rx2fits - Live Waterfall Viewer
Installation & User Guide

Jacob F. Hansman
Dr. Richard Gray
Appalachian State University

Installation

fitsview-live.py provides a way to view spectrographic output in real-time from a RX888 MKII SDR interfaced through rx2fits, a software-defined receiver package for RX888 MKII developed by Nathan Townes to run on Linux-based operating systems.

To use the live viewer, first ensure rx2fits has been installed properly on a Linux system (Click here for the rx2fits repository). If you get an error message like the following when trying to start rx2fits:

```
grayro@Specstation2:~/rx2fits-code$ ./rx2fits start dso.xml rjove.xml
./rx2fits: line 73:  3359 Aborted (core dumped) nohup $spec oby=$obyfile obn=$obnfile >> "$stdout_log"
grayro@Specstation2:~/rx2fits-code$
```

This means that the fft library cannot be found by rx2fits. The solution is to place the following symbolic link in the directory "~/rx2fix-code/spec/bin/Release/net9.0", libfftw3f.so → /usr/local/lib/libfftw3f.so. Accomplish this with this command in that directory:

```
sudo ln -s /usr/local/lib/libfftw3f.so /home/observer/rx2fits-code/spec/bin/Release/net9.0/libfftw3f.so
```

Reboot your system and try again after the symbolic link is established. Ensure python3 is installed. Ensure .NET 9 has been installed properly and that its directory has been added to the $PATH variable that can be accessed by rx2fits' installation directory to allow use of the "dotnet" commands:

```
export PATH=$PATH:$HOME/dotnet_install
```

```
export DOTNET_ROOT=$HOME/dotnet_install
```

Reboot and then check the path. It should have /home/observer/dotnet_install at the end.

Download the folder "FITS" from the following link:
https://drive.google.com/drive/folders/1kIwOKvldWc7bgCBxemPPY-QzhwHq5L_A?usp=sharing. After downloading, navigate to within the rx2fits installation directory (~/rx2fits-code719382z, or something similar). In a terminal set to the rx2fits installation directory, run the following command:

```
dotnet clean rx888.sln
```

Then, delete the current "FITS" folder. Paste in the downloaded "FITS" folder from the above link in this directory to replace the deleted folder. Proceed with the following commands in a terminal set to the rx2fits installation directory:

```
find . -type d \( -name bin -o -name obj \) -print0 | xargs -0 rm -rf
```

```
dotnet restore rx888.sln
```

```
dotnet build rx888.sln -c Release
```

The commands above replace the base version of rx2fits with a modified version, with tweaks to *fits.vb* that allows for a mirrored output written to a local port on the same system. This mirrored output is picked up by the program *fitsview-live.py* and displayed in real-time. The full fits file of the current observation is untouched by the live viewer and saved to the directory set in the observation file.

Appalachian
STATE UNIVERSITY

For a more "modern-looking" UI, install the python package *customtkinter*. This is not necessary and the program will fall-back to base *tkinter* without it. If you plan on running the live viewer in a dedicated virtual environment, use the following command from within that environment to install this package:

```
pip3 install customtkinter
```

If you do not want to run the live viewer from a virtual environment, but still want the modern look, then use this command:

```
sudo pip3 install customtkinter --break-system-packages
```

User Guide

To use *fitsview-live.py*, within a terminal navigate to the "FITS" folder within the rx2fits installation directory ($\sim$/rx2fits-code719382z/FITS , or something similar) and run the following command:

```
python3 fitsview-live.py
```

This will bring up the live viewer's UI. To begin viewing, start rx2fits from the installation directory with:

```
./rx2fits start <observatory file[.xml]> <observation file[.xml]>
```

This will begin the mirrored output from the RX888 MKII and stream the spectrographic output to the displayed UI.

The live viewer has the capability to subtract a calculated noise floor for clearer visual output. To enable this function, run the program *find_noise_floor.py* (from a terminal set to the "FITS" folder inside the rx2fits installation directory) on a calibrated fits file generated on the same SDR and antenna:

```
python3 find_noise_floor.py
```

You will be prompted to enter the name/directory of the calibrated fits file. The output of this program is a file, *median.dat*, which is a calculated median noise floor that is then subtracted from the image displayed when the user selects the "- noise floor" check-box.

The program allows for optional bilinear interpolation, logarithmic scaling, visual limit adjustments for stretching/compressing the visual dynamic range around desired ADU/dB intensities, color gain adjustment, and color offset adjustment. These options are toggled by check-boxes and sliders on the UI. The program also allows for the use of 5 different color maps: Typinski/unipolar, AJ4CO, SAO, UNK, and Bipolar. Choosing different color maps is achieved through the drop-down menu at the top of the UI and changes the entire visual output instantly. The program has the flexibility to add any number of user-desired color maps through slight modifications of the script. Contact me (hansmanjf@appstate.edu or jacobfhansman@gmail.com) if you need assistance in doing this.

While this program is still running, user configured settings will remain stable with new observation sessions with rx2fits.

Notes

The "dotnet" command re-installation for rx2fits seems to be required. The base rx2fits' *./install configure* command does not allow the tweaked *fits.vb* to be installed.

This program was heavily modeled after Nathan Townes' *fitsview.py* and Huub Hameleers SWL/JO21N's *sps_fits_visualiser*. Development of this program would not have been possible without the support of Dr. Richard Gray.
Python/.NET package documentation analysis was supported by OpenAI.

Appalachian
STATE UNIVERSITY