

## **Replacing the MiniTouner for Digital ATV Reception - Discussion Paper V1.0**

For over 4 years the European Amateur TV community have used the Serit FTS-4334 tuner as the basis of Reduced Bandwidth TV (RB-TV) receivers capable of receiving DVB-S and DVB-S2 at symbol rates from 35 kS to 4 MS and at frequencies from 144 MHz to 2450 MHz.

The FTS-4334 (and the similar FTS-4335) are no longer available from Serit, and a replacement needs to be found. This paper is intended to propose some options for the development of a replacement capability.

### **The Requirement**

The requirement is for a system that will receive and decode DVB-S and DVB-S2 Digital ATV signals at symbol rates from 125 kS to 2 MS.

**Constraints.** The following constraints are imposed, although they may have to be relaxed if a satisfactory solution cannot be found:

- The system should be easy to use by beginners, and not need tuning or “tweaking” to achieve a decode once the correct frequency (within 50% of SR) and correct SR are set.
- The system should be affordable with a total cost (excluding display) of less than £250 (or EUR250 or \$250). If an existing SDR is used (such as a LimeSDR or Pluto) this may be excluded from the cost calculation.
- The system should be practical to use in a portable situation.

### **Desirable Features**

- The receiver should cover as wide a frequency range as possible as it is intended to replace a system that tuned from 144 to 2450 MHz.

### **Possible Solutions**

Four initial possible solutions have been identified as a starting point for discussions:

Option 1. Writing/modifying code (for Windows, Linux x86 or Linux RPi) to work with an existing (obtainable) commercial tuner to receive RB-TV.

Option 2. Sourcing the Tuner chip and ASICs to enable us to build a new tuner and then writing the (Windows, Linux x86 or Linux RPi) code to control it.

Option 3. Using an SDR or Tuner chip and a custom FPGA (with amateur-written firmware) with simpler external (Windows, Linux x86 or Linux RPi) software.

Option 4. Using an SDR and custom software on a high-end PC.

The major challenge is the error correction for DVB-S2 decoding; this is where the custom ASIC has been used in the past. The task has only recently become possible using a software solution with development of high-end PCs.

## **Discussion of Possible Solutions**

### **Option 1 - Use an Existing Tuner with new External Software**

Serial continue to manufacture tuners and 2 of the most promising ones have been investigated.

The FTM-4762 is used in the BATC Knucker DVB-T receiver, but is also capable of receiving DVB-S and DVB-S2. It is used for this purpose in the promised VersaTune-Express Receiver being developed in the USA by Art WA8RMC. However, it does not work at low SRs and the internal firmware is closed source so is almost impossible to modify using amateur resources.

The FTS-3261 is uses Availink's 3rd Generation DVB-S/S2/S2X demodulation chip, the AVL6261C. See <https://github.com/availink/documentation>. Whilst it has proved possible to connect this tuner to the same USB interface as is used for the existing MiniTioner, no progress has yet been made on getting the tuner running. It's specification only extends down to 1MS symbol rate, but the actual lower limit can only be determined once the tuner is being driven by custom software. The BATC has purchased a number of FTS-3261 tuners for evaluation.

### **Option 2. Sourcing the Tuner chip and ASICs and building a New Tuner**

If the STV0910 and STV6120 integrated circuits could be obtained, it would theoretically be possible for the BATC to build a plug-in replacement for the FTS-4334. However, it is believed that the reason for the FTS-4334 no longer being available is that the ICs have gone out of production.

Use of different tuner and demodulator integrated circuits could potentially incur all of the problems of Option 1, with the additional burden of a hardware build.

### **Option 3. Using an SDR or Tuner chip and a custom FPGA**

The most difficult part of DVB-S2 decoding is applying the LDPC error correction; this is processor-intensive and has until recently been handled in custom ASICs or FPGAs.

There is at least one initiative (referenced by Roque PU2RQV recently) to replicate the GR-DVB GNU Radio software decoder on an open-source FPGA. His initial investigations are based on the ZC706 Xilinx Board and the Analog Devices AD9361 board AD-FMCOMMS3. Mike G0MJW is hoping to be involved in this project.

Another related potential solution might be to use an add-on FPGA "hat" for a Raspberry Pi 4 to provide the FPGA capability whilst controlling the SDR using the

Raspberry Pi native capabilities. One such FPGA board is described here:  
<https://www.latticesemi.com/products/developmentboardsandkits/raspberrypifpga>

Whilst DVB-S2 decoding is difficult, the problems of automatically fine-tuning both frequency and symbol rate should not be ignored; both of these problems need to be solved for a workable solution.

#### **Option 4. Using an SDR and Custom Software**

Solutions such as Lean-DVB <https://www.pabr.org/radio/leandvb/leandvb.en.html> using an SDR and software decoding have been under development for many years. In fact, Portsdown 2020 included an early version of the LeanDVB software which worked with an RTL-SDR to receive DVB-S signals on a Raspberry Pi and a touchscreen. This software has now been included in SDRAngel <https://www.sdrangel.org/> and decodes both DVB-S and DVB-S2 signals.

The problem with the current version of SDR Angel is that it is not easy to set up, and even once a signal is acquired, it often requires “tweaking” to achieve a decode. Additionally, it needs a reasonably high-end PC to operate. A custom development of SDRAngel might provide a PC-based solution.

There are a number of GNU Radio out-of-tree modules that can be used to decode DVB-S/S2. gr-dvbs2rx is described here <https://igorauad.github.io/gr-dvbs2rx/>. Additional modules would need to be incorporated to handle symbol sync and frequency offset correction. It might be possible to encapsulate all of these modules in an application wrapper, in a similar way to the Langstone microwave transceiver does for voice communications.

#### **Resources**

A number of volunteers have come forward to help with the project; it is hoped that more will be able to contribute. The current known volunteers are listed at the Annex. The BATC can potentially provide start-up capital funding for the development of promising solutions. BATC Committee members do have experience of arranging for the manufacture of small batches of populated surface mount PCBs for sale to members.

#### **Way Ahead**

This paper will be widely circulated for comment. What would be really valuable is suggestions for other credible solution options, and suggested means to achieve any of the solution options. Equally, it would be useful to know if any of the options are a dead-end.

Following analysis of the responses, the BATC Committee will decide which options deserve promotion and support.

Dave, G8GKQ  
10 October 2023

## **Annex**

### **Volunteers as at 10 October 2023**

Charles Brain G4GUO. Subject domain expert with extensive DVB-S/S2/T and tuner support expertise.

Brian Jordan G4EWJ. FTS-4334 interface programming expertise.

Achim Vollhardt DH2VA. PCB Layout and Design; prototype assembly.

Ohan Smit ZS1SCI. Polyglot programmer.

Tim Keep M0KEP. Embedded systems (software, firmware and hardware) developer.

Mike Willis G0MJW. MiniTiouner PCB designer and much more.

Roque PU2RQV. Potential FPGA-based DVB-S2 receiver designer.

George Blake W6BDD. Electrical engineer with CAD and PCB experience.