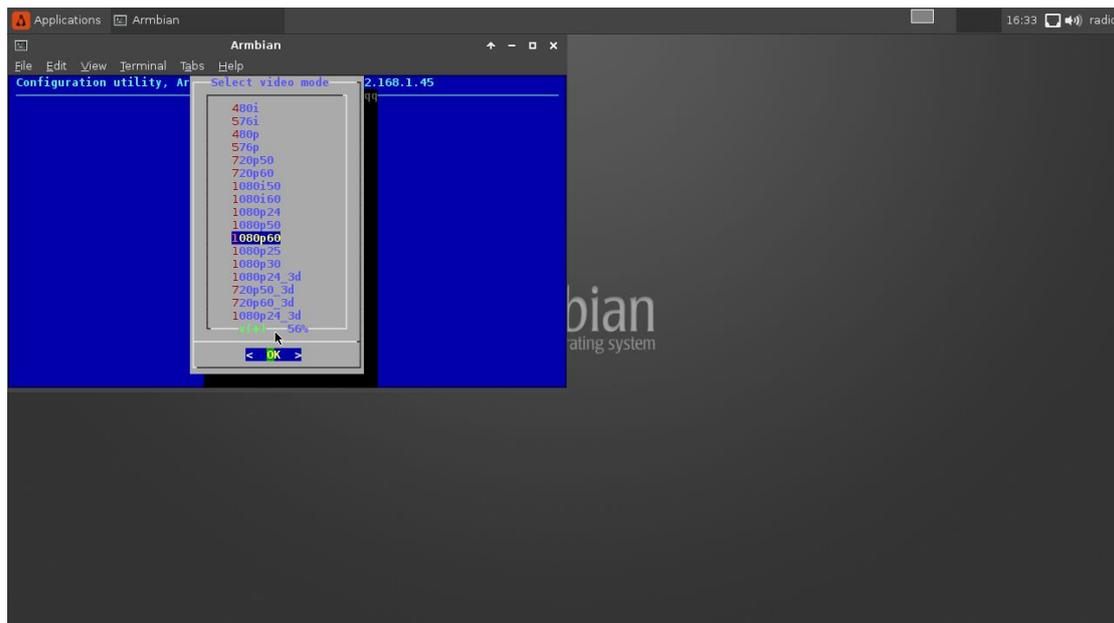
Navigation in this utility is with arrow keys, space and enter, first select the “System” settings with arrow keys and press enter



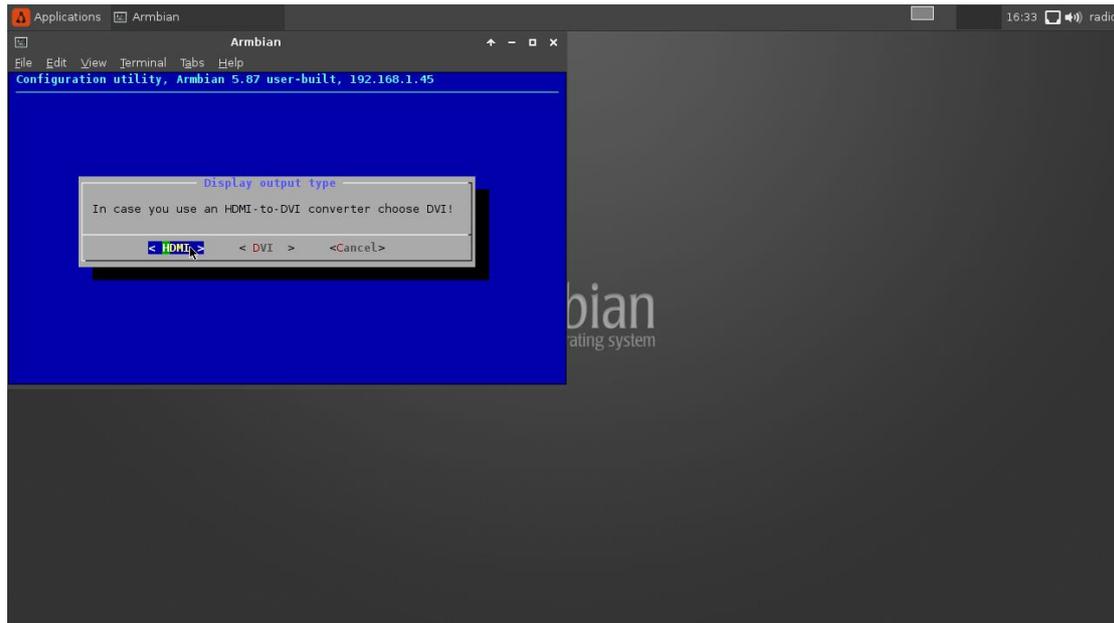
Then in the following screen select the "Display" and press enter



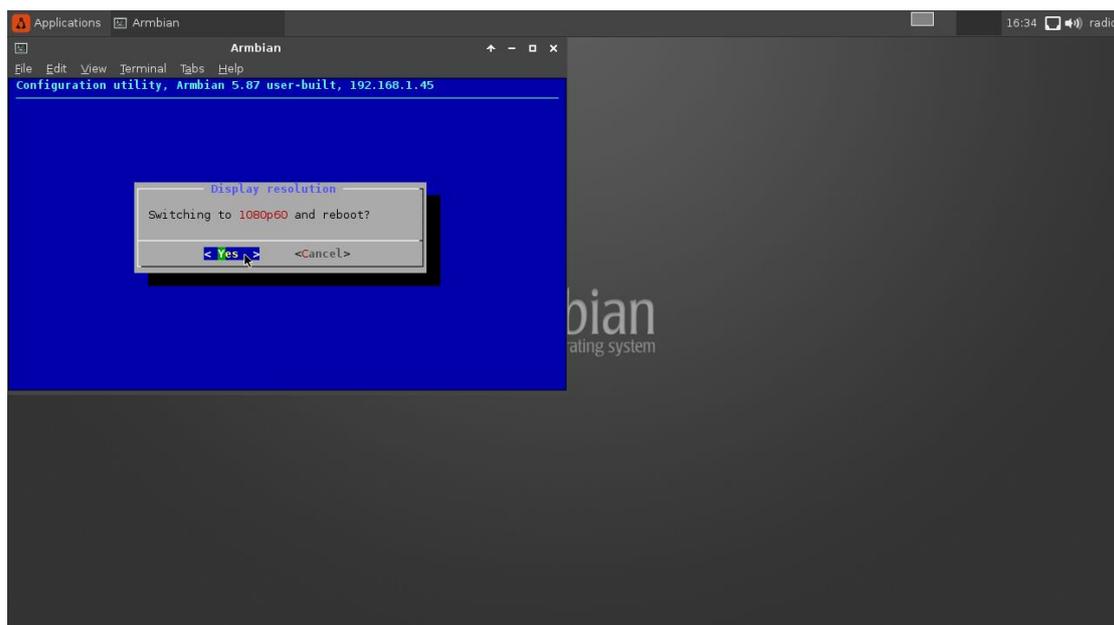
In this list select the resolution type of the attached HDMI monitor (as shown a new one 1080p60 is selected) and press enter



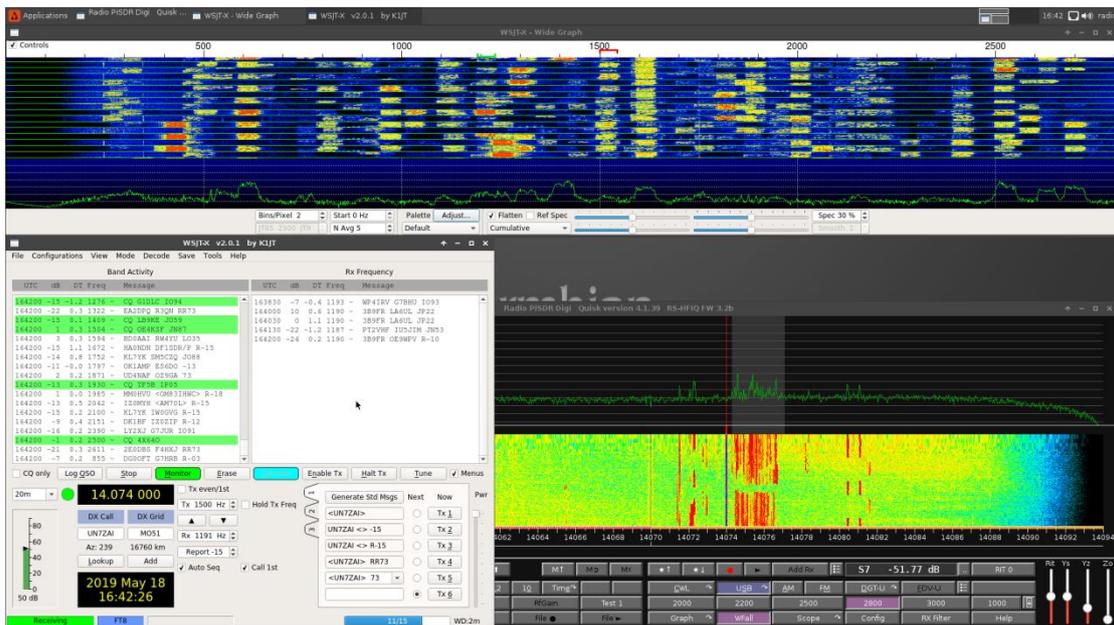
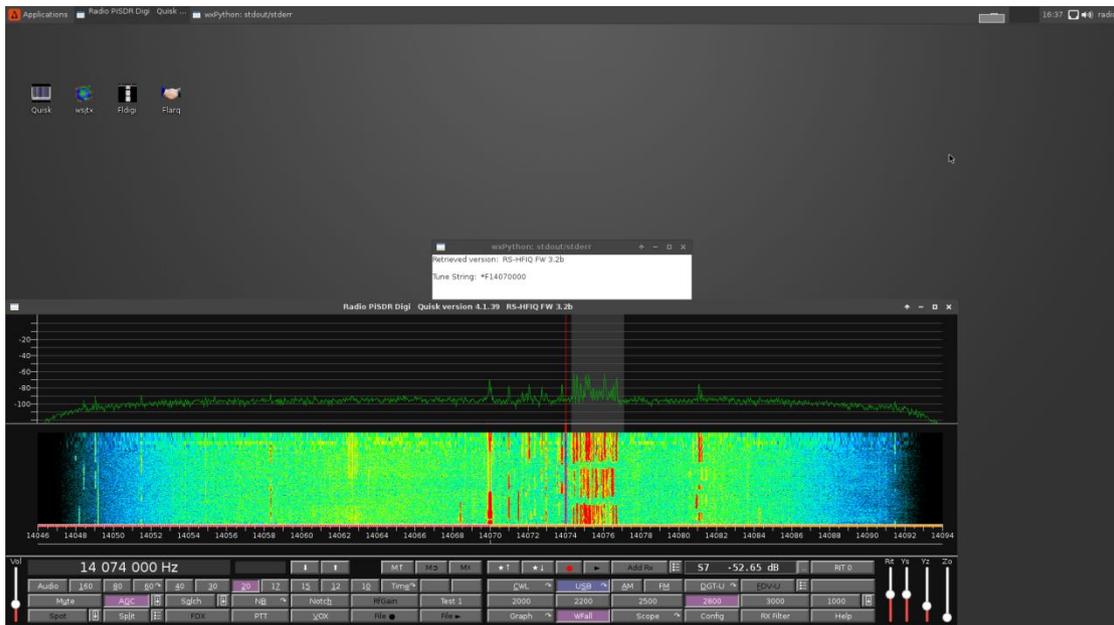
Enter again if you have a plain HDMI cable.



And the final confirmation since after that a system reboot will happen.



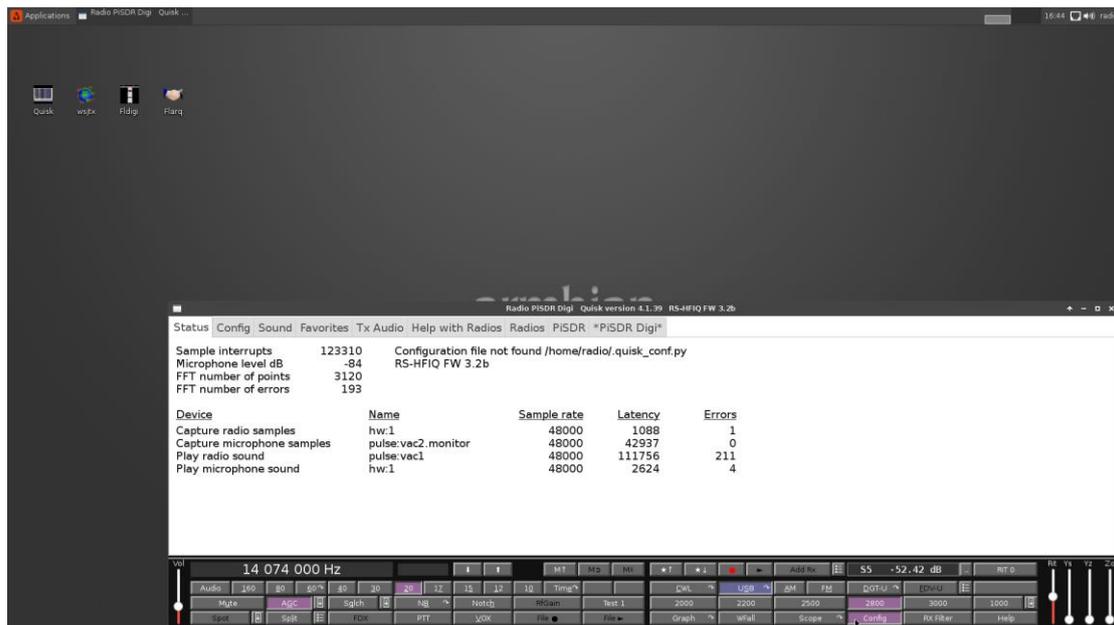
Upon the next start and by clicking the Quisk icon, the new screen size is applied



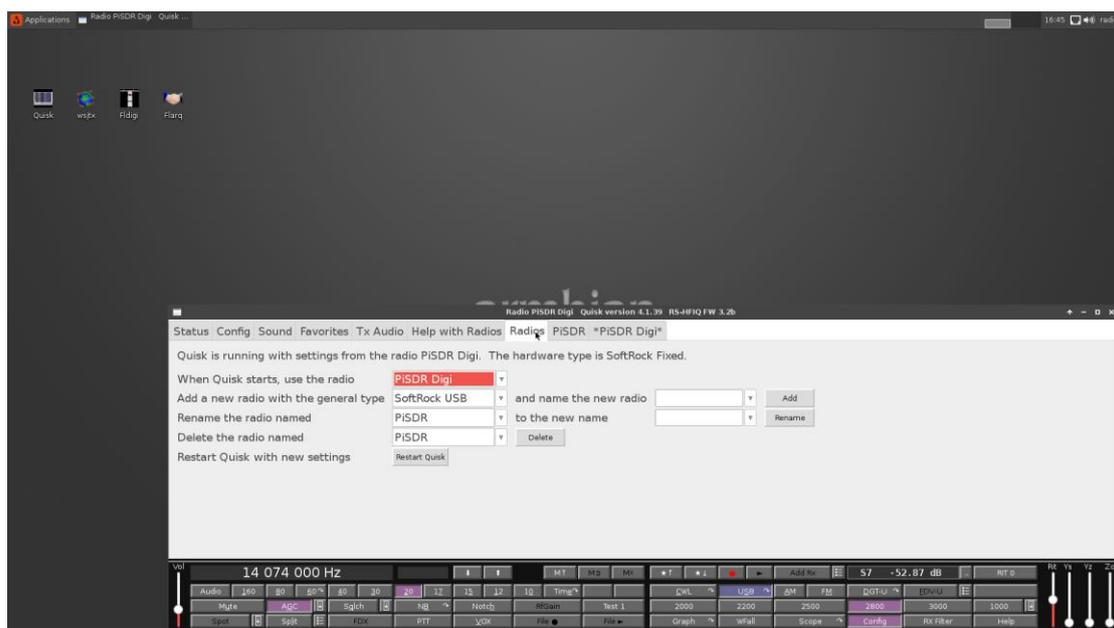
As mentioned, there are two Quisk configurations, a phone and a digimode. Upon firmware start the digimode is the active one, user can change between them and decide which one will be the active for the next boot.

Also if phone operation is required, it is needed a third-party USB dongle with headphones/mic that should be connected to one of Pi SDR IQ Plus USB ports, since the on-board mic jack cannot be used in this configuration. If a user do not use Tx in phone modes (Rx only) and not having a USB audio dongle, it can place in Quisk configuration "Radio Audio Output" to "hw.0", thus the onboard speaker output will be used.

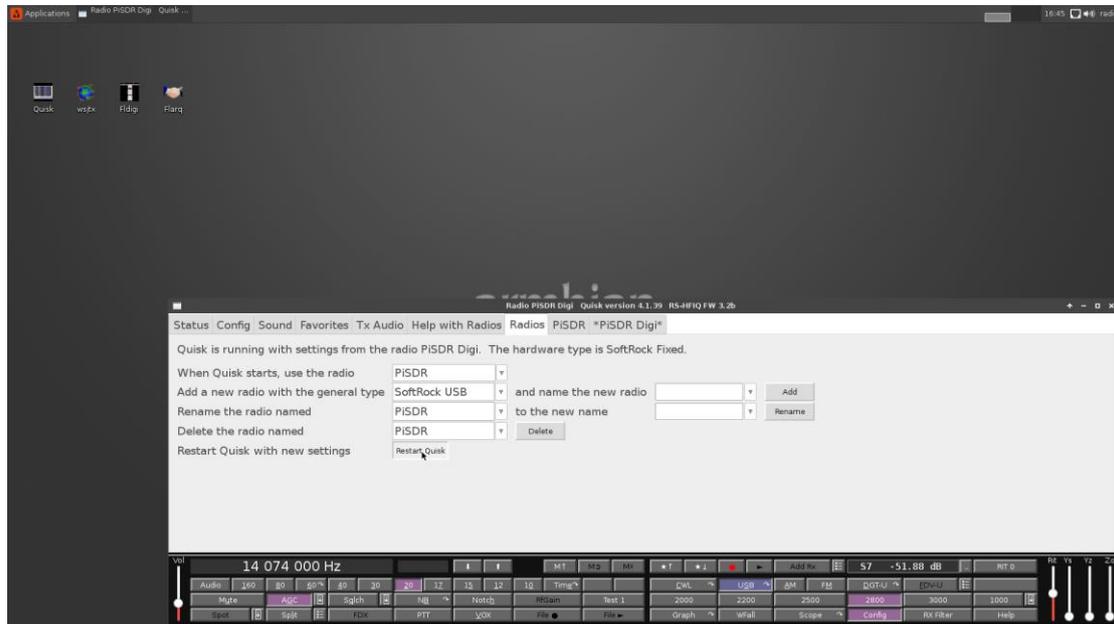
For this, user should click on Quisk's "Config" button



And then on "Radios" where it is shown that PiSDR Digi is the active one

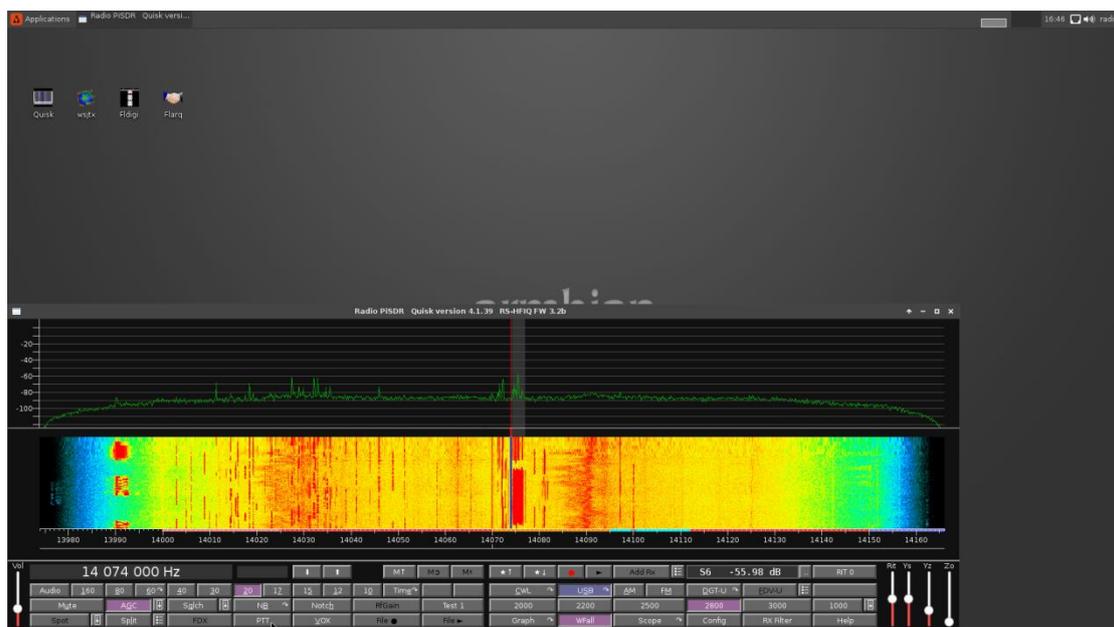


There user can select that the “PiSDR” configuration should be active next on instead of the PiSDR Digi and then click on the “Restart Quisk” button. Quisk will close and when it comes up again it will be in phone configuration.

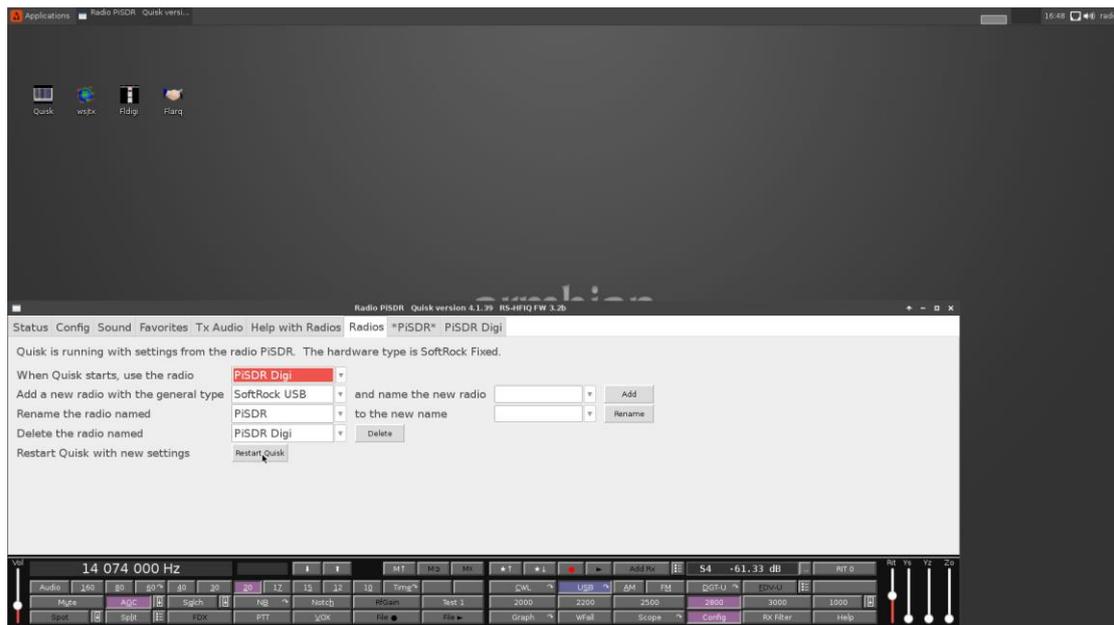
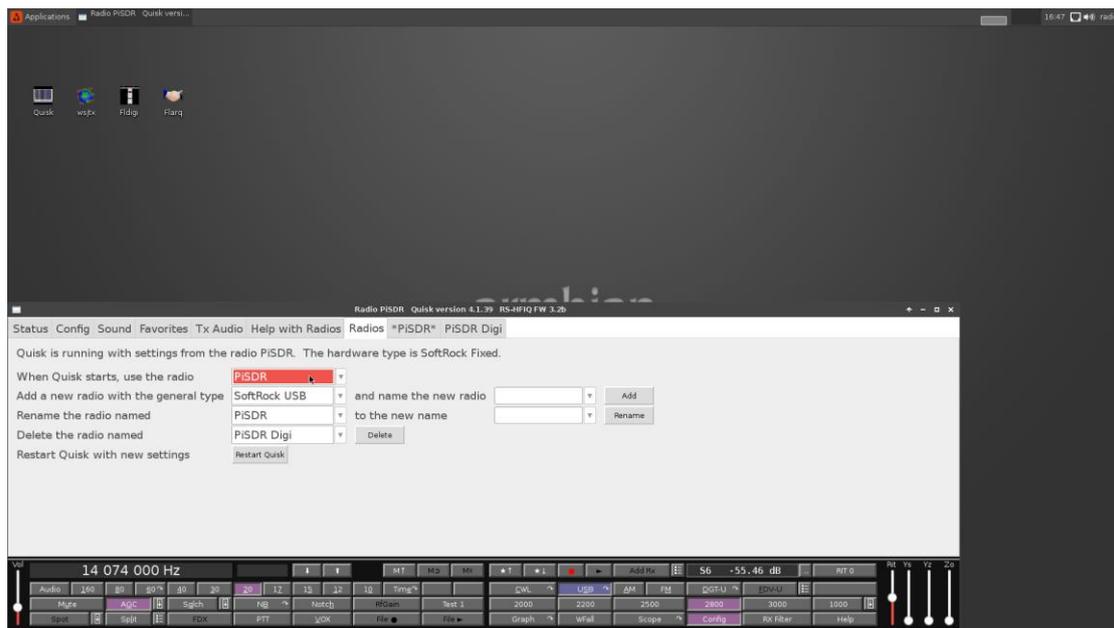


And here is Quisk in phone configuration.

A difference between the two configurations, appart from the audio streams, is the sample rate, in PiSDR Digi the samplerate/bandwidth is 48KHz whereas in PiSDR it is 192KHz. User can change that if it is needed via the relevant Quisk “Config” menu.



Also, user can change back to Pi SDR Digi via the same method



After each time “Restart Quisk” button is used to switch Quisk configuration, Quisk closes and restarts automatically. Even though that happens, after the restart, close manually and start manually Quisk as a precaution to properly store settings.

The standalone firmware works with Softrock/EnsembleRxTx or RS-HFIQ rigs, for those two types there are relevant configurations that are enabled via desktop executable scripts. These scripts and configurations are located in /home/radio/quisk_configurations folder if a user wants to examine or modify them.

Overall, the supported rigs are those that there is Quisk support for them.

For anything regarding additional configuration a user can read and study the relevant documentation pages from the installed applications, so here are their links

Quisk

-> <http://james.ahlstrom.name/quisk/docs.html>

WSJT-X

-> <http://physics.princeton.edu/pulsar/k1jt/wsjt-x-doc/wsjt-x-main-2.0.0.html>

FLDigi

-> <http://www.w1hkj.com/FLdigiHelp/>