Exploring new technologies and approaches for DATV Repeaters

The HB9TV Team who manages, operates and develops the HB9TV Network (**www.hb9tv.ch**) has started thinking about a new generation of repeaters for its network.

The current network is mainly based on DVB-S modules from SR-Systems, which have been working well in our 4 repeaters for over 10 years. Unfortunately, the hardware is no longer available and spare parts are becoming difficult to source.

In addition, the modules used only support the DVB-S standard with a minimum Symbol Rate (SR) of 1 Ms/s with MPEG2 video encoding, thus closing repeater access to stations using the DVB-S2 standard with SRs lower than 1Ms/s with H.264, H.265 video codec and AAC or AC3 audio codec in their DATV transmitter.

Finally, it should be mentioned that during the migration of our FM relays to DVB-S technology, in order to allow the coexistence of access with FM or DVB-S TXs, the repeater at the center of the network was equipped with FAGOR IFL6000 (transponder) and IFA600 (amplifier) modules designed for processing Analogue and Digital TV.

In view of the above, it seemed interesting to investigate in the technology of a linear transponder for our future repeaters.

A prototype was made with an ADALM-PLUTO module from Analog Devices.

By carefully reading the documentation for the AD936x RF Agile Transceiver used in this module, you can see that a **RF RX to RF TX loop-back function** is available. The loop-back happens in the ADI provided HDL core. The transmitter will transmit anything that the receiver receives. The entire RF chain is active (Sample rates, RF bandwidth and FIR settings will all effect the transmission).

We will use this function to create our linear transponder prototype. To make programming the AD936x of the Pluto easier, Analog Devices provides **pyadi-iio**, the Pluto's Python API.

Before you can use the API, you need to install the **libiio** and **libad9361-iio** libraries and after **pyadi-iio**.

The installation of these libraries is OS dependent. For non-geeks like me, **PySDR:A Guide to SDR and DSP using Python** (https://pysdr.org/index.html) will give you a starting point.

The very basic python program below configures the Pluto in "linear transponder" mode with the following parameters:

- Pluto connected via Ethernet interface at 172.22.22.150
- Sample rate at 8.192 Ms/s
- Rx Lo at 437 MHz with a bandwidth at 2 MHz
- AGC automatic in ‘slow attack’
- Tx Lo at 1280 MHz with a bandwidth at 2 MHz
- Tx hardware gain at -10 dB

Python program:

```
#
#  version 1.0   2021-12-20,  HB9DUG Michel
#
#  proto transponder DATV
#  input = 437 MHz
#  output = 1280 MHz
#  rf bandwidth = 2 MHz

import adi

# setup interface
sdr = adi.Pluto('ip:172.22.22.150')
sdr.sample_rate = 8.192e6

# Configure RX channel
sdr.rx_enabled_channels = [0]
sdr.rx_lo = 437000000
sdr.rx_rf_bandwidth = 2000000

# configure TX channel
sdr.tx_enabled_channels = [0]
sdr.tx_lo = 1280000000
sdr.tx_rf_bandwidth = 2000000
sdr.tx_cyclic_buffer = True

# Mute TX on power up
sdr.tx_hardwaregain_chan0 = -60

# Use RF loop back mode
sdr.loopback = 2

# AGC
sdr.gain_control_mode = 'slow_attack'

# TX on  (-60 to 0 dB)
sdr.tx_hardwaregain_chan0 = -10


while True:
    print(' ')
    stop = input('Return to Exit')
    sdr.tx_hardwaregain_chan0 = -60
    break
```

This simple prototype has been tested with success on the bench and in real conditions in the Lake Geneva area with DVB-S, DVB-S2 and DVB-T transmissions.

Its use is not only limited to a DATV transponder. It can also be useful as an up and down converter in front of a DVB-S/S2, DVB-T tuner or a spectrum analyzer.

We are sure that this demonstrator will foster interest in these new technologies and approaches for our DATV repeaters.

Please share your experiences with us and the ATV community !

References:

1. *PySDR: A Guide to SDR and DSP using Python*
   https://pysdr.org/

2. *Analog Devices Hardware Python Interfaces*
   https://analogdevicesinc.github.io/pyadi-iio/

3. AD9363 RF Agile Transceiver
   https://www.analog.com/media/en/technical-documentation/data-sheets/AD9363.pdf