QRPi – A Raspberry Pi QRP TX Shield Design

Zoltán Dóczi, HA7DCD
Budapest, Hungary
rfsparkling.com

Abstract

"Be Smart, Not Strong" this should be the self explaining phrase of the QRP term in amateur radio. Low power operation is always more difficult than using hundreds or thousands of watts RF power. But the smile on your face after the first thousands miles long QSO, using portions of one watt is worth the challenge! QRP enthusiasts instead of spending time and money on increasing power capabilities of its station prefer a smarter way: to learn about new modulations and coding techniques and applying them in everyday HAM operation practice.

Nowadays one of the most impressive QRP mode is Joe Taylor, K1JT's [8] WSPR [9] (pronounced "whisper"). WSPR stands for "Weak Signal Propagation Reporter". Programs written for WSPR mode designed for sending and receiving low-power transmissions to test propagation paths on the MF and HF and recently UHF bands. Users with internet access can watch results in real time at wsprnet.org

The QRPi board (or shield as referred by the community today) is an inexpensive way turning a Raspberry Pi single-board computer into a QRP transmitter.

Keywords: QRP, RPi, SDR, WSPR, open-source

Introduction

My QRPi shield is inspired by the WsprryPi [10] open source program, I've started to play with it as any other HAM operator, experimenting with the WSPR mode. At the beginning I followed the available articles [2] and DIY [10] guides about connecting a Low Pass Filter (LPF) to the RF output pin (GPIO 4) of the RPi computer. As an enthusiastic RF engineer and HAM operator I was instantly measuring the output signal with a signal analyzer and found a broadband noise from 0 Hz up to several harmonics [Fig 1] That was obvious that a LPF solves only the harmonic content attenuation and doesn't help against the broadband noise of the RF signal synthesized with the BCM2835's "General Purpose GPIO Clocks".

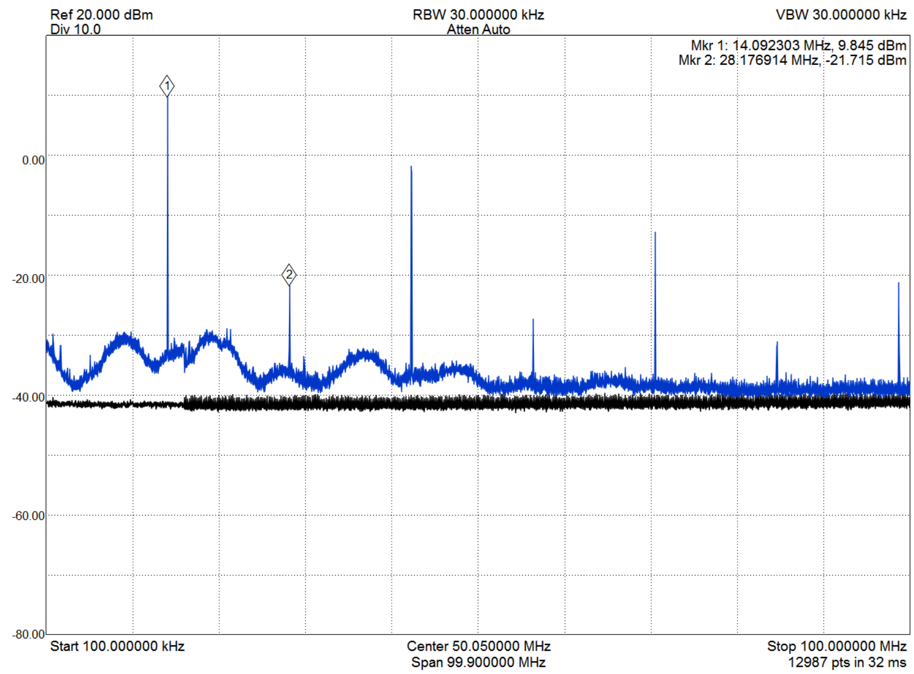

Figure 1. - RF Output spectrum of RPi's GPIO4 pin without filtering

At that point I made a research to find the possible inexpensive but efficient way to filter out the noise around the carrier. Lew Gordon's excellent article [3] led me to start my circuit simulations and to build my early prototypes based on his Band Pass Filter (BPF) advice. After successfully optimizing the BPF [Fig 3] values considering the parasitic parameters of the applied SMD inductors I saw a great improvement at the output spectrum [Fig 2].

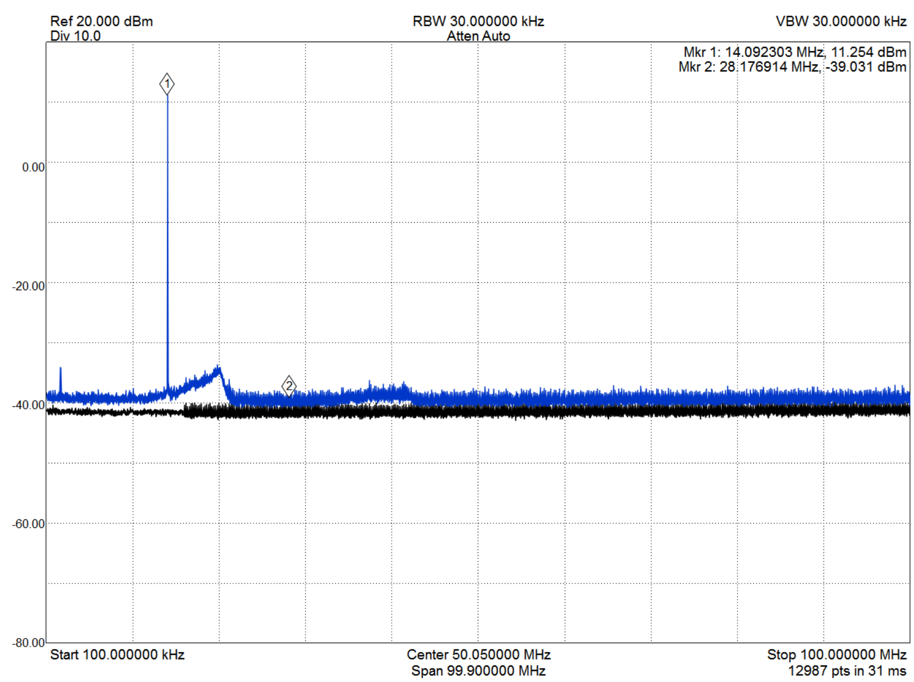

Figure 2. - RF Output spectrum of RPi's GPIO4 pin after BPF

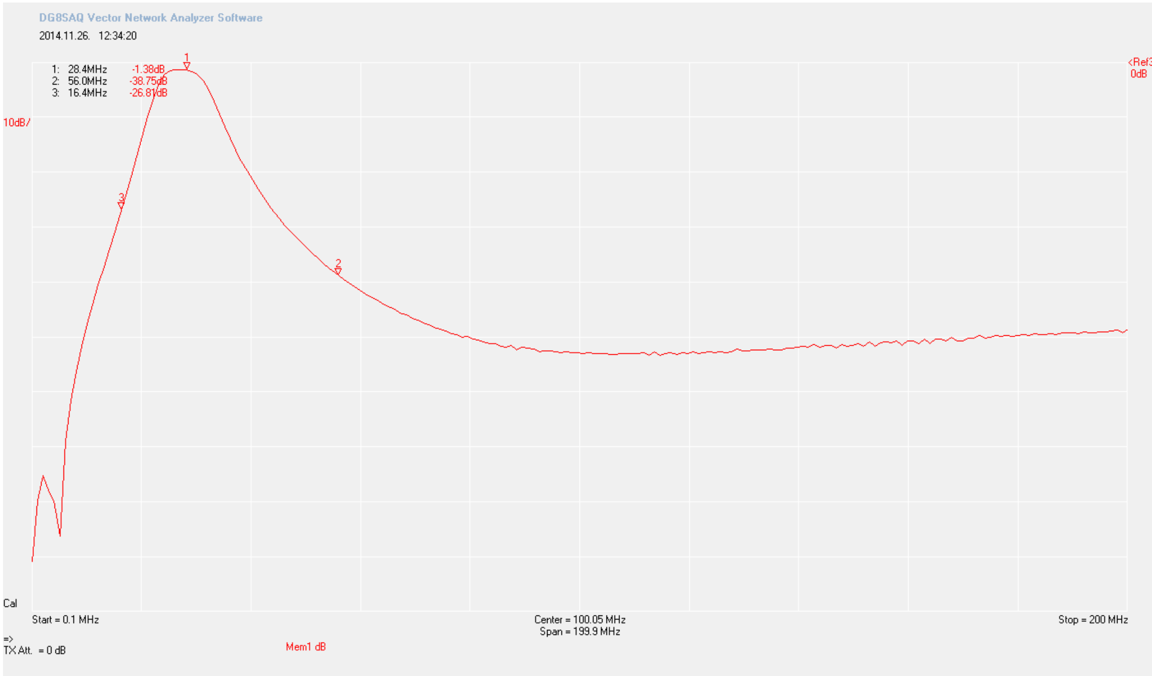

Figure 3. - Frequency response of the 10m, 3 element BPF on a VNA

At that stage of the design the harmonics filtered by the LPF, and the broadband noise filtered by the BPF were both acceptable. However there were still one thing missing: no buffer stage to protect the BCM2835 SoC's clock generator output stage. Hardware failure due to the unbuffered operation of the WsprryPi program was reported by a few HAM operators, possibly due to overload from nearly broadcast transmitter stations. If buffer amplifier was already needed it was a good idea to add some gain to the system. Eventually using a single FET amplifier stage [fig 7] 10 dB gain achieved, delivering +20 dBm output power at the end of the LPF [fig 4].

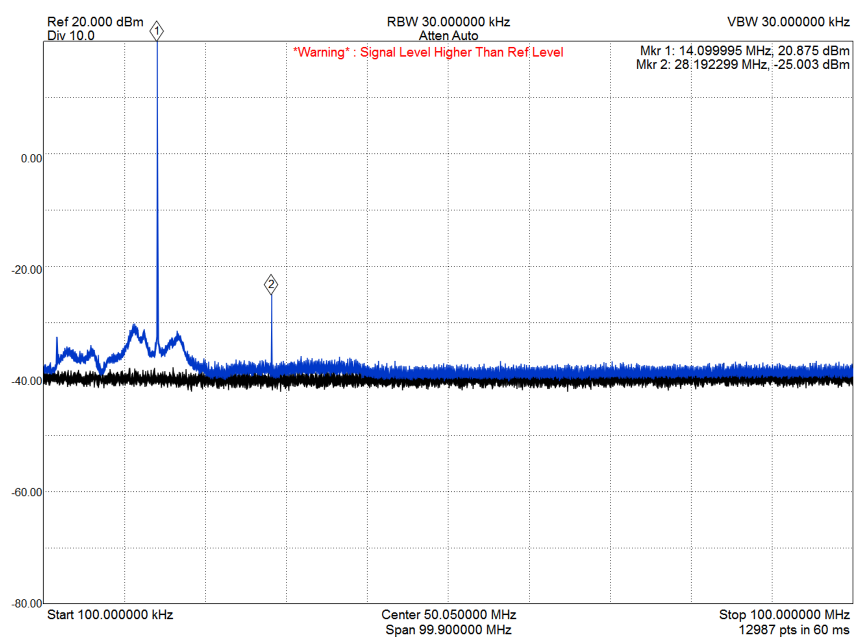

Figure 4. - RF output spectrum of RPi's GPIO4 pin after BPF + PA + LPF

For ESD and static discharge protection an ESD suppressor diode was added to the antenna terminal of the circuit.

I've targeted the absolutely smallest and most compact form factor. I've seen several RPi HAM accessories which were too "bulky" in my point of view. Using large external PCBs with long cables attached to the RPi it's destroy the true value of the card sized computer: small, mobile and flexible.

As I wanted to give an inexpensive QRP shield for the HAM and RPi community, mass production capability (SMD parts) and cheap component selection (eg. no SMA connector) was mandatory from the initial planning phase.

Regarding compactness I've exploited the advantages of inside PCB milling, leaving a gap for the RPi's LCD ribbon cable connector. This way the QRPi shield has low profile while sitting on the RPi, not even the highest point of the card-sized computer. This hopefully enables the use of popular stock RPi plastic enclosures with the QRPi.

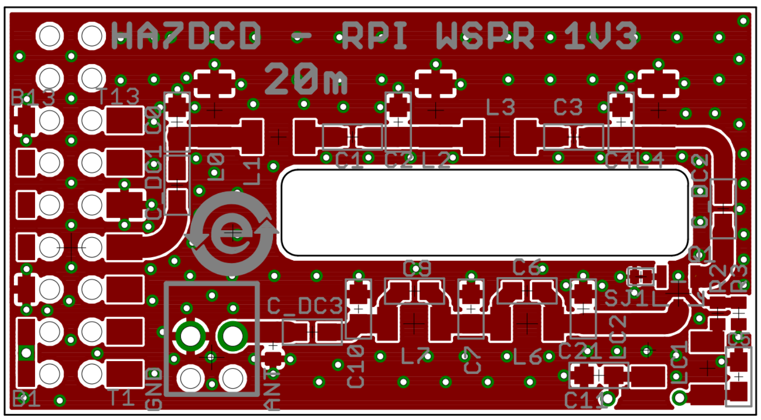

Figure 5. - CAD Layout screenshot of the QRPi shield

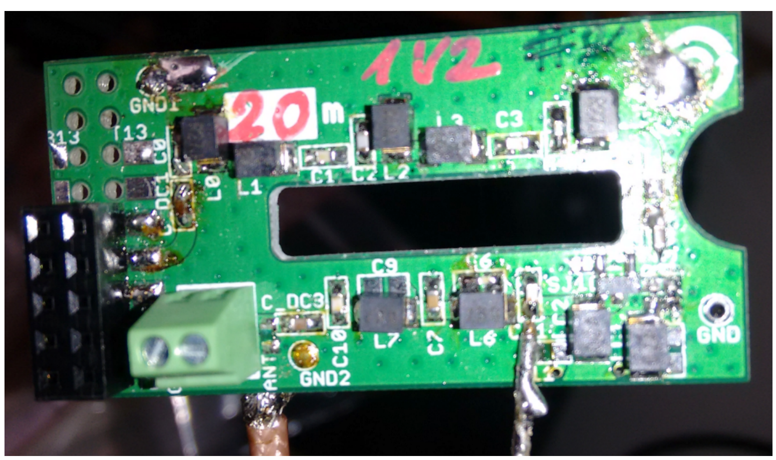

Figure 6. - Close up of a QRPi prototype PCB

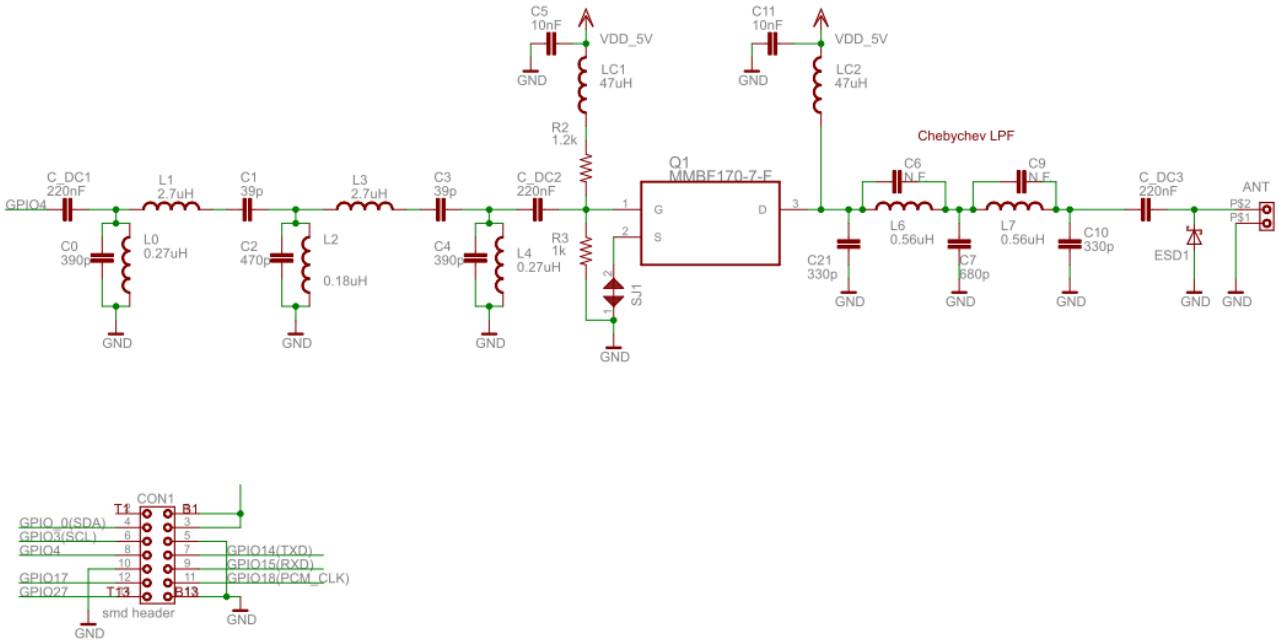

Figure 7. - Schematic of the QRPi shield

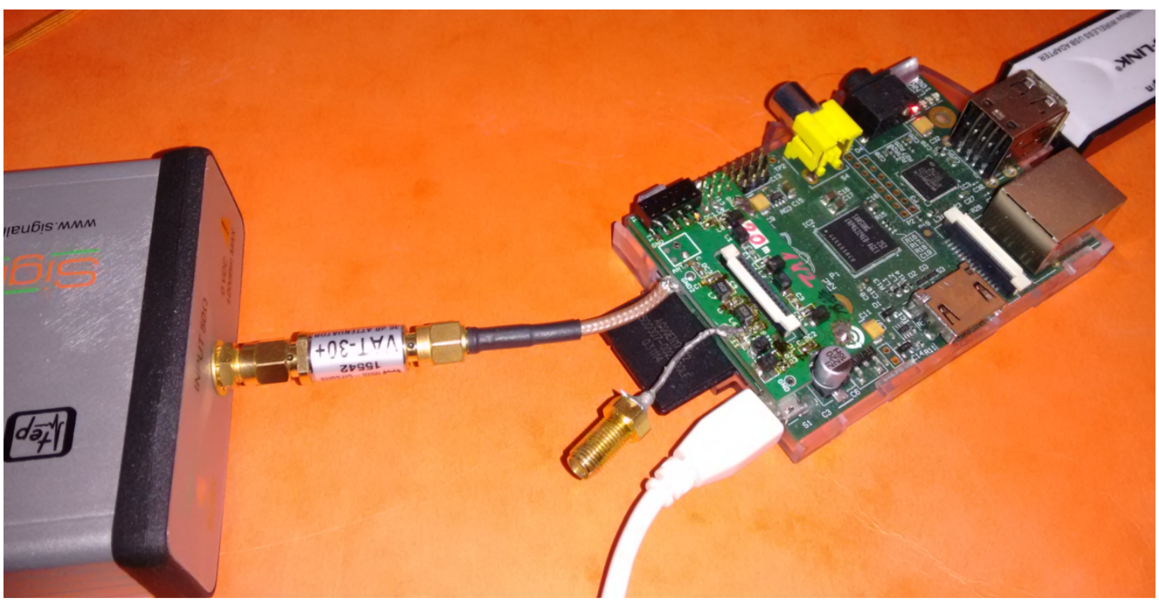

Figure 8. - Device Under Test on the spectrum analyzer

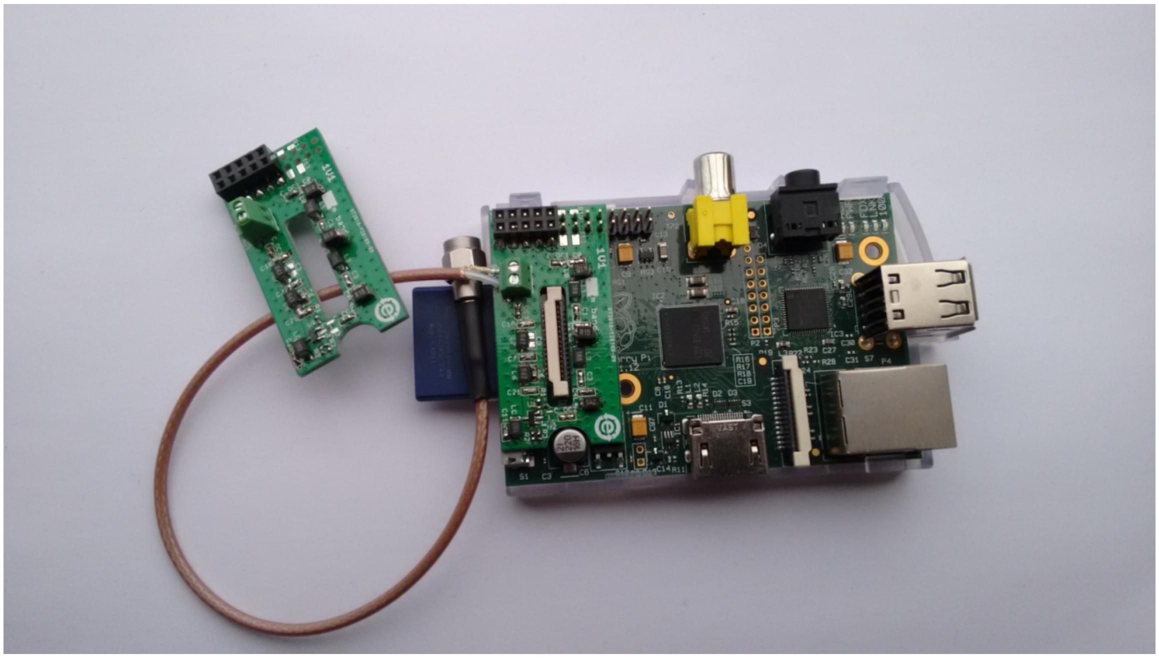

Figure 9. - QRPi on a Raspberry Pi computer

QRPi WSPR field tests

Laboratory measurements and fine tuning is one thing, another important factor is the real life operation and feedbacks from beta testers. Measurement and reports accumulated using WsprryPi and the QRPi shield since December 2014 till nowadays on daily basis using the latest prototype [fig 9]. Without any sophisticated antenna, using only a simple outdoor random wire at 2m height 1000...2000 km QSOs are typical on the 10 and 20m band with the +20dBm output power [fig 10-11].

Until the release of this paper the following digital modes and tools were tried and measured using QRPi:

- WSPR TX [10] - laboratory and field tests
- WSJT [11] - due to lack of resources not tested yet by the author of this paper, but seen feedbacks from HAM operators who use this with RPi
- CW TX [12] - laboratory tests
- SSTV TX [13] - laboratory tests
- Signal Generator tool [14] - laboratory tests

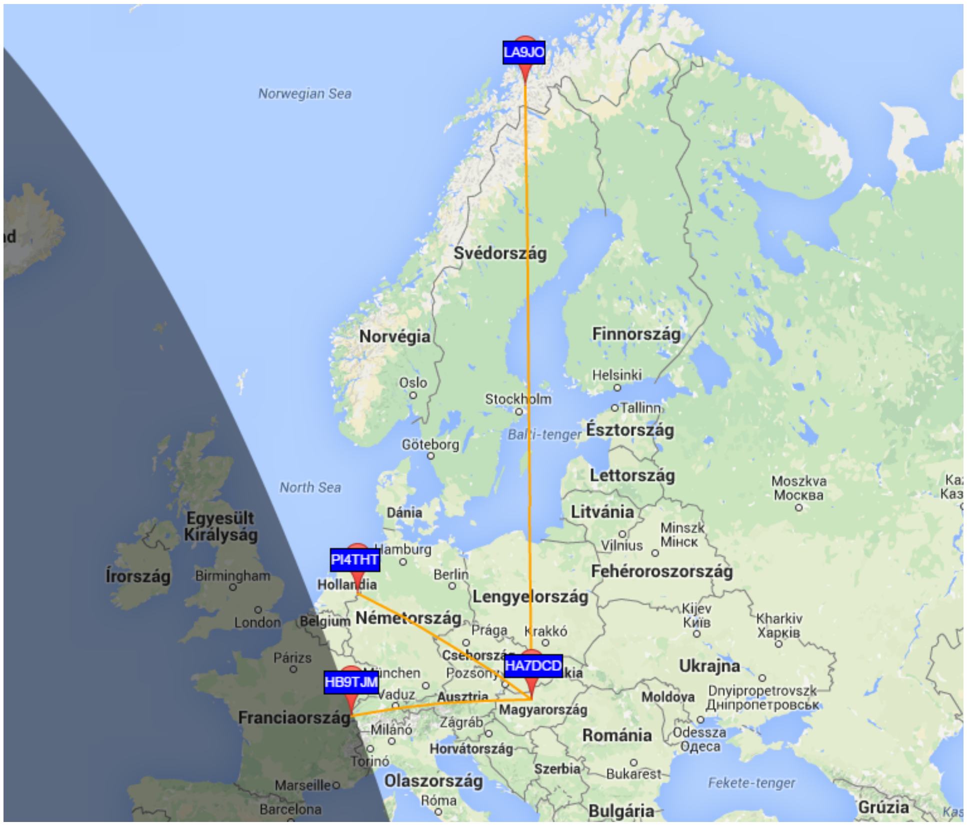

Figure 10. - HA7DCD - LA9JO, 2400km 14.097185 MHz, WSPR

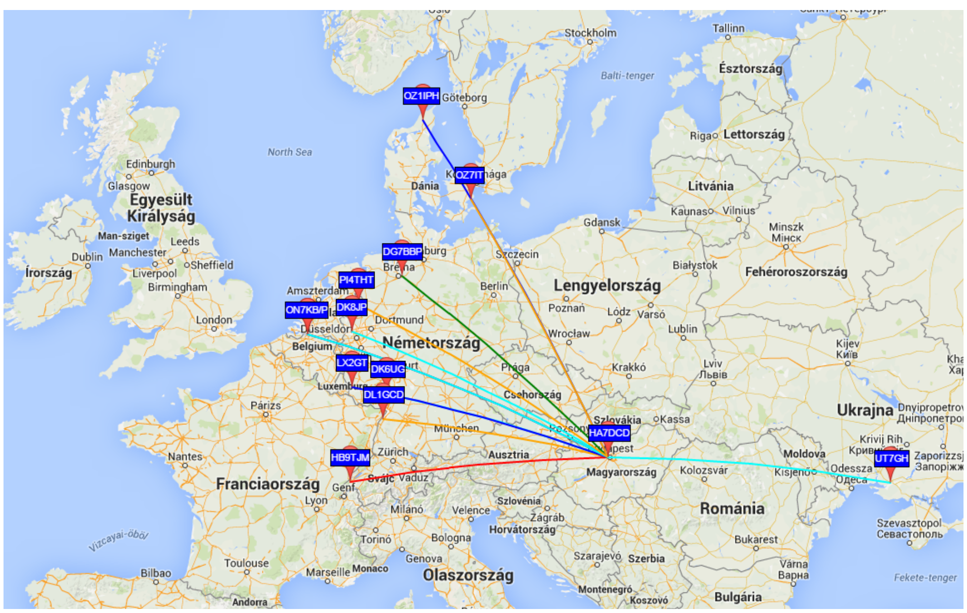

Figure 11. - Several European stations copying QRPi WSPR beacon, WSPR

Possible further developments

However the BPF filter does the job when filtering broadband noise coming from the BCM2835 SoC, there are still issues when amplifying the relatively wide band RF signal. From my measurements and investigations I realized that the FET amplifier starts to work as a mixer when the CW signal and the broadband noise filtered portion driven to its gate. The mixing product can observed at the spectrum analyzer screenshot [fig 12], ranging from 0 Hz to 10 MHz on the left side. I've successfully simulated the QRPi mixing product behavior using noise signal from a signal generator, or driving two sinus signal on the gate of the FET.

Investigation with FET biasing trials doesn't showed solution for this. The spurious content is still at acceptable level [fig 4] however for upcoming revisions considering another PA structure with less mixing behavior might be a good idea.

From software side a possible workaround should be to understand BCM2835's clock generators usage deeply. Fine tune it and decrease noise.

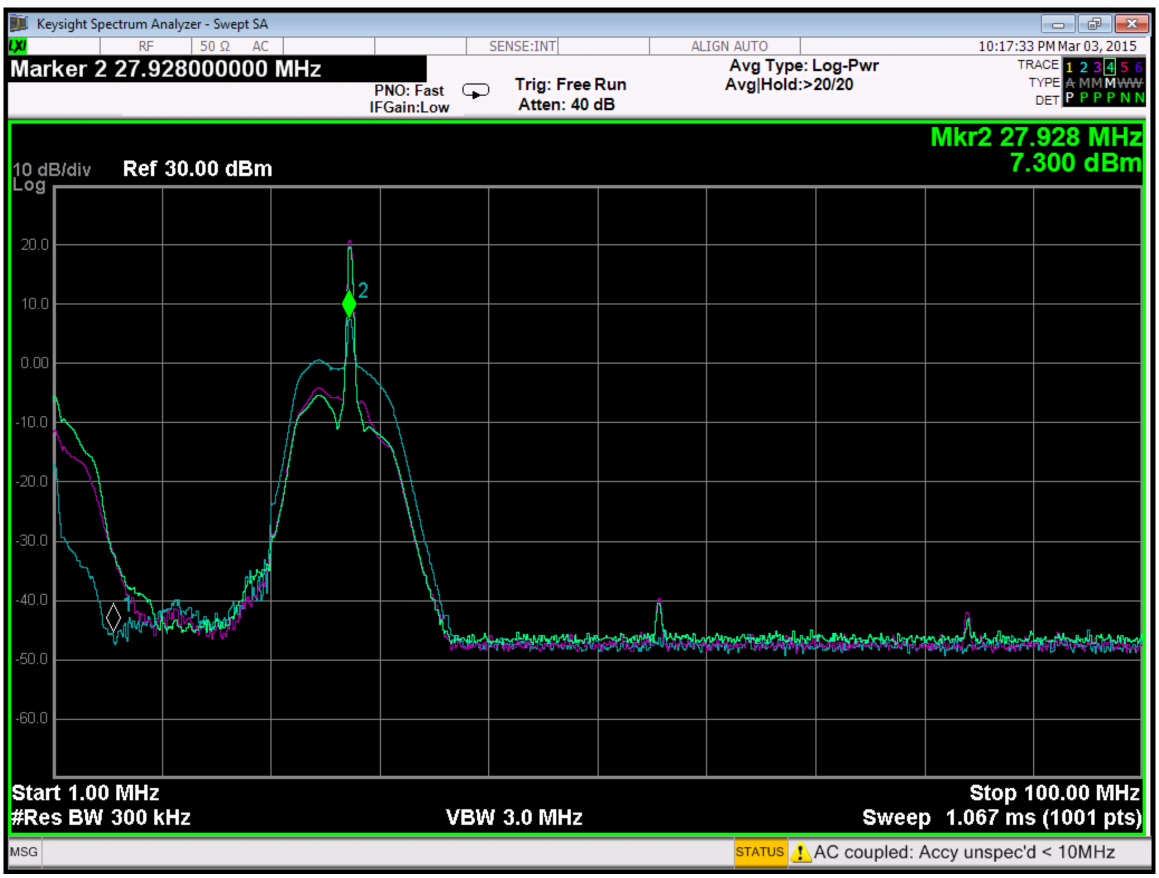

Figure 12. - FET amplifier acting as a mixer because of wide band noise

I would like to say thanks for the kind help of:

Steven Bible - N7HPR
Andris Retzler - HA7ILM
Hackerspace Budapest and András Veres-Szentkirályi "dnet" - HA5KBP
Tucson Amateur Packet Radio Corp
HA5KFU Schönherz Amateur Radio Club

Reference List

1. QRP definition - https://en.wikipedia.org/wiki/QRP_operation
2. TUTORIAL: TRANSMITTER (PA) OUTPUT FILTERS by Paul Harden, NA5N
   http://www.aoc.nrao.edu/~pharden/hobby/_lpf_pa.pdf
3. Band Pass Filters for HF Transceivers by Lew Gordon, K4VX
   http://www.arrl.org/files/file/Technology/tis/info/pdf/8809017.pdf
4. Raspberry Pi website - https://www.raspberrypi.org
5. Raspberry Pi Low Level Peripherals - http://elinux.org/RPi_Low-level_peripherals
6. Raspberry Pi SoC, Broadcom BCM2835 - http://www.raspberry-projects.com/pi/pi-hardware/bcm2835
7. BCM2835 datasheet - http://www.farnell.com/datasheets/1521578.pdf
8. K1JT website - http://physics.princeton.edu/pulsar/K1JT/
9. Weak Signal Propagation Reporter Network - http://wsprnet.org

Raspberry Pi transmitter programs working with QRPI TX shield:

10. WSPR - https://github.com/JamesP6000/WsprryPi
11. WSJT - http://hajos-kontrapunkte.blogspot.hu/2014/04/silent-whisper-jt9-on-cubie-truck.html
12. CW - https://github.com/JamesP6000/PiCW
13. SSTV - https://github.com/JennyList/LanguageSpy/tree/master/RaspberryPi/rf/sstv
14. Signal Generator - https://github.com/JennyList/LanguageSpy/tree/master/RaspberryPi/rf/freq_pi

15. HA5KFU Radio Club - http://ha5kfu.sch.bme.hu
16. Hackerspace Budapest - http://hsbp.org/