Raspberry Pi (security) Slow Scan Television Camera : Converting the image to a SSTV sound file

Notebook: Jonathan's engineering notebook
Created: 9/12/2014 2:10 PM
URL: http://www.instructables.com/id/Raspberry-Pi-security-Slow-Scan-Television-Camera/s...

Next we need to convert the image to a sound file which can be transmitted over the air. There are several SSTV implementations available for the Raspberry Pi.

PySSTV

First I had a look at PySSTV, a Python implementation which can be installed using pip:

```
pi@rpicamera ~/sstv $ sudo apt-get install python-setuptools
pi@rpicamera ~/sstv $ sudo apt-get install python-imaging
pi@rpicamera ~/sstv $ sudo easy_install pip
pi@rpicamera ~/sstv $ sudo pip install setuptools --no-use-wheel --upgra
de
pi@rpicamera ~/sstv $ sudo pip install PySSTV
```

This works, but it is very slooooooooooooow, it takes many minutes to convert a single image. So I continued to search for another option.

C implementation

Next I found a plain C implementation here:
https://sites.google.com/site/ki4mcw/Home/sstv-via-uc Unfortunately there were some errors in the preamble tones, but those were easy to fix. I also made it a little bit more flexible so that you can set the audio sample rate from the commandline line. Source of my implementation can be found on GitHub. To compile the source code:

```
pi@rpicamera ~/sstv $ sudo apt-get install libgd2-xpm-dev
pi@rpicamera ~/sstv $ sudo apt-get install libmagic-dev
pi@rpicamera ~/sstv $ gcc -lgd -lmagic -o pisstv pisstv.c
```

To run the program:

```
pi@rpicamera ~/pisstv $ ./pisstv /tmp/image.png 22050
Constants check:
        rate = 22050
        BITS = 16
      VOLPCT = 20
       scale = 6553
     us/samp = 45.351474
     2p/rate = 0.000285
 Checking filetype for file [/tmp/image.png]
 File is a PNG image.
 Input  file is [/tmp/image.png].
 Output file is [/tmp/image.png.wav].
 Writing audio data to file.
 Got a total of [2589556] samples.
 Done writing to audio file.
 Created soundfile in 4 seconds.
```

As you can see the SSTV sound file is created in just 4 seconds. So far
so good, next step, how to transmit the audio over the air.