Tracking Wildlife with Software Defined Radio in the VHF Band

Dave Witten, KDØEAG
2322 Deer Creek Ct.
Columbia, Missouri 65201
wittend@wwrinc.com

Abstract

Recent work in many fields involves use of technologies of interest to or popularized by radio amateurs. The low-cost, light weight, and low power demands of simple Software Defined Radios (SDRs) in particular have led to new capabilities in many fields. When combined with the processing power of small development boards such as Raspberry Pi and Arduino and the platform mobility provided by relatively affordable Unmanned Aerial Vehicles (UAVs) new areas of application appear. This paper discusses the recent testing of a system intended to assist the tracking of invasive carp species in the Mississippi river. It may also have potential for use in the 144 MHz band (or elsewhere) for 'fox hunting', tracking sources of interference, and similar purposes.

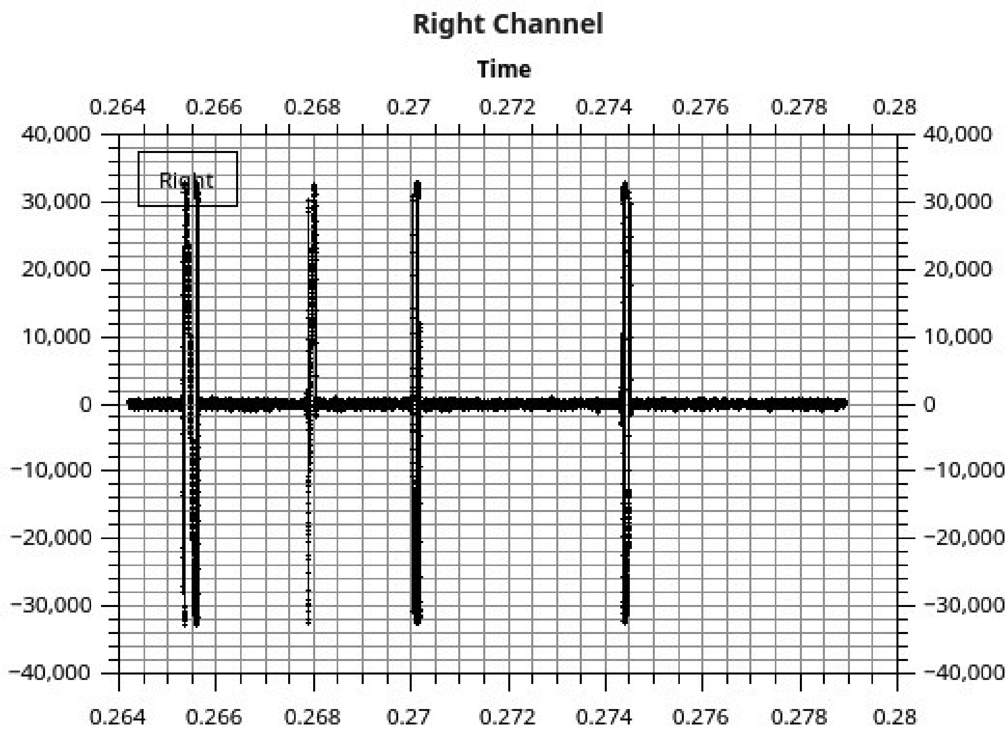
Keywords: Tracking, UAV, RTL-SDR

Background

This project began when our local radio club, the Columbia Missouri Amateur Radio Association (CMRA), was approached for assistance by an employee of the local United States Geological Services (USGS) office. They were pursuing a project to track the movement and life cycle of invasive species of Asian carp in the Mississippi and its tributaries. Many government agencies are recognizing the potential for UAV (drone) technologies to assist in their activities. But not all of these groups have access to the skills required to adapt existing methods to these new tools.

Much of the wildlife tracking performed by US government agencies involves signals emitted in the 162.0125 – 173.20 MHz band. It is significant that the transmission properties and equipment used in this band are generally similar to those used in the 144.0 – 148.0 MHz two meter Ham band. The most important difference, at least in this case, is that the signals used are groups of very narrow on/off keyed (OOK) pulses generated every 3 or 5 seconds from a tag placed in or on the animal of interest. The tags are expected to last three to five years. In this case, the signal encodes an identifier that is unique to a fish on a particular frequency. Specific frequencies are assigned to individual Principle Investigators (PIs). The coding scheme used can identify approximately 512 distinct fish.

Existing equipment used for tracking these tags by the USGS weighs 3.2 Kg excluding the antenna. The antennas typically used for boat-based tracking are usually large six or nine element Yagis of the sort usually found permanently mounted on a structure. They are heavy and impossible to mount on typical UAVs.



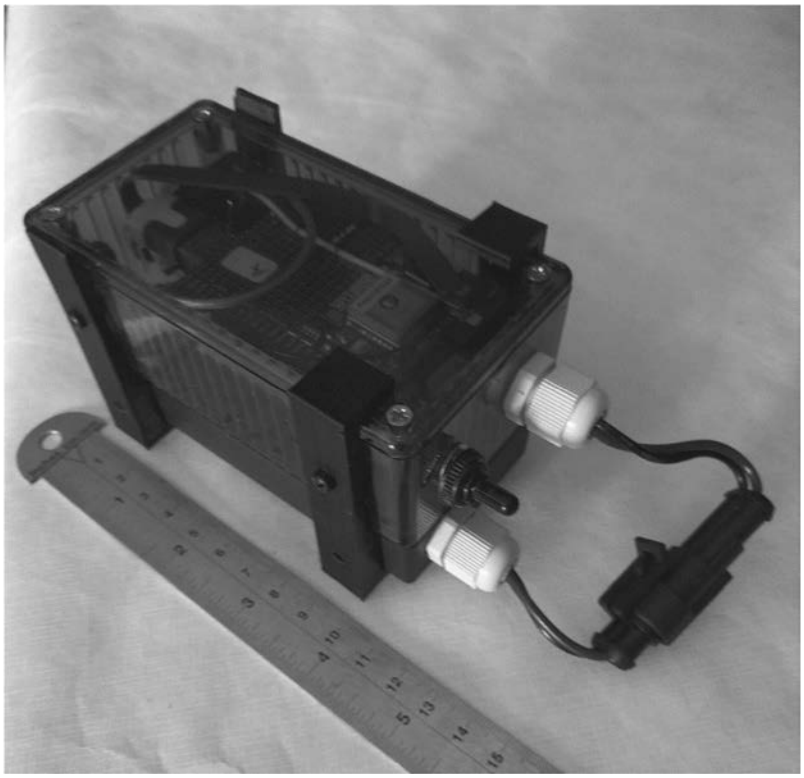
Single ID Code Received From a Tracking Tag

The choice was made to use a flight platform produced by the manufacturer DJI and owned and supported by NASA. This model, the Matrice 600, is a hexacopter typically designed to agriculture, public safety, and motion picture markets. It supports a total payload of about 5.5 Kg depending on the batteries used.

Design

Antennas

The VHF antenna for this project was developed by a separate contractor and information about the design used was shared under the terms of a non-disclosure agreement. While the details of the antenna used cannot be revealed it can be said that it used a serpentine design not commonly seen in the amateur literature. It appears to have performed well.



Tracking Receiver

The approach chosen for reception was to use a small USB dongle of the RTL-SDR type. This small receiver consists of a tuning chip (Rafael r820T2), a processing chip (Realtek rtk2832U), and support electronics to run on 5 V.

We were unable to obtain details of the specific encoding used by these tags. Various methods of analyzing the code sequences used have been applied by others. Because our goal was to provide localization of fish on a defined frequency rather than identify specific fish, we decided to focus on measuring signal strength from a UAV while it was rotated in fixed steps in the air.

Global Navigation Satellite System (GNSS)

Position over ground is determined by using GNSS signals captured by a dedicated receiver. Currently the only constellation used for determining a position fix is the American GPS system. It is possible that in the future the GLONASS and Galileo constellations may be captured as well.

Inertial Measurement Unit (IMU)

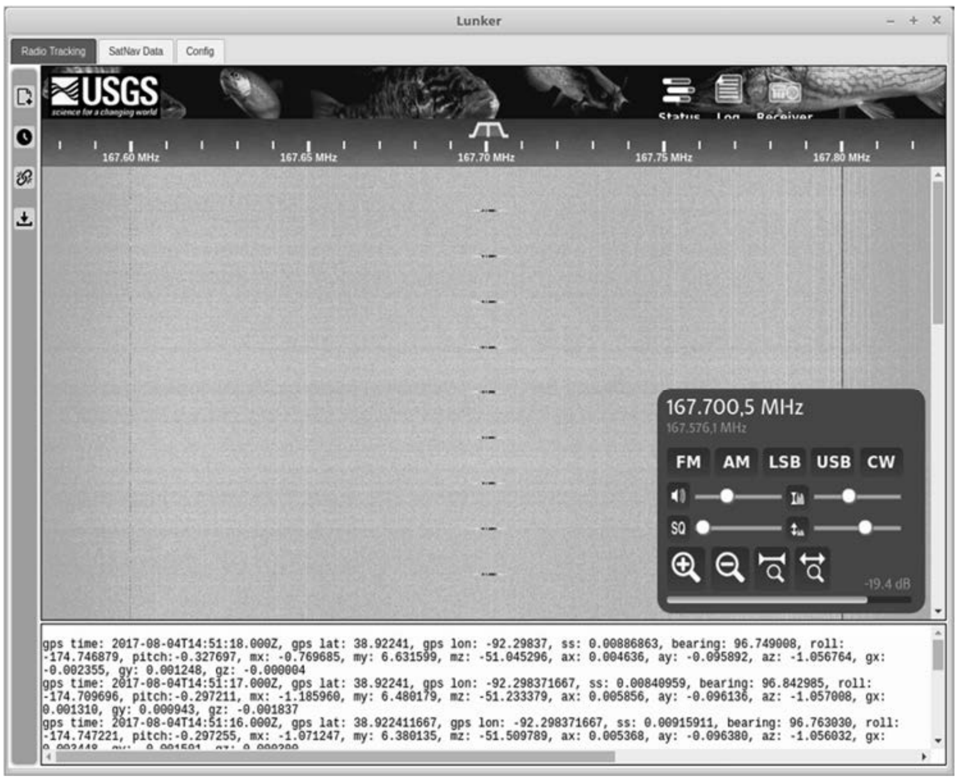
The system incorporates a common 9 degree of freedom (9DOF) IMU. The chip used is the LSM9DS1 manufactured by ST Microelectronics. This is a chip that combines an accelerometer, a magnetometer, and a gyroscope. Full fusion of the IMU measurements were performed by the telemetry CPU and relayed to the ground station. The individual components measured by the IMU were transferred as well as a mechanism for validation of the fused values during post processing.

Processor

The on board processing was done with an Odroid C0 board manufactured by HardKernel, a Korean firm. This board uses an Amlogic ARM® Cortex®-A5(ARMv7) 1.5Ghz quad core CPUs. This is a low cost board of the Raspberry Pi 3 class with a number of features useful to this project. It includes on board provisions that support running from and charging of a 1S (~3.7 V) lithium polymer (Li-Po) battery source, a small footprint, and optional installation of I/O connectors that may not be routinely required.

Telemetry Software

This version of the project used a variety of software, all published under one or another



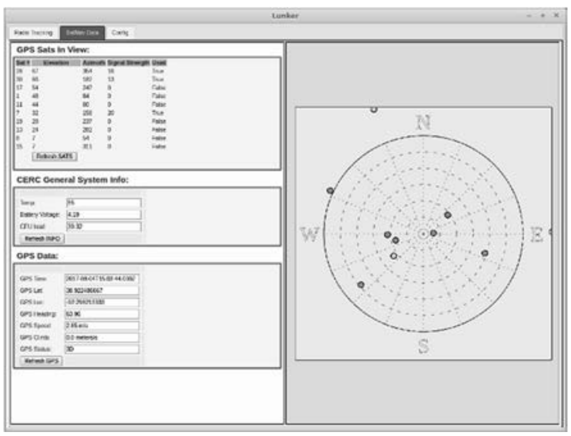
OpenSource license.

- GPSD (the GNSS daemon) is commonly considered a piece of the standard Linux/Unix canon. [1]
- RTIMULib, used to communicate with the IMU devices, have been placed in the in the RPi-Distro on Github. [2]
- Rtl-sdr tools are maintained by the Open source mobile communications (Osmocomm) project [3].
- csdr and OpenWebRx, written and supported by András Retzler. [4]
- Various python scripts to glue it all together [5]Telemetry Down Link

Communication from the sensor package to the ground station was accomplished using common, off the shelf 2.54 GHz WiFi link. This is one aspect of the design that needs careful consideration in future revisions. Probably due in part to the fact that the control signal for the DJI drone also use the 2.54 GHz ISM band, a number of telemetry anomalies were experienced that may well be due to an excessively high noise floor in that band.

Ground Station

The ground station hardware was based on a typical personal laptop running Windows 7 Ultimate. As the ground station software was written for portability, an alternate laptop machine running Linux Mint 18.1 with Ubuntu was also tested. This was linked via Cat 5 cable to a Ubiquity Rocket M2 access point coupled with a pair of modestly sized patch antennas in a MiMo configuration. The antenna was mounted on a sturdy tripod that allowed rough steering to keep the UAV in view. This entire configuration allowed easy deployment at a streamside location or on the flat deck of a pontoon boat.



The software application was developed and tested on Linux Mint 18.1 (Ubuntu 16.04). It is trivially portable to current Mac OS/X and newer versions of Windows. This flexibility is due to to having been written using NWJS and Node.js, a portable user interface development environment. This combination combines the visual interface features of Web development without the hardware access restrictions required by Internet browsers.

The most important element is a modified version of András Retzler's OpenWebRx. The OpenWebRx software runs as a server on the telemetry box, and presents a customized presentation of a waterfall display that allows tuning and bandwidth windowing.

Other data, shown in the log window, shows the data that is being transferred via AJAX requests. This data is being captured in CSV log files for post processing and analysis.

Testing

The telemetry system was tested in various settings for performance and compatibility with the flight control communications.

Columbia, MO

Initial testing was conducted at the USGS Columbia Regional Environmental Center. The telemetry package was mounted on a 40 foot (~30 M) crank-up tower. Reception of the VHF tag reception and transmitted telemetry data was verified at all available distances. The maximum available distance was about 600 meters. Reception with the fish tag submerged in water was successful up to about 200 meters. The conductivity of the water was measured to be 600 micro Siemens/cm, approaching the maximum generally regarded as usable at this frequency. Higher concentrations of dissolved ionic materials (and higher conductivities) defeat the ability of radio waves to propagate through water.

NASA Ames

Photo by Robert Dahlgren, PhD

The next step in testing took place at the NASA Ames Research Center in Mountain View California. There the telemetry payload was tested attached to the UAV platform which was flown tethered at NASA's Roverscape testing area. These tests proved the ability of the WiFi based down link to coexist with the control signals of the UAV without catastrophic interference.

Here, however, we did observe a problem with a 'donut hole' effect in the downward vertical 2.54 GHz antenna used by the system. As the ground station was operating at a fixed distance (about 45 M) from the launch point, we experienced a highly repeatable drop in signal strength that resulted in a failure of the telemetry transfer whenever the UAV rose above a height that created a 25 degree angle relative to the ground. Experimentation implicated the typical horizon-oriented donut shaped radiation pattern of the vertical antenna in use. Further study convinced us that this problem would diminish when the system was operated with a more realistic separation of the flight platform and the ground station. In subsequent testing this proved to be the case.

Ted Shanks Wildlife Area

The final evaluation of the system occurred just upstream of the mouth of the the Salt River, a tributary of the Mississippi about 85 miles north of St. Louis, MO. This location is part of the Ted Shanks Conservation Area operated by the Missouri Department of Conservation. Here the river water conductivity was measured to be on the order of 200 uS/cm, a level relatively favorable for VHF signals, possibly due to recent rains in the area. At this site, we performed extensive tests designed to test the capability of the system to detect and report fully submerged targets.

Amateur Uses

Fox Hunting and other activities involving direction finding are both popular as recreation and practical tools for locating sources of interference with normal operations (QRM and QRN). Amateurs also experiment with related tools for location in emergency rescue situations. Very little modification of this method is required to adapt it to these purposes.

The primary adaptations for amateur use vary with the frequency band used. Customization of the antennas used is the most significant area for conversion of this project to HAM VHF bands. For the size of drone that we used, the 167.700 MHz frequency made ideal antennas difficult. At 144 MHz the problem is probably somewhat worse. The 220 MHz band would have been ideal for the Matrice 600, but it is larger than most consumer UAVs. In the 70 cm band and higher antenna size (and usually weight) become less of an issue.

One difficulty encountered was a lack of information about less-than-optimal antennas. We routinely use antennas with dimensions far less than ideal for the frequency involved. A cell phone operates on any number of frequencies for which optimal antenna designs would be larger than the device itself. All elements of an integrated system require compromises. Which parameters of common antenna designs can be sacrificed, for instance, to trade away sensitivity to reduce size without compromising directivity? Few common references seem to approach the subject from this perspective. In this case, there is plenty of room for antenna experimenters.

Other adjustments to make this Ham friendly mostly revolve around tweaking parameters such as frequency and data interrogation rates for the sensors.

Conclusions

IMUs are aggravating. The calibration required is a major complication to the operation of this design in the field. Here the IMU acts only as a compass to determine the orientation of the antenna relative to geographic North. Other methods are possible. One approach might be to employ two GNSS (GPS) receivers and antennas on a baseline and attempt to implement carrier-phase analysis between two received signals. Typical GNSS software employs code-phase analysis for position determination, but the uncertainties involved yield orientation values that are too imprecise to use. Carrier-phase analysis is said to increase the accuracy of orientation determination substantially.

Unless specifically needed, lower rate down link of telemetry data should be implemented. The data rate provided by the OpenWebRx software seems modest and provides a waterfall display that is appealing to users. In practice, however, it is still too likely that available bandwidth will drop during a session causing a ‘digital cliff’ effect. In this situation receive buffers under-run and an unrecoverable break in the decoding of data occurs. While some of this might well be optimized out of the system by further tweaking, much of this data transfer was unnecessary.

The digital cliff issue was not observed under test conditions with the telemetry components transferred via JSON packets based on AJAX requests issued by the client. In future designs it may well be useful to consider 900 MHz LoRa or other options for the telemetry data. The lesser data rates and lower free air absorption at this frequency may well increase reliability.

Acknowledgments

Purpose, Advice, and Patience

Duane Chapman, Research Fish Biologist, USGS Columbia Environmental Research Center

Karl Anderson, Fish Biologist, USGS Columbia Environmental Research Center

Leanne Hanson, Biologist, USGS Fort Collins Science Center

Robert Dahlgren, Research Scientist, NASA Ames Research Center

Ethan Pinsker, Bay Area Research Institute, NASA Ames Research Center

Open Source Developers

András Retzler, OpenWebRx, csdr: http://blog.sdr.hu/about

Eric S. Raymond, *et. al.*, GPSD: http://www.catb.org/esr/

Too many others to cite...

References

[1] OpenWebRx and csdr: https://github.com/simonyiszk/openwebrx

[2] GPSD: http://download-mirror.savannah.gnu.org/releases/gpsd/

[3] RTIMULib: https://github.com/RPi-Distro/RTIMULib

[4] rtl-fm, rtl-tcp, *et. al.:* http://osmocom.org/projects/sdr/wiki/rtl-sdr

[5] Project code: to be published soon on at: https://github.com/wittend/CERC-tracking