The next step was to get Red Pitaya working with GNU Radio. There is a build of linux from Pavel Dimon. A website with some documentation can be found here: http://pavel-demin.github.io/red-pitaya-notes/

The first attempt didn't work. The second attempt worked. The problem was probably due to getting interrupted and failing to expand the filesystem.

My first impression, that the SD card image from Pavel would have the SDR transceiver code already installed, seemed to be in error. This article about using the Red Pitaya with gqrx was relevent: http://gqrx.dk/blog/gqrx-with-the-red-pitaya because it mentioned installing the app instead of using Pavel's build from github.

After installing the SDR Transceiver application from the Red Pitaya application marketplace, the SDR transceiver directory appeared in the right location, and the bit file could be loaded to the FPGA. The marketplace is accessed from a page visible after one logs in to redpitaya.com and uses the MY RP function. Click on the LAN IP address. It's a highlighted link that takes you to a page. That page has applications that are already on the Red Pitaya as well as an icon for the Application Marketplace. Click that to see all the applications available. GNU Radio successfully then recognized the Red Pitaya.

The AM transceiver flow graph was run, but no signals were heard. The antenna was not ideal, but AM radio signals are strong enough to where it shouldn't have mattered.

Next Steps:

1) Get a better antenna to prove the Red Pitaya isn't as deaf as it seems

2) Run gnuradio-companion directly on the Red Pitaya? We installed it and it ran. Why not use it that way instead of it being, essentially, a USRP.

3) Why doesn't the Red Pitaya build from Pavel have a fixed IP address out of the box like the documentation says it should?

4) What is the CPU utilization look like for the AM transceiver? Top reported 4-5% when gnuradio-companion was running on a host machine. Would something like oprofile help here? What's the right tool for this?