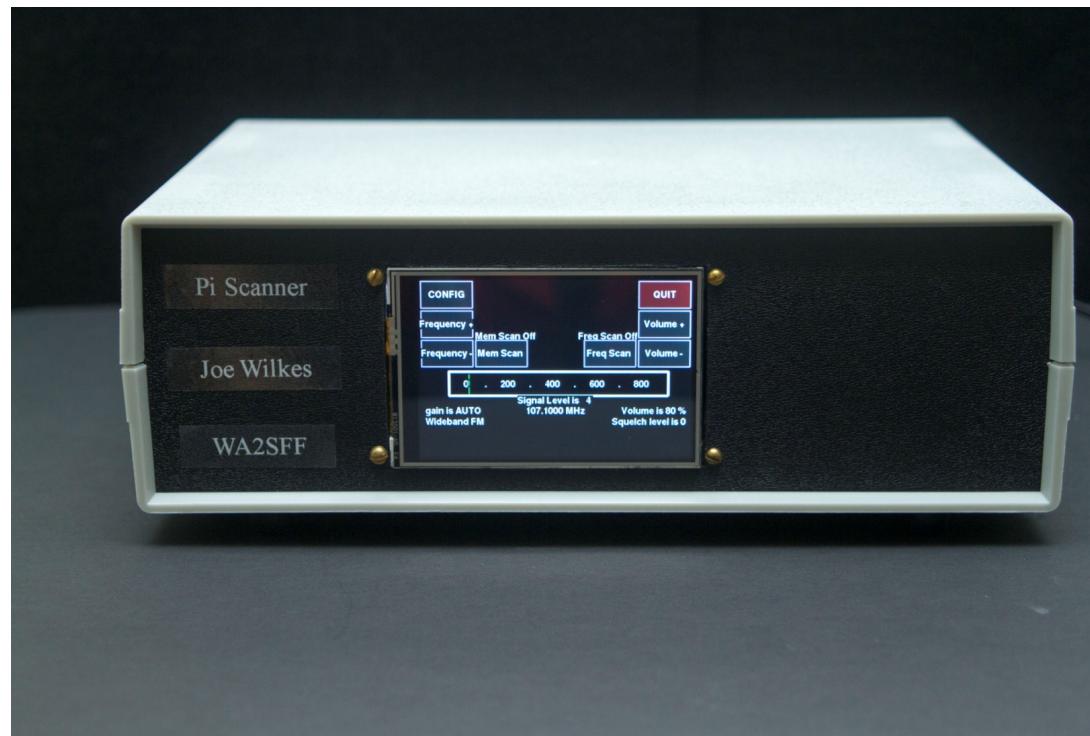


50 MHz-2 GHz Radio Scanner Using Raspberry Pi

Joe Wilkes, PhD

WA2SFF

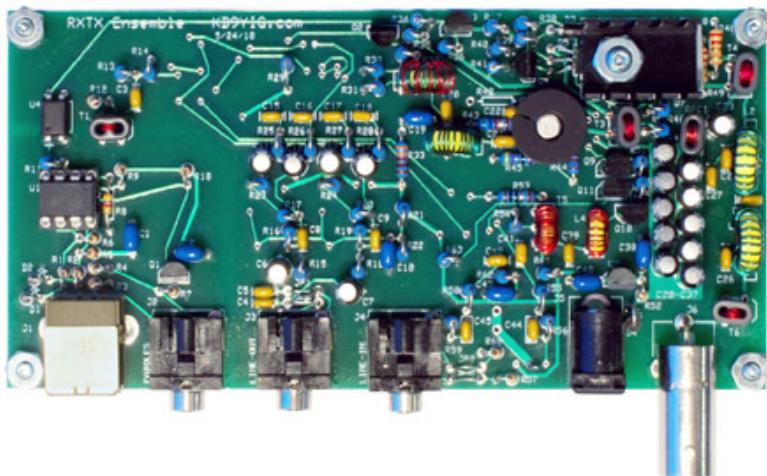


Pi Scanner copyright Joseph Wilkes 2017

History of my SDR Interest

- Initial Interest started around 1997
 - Attended early SDR standards meeting
 - Built some SDRs for work
 - Developed draft for SDR book (never published)
- Amateur Radio SDR Activities
 - Read Jan 2013 QST Article on USB radio dongle using SDR#
 - Immediately purchased parts for project
 - Gnuradio for linux
 - rtl_sdr and gr-osmo for linux
 - Multimode for linux
 - Unfortunately, Raspberry Pi is not fast enough to decode fm under gnuradio
 - After extensive web search found:
 - rtl_fm-python written specifically for ARM processors
 - Decided to build python based User Interface for Pi

Three Main SDR Approaches used by Hams



Softrock transceivers or receivers

Commercial SDR



Pi Scanner copyright Joseph Wilkes 2017

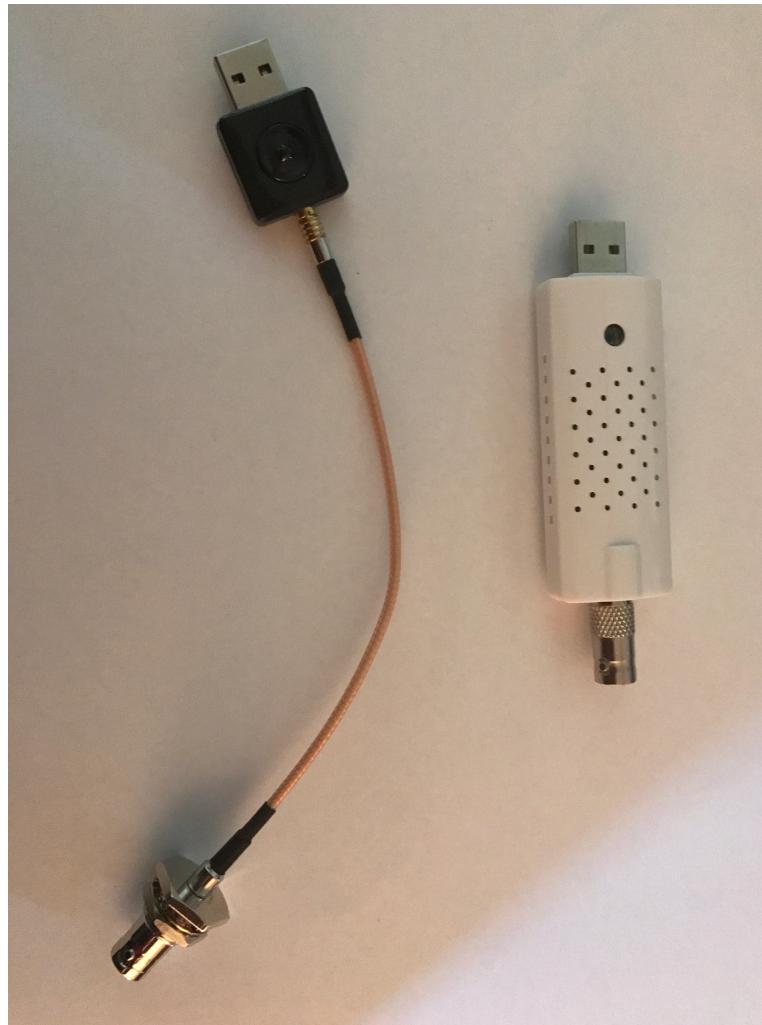


TV receiver USB dongle repurposed

This talk will focus on using dongles with Raspberry Pi

Unfortunately,
All dongles require
Custom adapters

Adds to cost



Raspberry Pi

- Small computer designed in UK to teach CS to students
- Has world-wide usage by students, teachers, hobbyists
- Many plug-in boards (called hats) available to extend capabilities



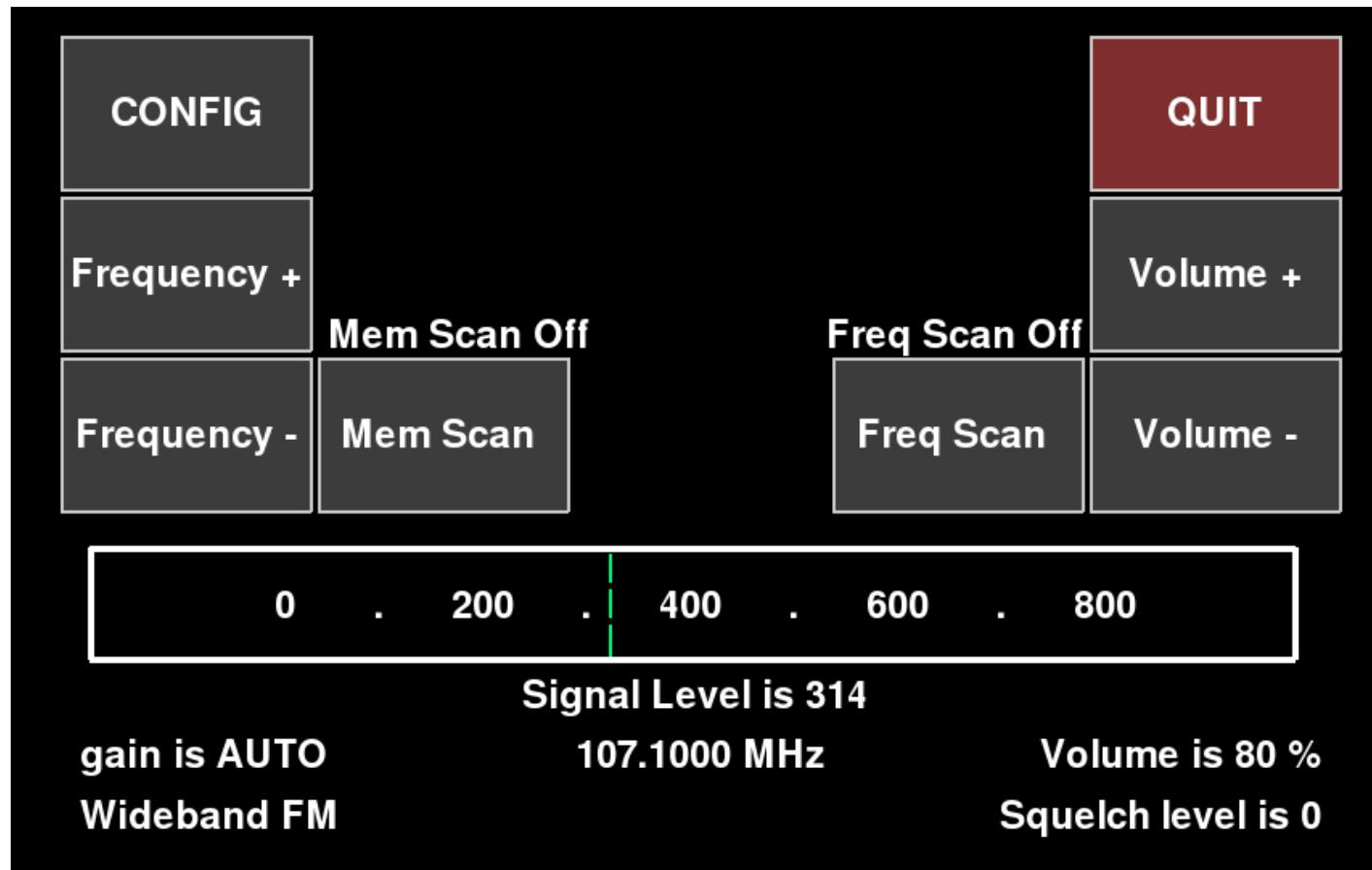
Hardware/Construction

- Raspberry Pi
- Adafruit 3.5" Touch Screen
- Adafruit 2.5 Watt Class D Amplifier
- SDR Dongle +adapters
- Misc cables, screws, rubber feet, etc
- Nice box (though large)
- USB Powered Hub
 - for Power and to use external USB devices
 - Pi can't power multiple USB devices

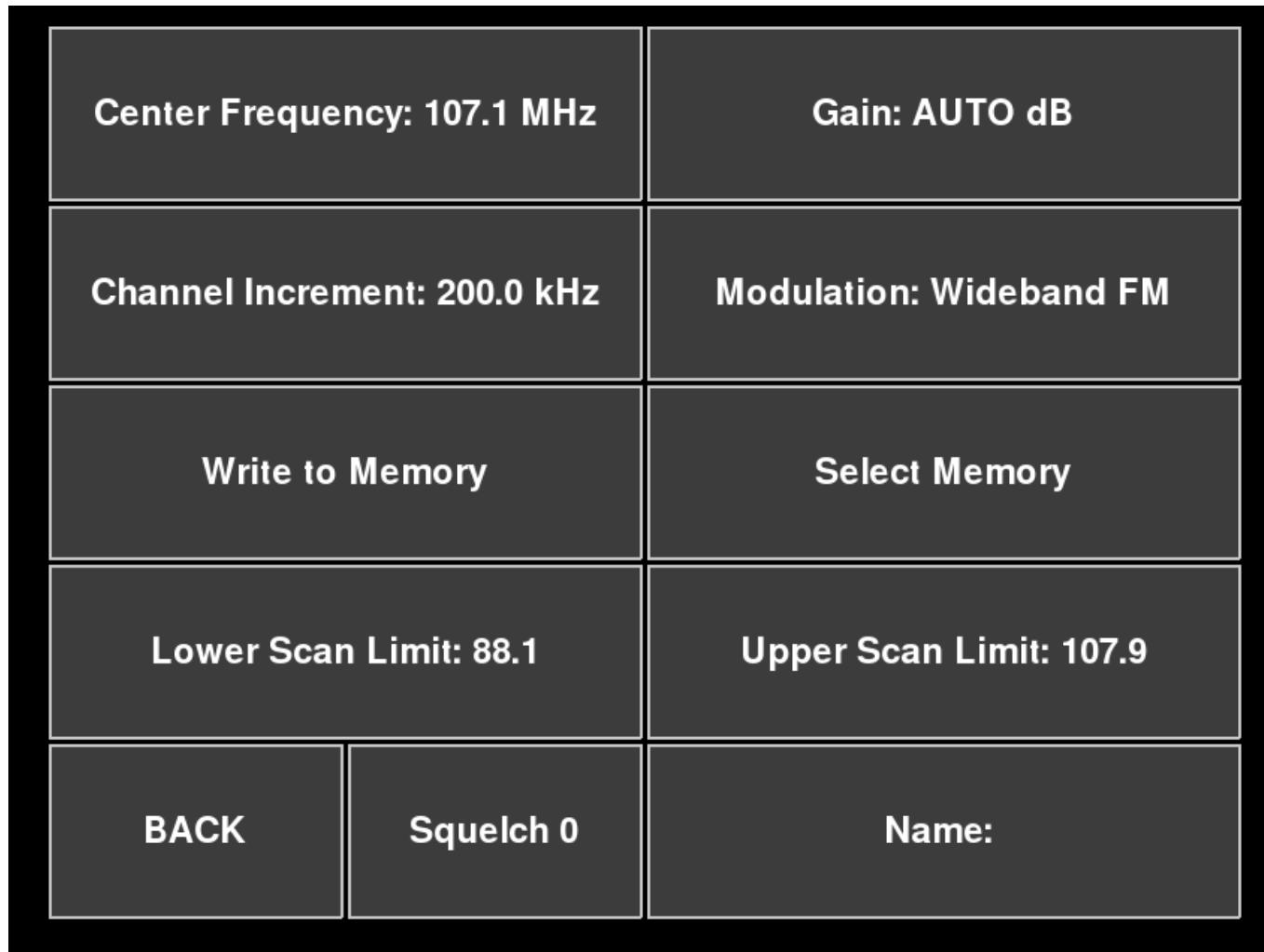
Features

- Self Contained
- Covers range of SDR dongle
 - Typically 50 MHz to 2 GHz (depending on dongle)
- Has touch screen
- Has signal strength meter
- Displays info on receiver
- Supports 100 memories
- Can scan frequency range or memories

Initial Screen



Configuration Screen

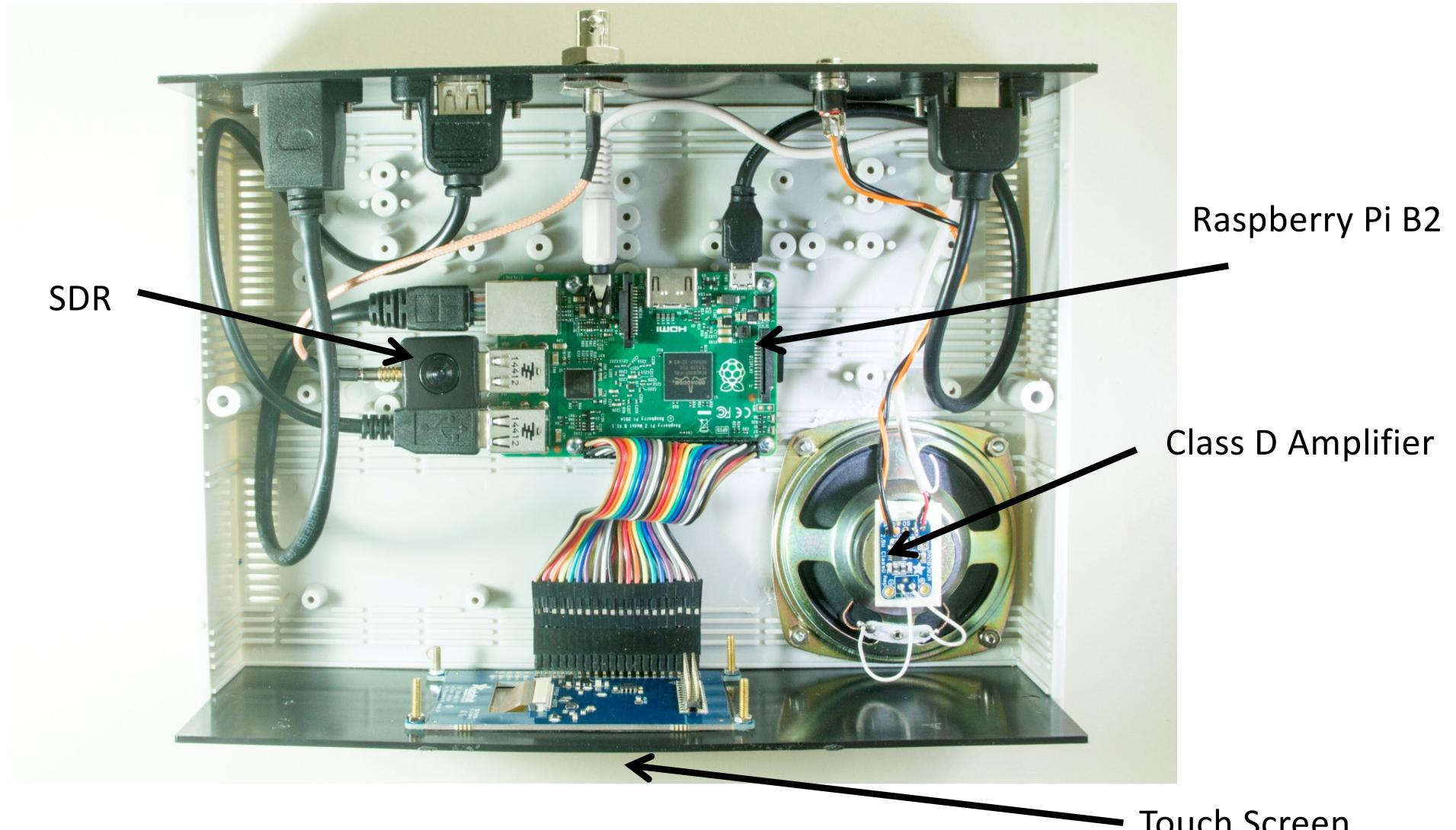


Front Panel



Inside View

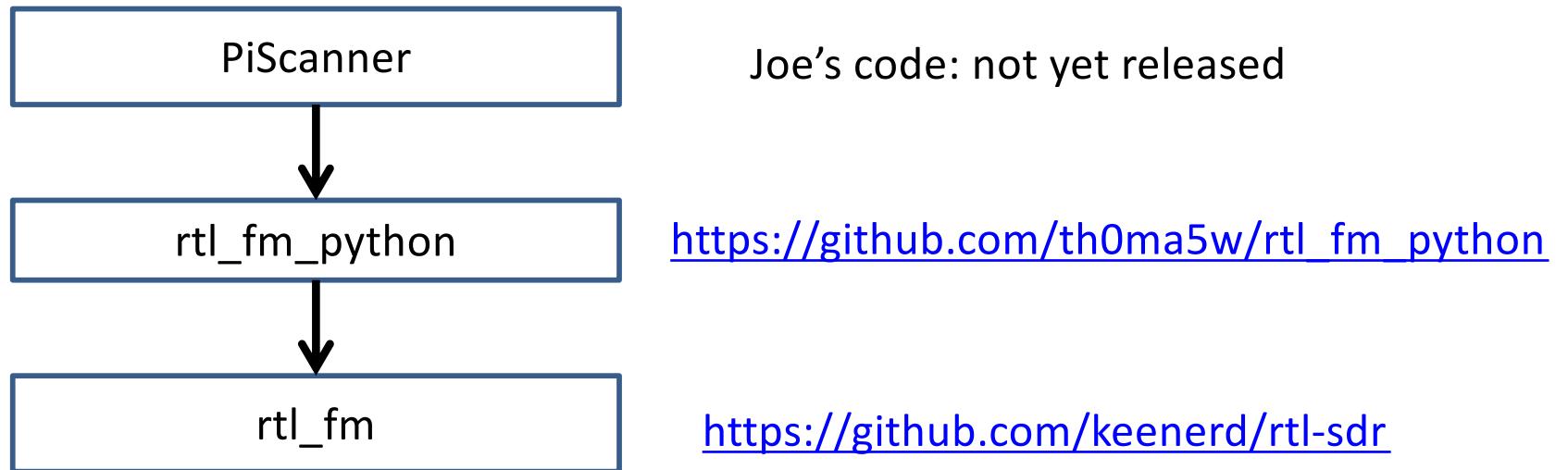
Ethernet USB RF AMP Pwr Pi Pwr



Design of Pi Scanner Program

- Based on adafruit spectrum analyzer program for Pi
- Written in Python
- Duplicated most common scanner functions
- sqlite3 database used for management of memories
- Calls to operating system using os command
 - Send and receive wget commands to control SDR
 - Control volume of sound card
 - All controls from touch screen (no knobs)

Code Organization



Code Development

- Code written on PC under Ubuntu
- Code uses MIT license
 - Anyone can use and/or modify code
- Code transferred to Pi
 - Implemented on touch screen
- config.py has parameters to switch
 - Ubuntu on PC vs Raspbian on Pi
 - Full screen display vs. Touch Screen

rtl_fm_python

- Set of programs written in:
 - C
 - Python
 - Java
- To implement SDR on low power processors
- Instantiates web server on local PC
- Communicate with SDR using web browser
- Linux supports command line version of web browser (`wget`)

Limitations

- Adjacent channel rejection is poor
- Dongle can be overloaded by strong signals

Improvements/Customization

- Add more memories (currently 100 in 10 banks of 10)
- Lock out memory banks when scanning
- Display actual frequency when using HF up converter
- Add knobs for volume, squelch, tuning in addition to touch screen
- Default upper/lower scanning frequencies and increment, when certain bands are selected (eg, FM band, 2m, 440, aircraft, etc)
- Use bluetooth speakers rather than internal speakers
- Improve signal processing algorithms

Summary

- Repurposed TV USB dongles cheapest way to get into Software Radios
- Most code is for Windows
- Raspberry Pi has limited capabilities for SDR but with proper code can be done

Software References

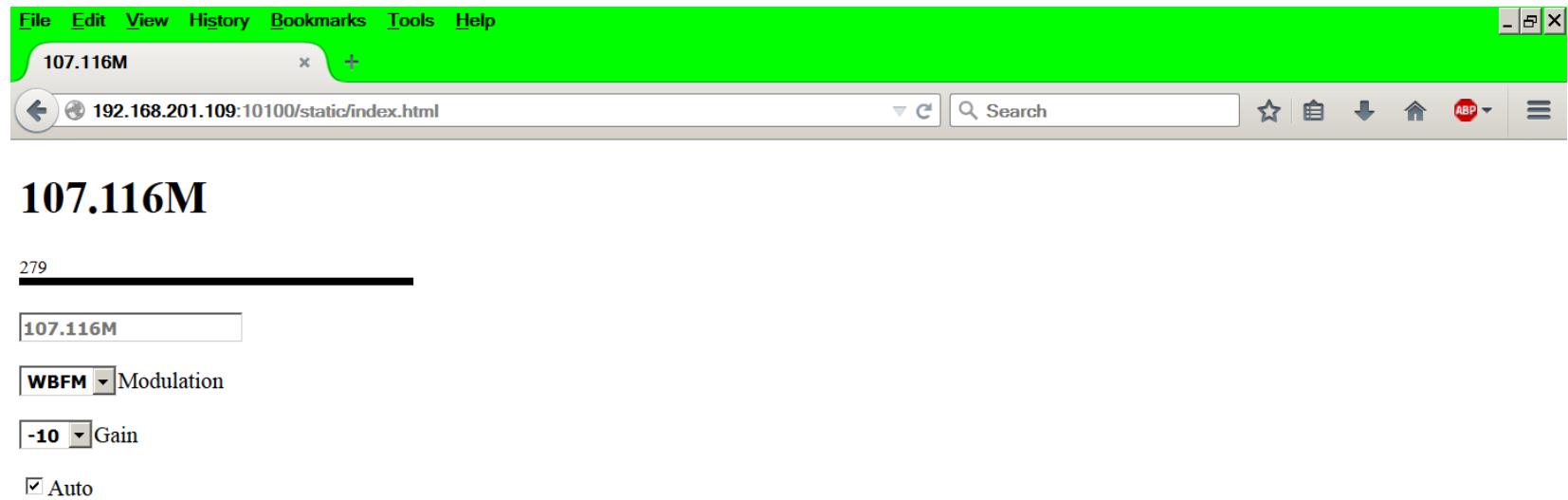
- Windows Software
 - <http://airspy.com/download/> (SDR#)
 - <http://www.hdsdr.de/> (High Definition Software Defined Radio)
- Linux Software
 - <https://www.gnuradio.org/> (GNU radio)
 - <https://osmocom.org/projects/sdr/wiki/GrOsmoSDR>
 - <https://www rtl-sdr.com/> (rtl sdr)
- MAC software
 - Very limited (still trying for find one that works)

Hardware references

- <http://www.nooelec.com/store/sdr.html>
- <https://www.adafruit.com/product/1497>
- <http://fivedash.com/>
- <http://www.sdrplay.com/>

Additional Background

Web Server View



Command Line web interface

- Instead of graphical web interface, I am using a command line program to get website data (wget)
- Example wget commands to get rtl_fm_python status:
 - wget -q http://127.0.0.1:10100/state -O - ; echo #get state of sdr
 - wget -q http://127.0.0.1:10100/demod/w -O - ; echo #set modulation to wide band fm
 - wget -q http://127.0.0.1:10100/gain/auto -O - ; echo #set gain to automatic
 - wget -q http://127.0.0.1:10100/demod/f -O - ; echo #set modulation to narrow band fm
 - wget -q http://127.0.0.1:10100/frequency/162450000 -O - ; echo #change frequency to 162.45 MHz
 - wget -q http://127.0.0.1:10100/frequency/162550000 -O - ; echo #change frequency to 162.55 MHz

Status result

- Example wget commands that are used to get status of SDR.
 - wget -q http://127.0.0.1:10100/state -O - ; echo
- # results of get state of sdr.
 - 127.0.0.1 -- [28/May/2017 15:24:33] "GET /state HTTP/1.1" 200 -
 - {
 - "autogain": true,
 - "freq_i": 107100000,
 - "freq_s": "107.1M",
 - "gain": -100,
 - "mod": "w",
 - "s_level": 330
 - }

Code Organization

Based on Adafruit Spectrum Analyzer Software

