

A Solar Spectrometer for the 38 MHz band

The active sun produces strong, dynamic and complex radio bursts at frequencies below 200 MHz. Characterizing such bursts requires monitoring a wide range of frequencies, ideally between 20 MHz and 200 MHz or even higher. I own an ICOM R75 short-wave receiver which goes up to 60 MHz and has an RS-232 serial interface for remote control. A fairly new interest of mine is in using micro-controllers, so this also looked like a good excuse to try some practical coding. The idea was to see what could be done by scanning the receiver across a range of frequencies as quickly as possible, measuring the noise level for each channel as I go.

This is not the way to observe radio bursts seriously. It would be much better to use Digital Signal Processing (DSP) techniques, as the professionals do. However, it is the way that the e-CALLISTO solar spectrometer works and this instrument has produced some impressive results. I don't have any DSP skills and it would take time to acquire them. This could easily be wasted effort as I live in the middle of a large housing estate which is next to the town's main industrial area. The chances were that interference would mask out any useful information from the sun. My plan was to have a quick look using equipment that I had to hand and see if anything could be observed before investing time and adrenalin in a purpose-built solution.

Controlling the receiver

The Hardware

I built a simple controller unit on strip board, using a PIC18F2420 micro-controller - see Figure 1. The hardware requirements are so simple that a wide selection of micro controller devices can be used.

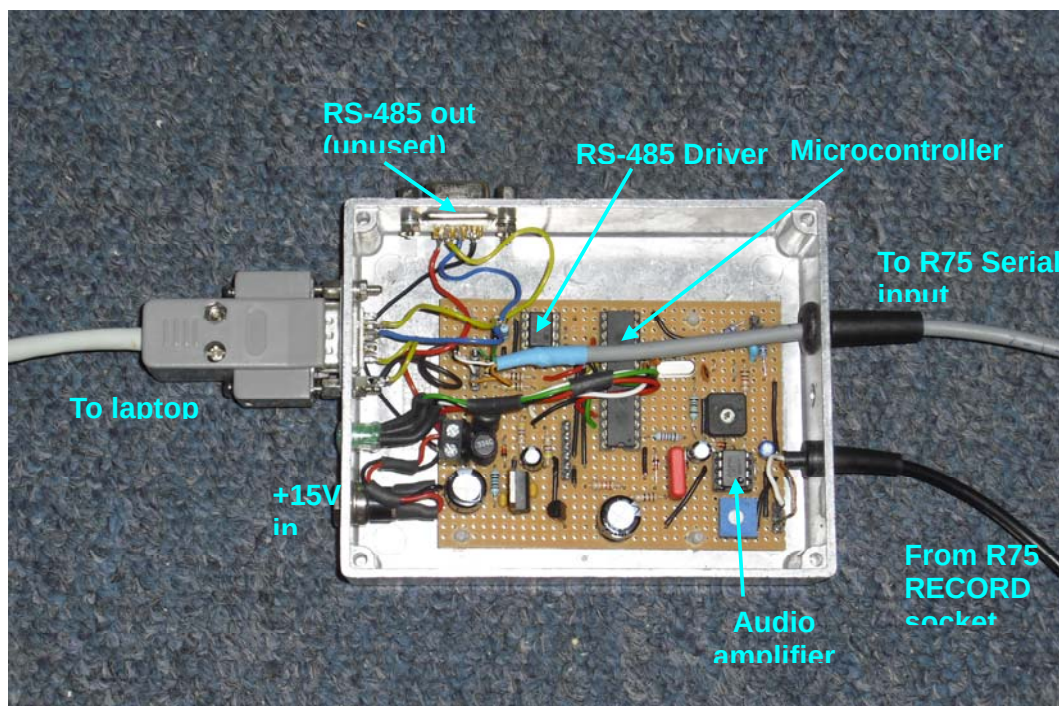


Figure 1 Receiver Controller unit

The basic building blocks of the controller are shown Figure 2. The 'keep-it-simple' approach guided noise level measurement, using the audio signal from the receiver's

A Solar Spectrometer for the 38 MHz band

RECORD output socket, thus bypassing the audio gain control and making it possible to monitor operation via the internal loudspeaker. A simple LM386 audio amplifier chip was used to boost the signal level and provide a pre-settable gain control.¹ This was followed by a Detector/Integrator circuit (courtesy of Colin Clements) with an adjustable time constant for experimentation.

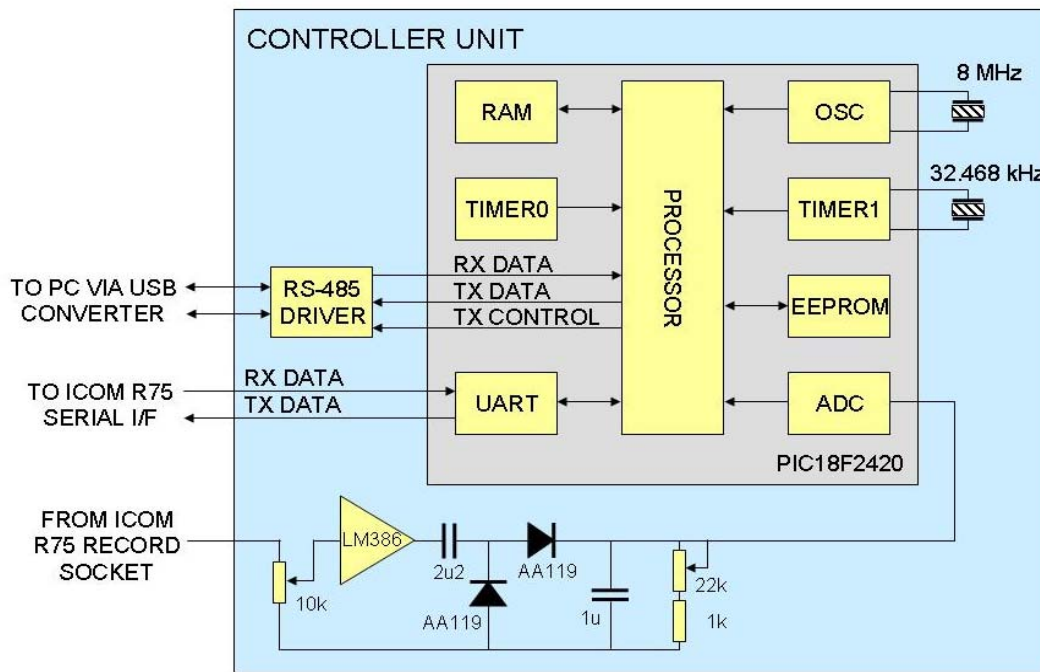


Figure 2 Controller block diagram

Most PIC18 and later devices support a Real Time Clock function which I use to stop and start the scanning process each day at set hours, depending on the month, though the timing accuracy is not good enough for the system to run uncorrected for days on end. The device's 256 bytes of EEPROM are used to store the configuration parameters including the frequencies to be observed. This is sufficient to support 10 kHz frequency steps across the 38 MHz radio astronomy band, which is 750 kHz wide.

A 10-bit Analogue to Digital Converter measures the output of the audio integrator to produce an indication of the noise level at each frequency. I have not made any attempt to relate this value to an actual received signal level as the relative noise levels are sufficient to indicate activity. For simplicity, I only save the most significant 8 bits in the output file.

The on-chip hardware UART is used to support communications with the R75 receiver at 19.2 k baud. In transmit mode it sends the Retune command to the receiver, though I have also used it to power the receiver on and off. In receive mode the UART services the Acknowledgement bytes from the receiver after each Retune has been completed.

The Firmware

The code was implemented in Assembly, as part of learning about these things. The intention is to go back and rework it in C, once I have mastered that! Working in Assembly

¹ See <http://www.national.com/mpf/LM/LM386.html#Overview> for the data sheet including the schematic used (Gain = 20)

A Solar Spectrometer for the 38 MHz band

did help meet the timing constraints imposed by using a bit-banged UART. I included a number of features which I thought would be useful in exploring what was out there, such as the ability to integrate noise over long periods of time (seconds) or average over a number of frequency sweeps, but the results showed me that this was exactly the wrong way to go. You need to sweep as fast and as wide as possible for solar radio bursts, though the functionality might still be useful for future work in observing galactic emissions at these frequencies.

Figure 3 shows the ICOM RS-232 messages. Only two bytes of the Retune command are varied: the least significant Megahertz value and the most significant kilohertz value. These are stored in EEPROM in addresses linked to the Channel Number. The other bytes are fixed values stored in EEPROM (first 7 bytes) or in the program code (last 2 bytes).

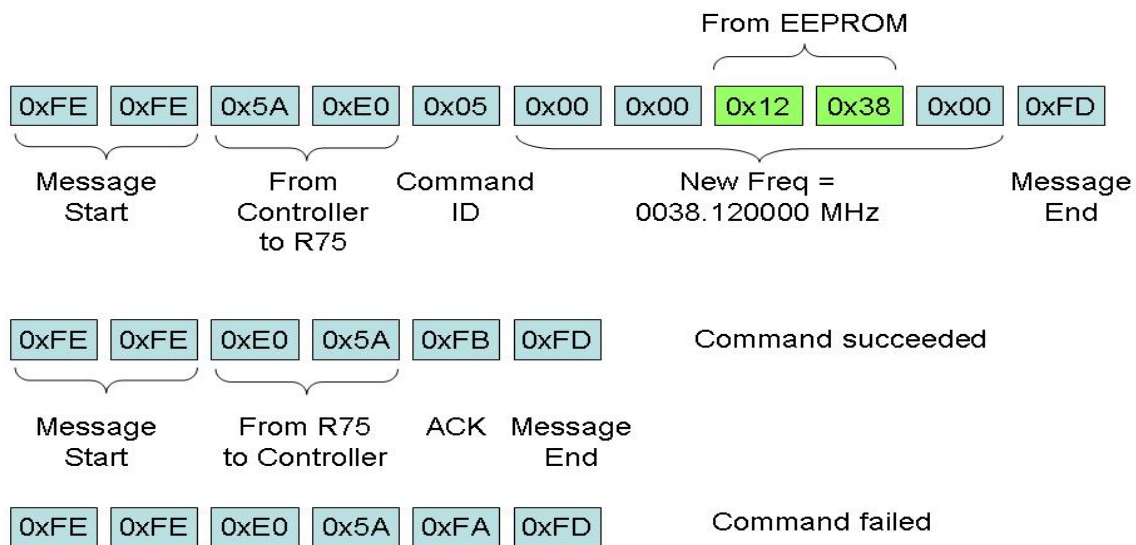


Figure 3 ICOM R75 Retune and ACK/NACK responses

The controller then waits for an interrupt generated by the UART when it receives the Acknowledgement message back from the R75. If the retune fails the application stores a zero value against that channel, increments an error counter and then goes on to the next channel. It would be fairly easy to implement a repeat attempt for a retune failure but I have never had any real problems here.

If an ACK message is confirmed the application goes on to measure the audio output by summing 64 samples taken at 50 us intervals. The number of samples was chosen on the basis that a two-byte accumulator would never overflow with the 10-bit ADC values, and the timing was the fastest that I could go. Both of these values were made configurable so that I could experiment once the system was up and running.

Initial testing revealed a key limitation in the whole approach. The ICOM serial protocol uses an 11-byte command to change frequency. I then need to wait for a 6-byte acknowledgement message before making the audio measurement. The fastest speed option on the serial interface is 19k2, which means that the retune sequence takes at least 9.5 ms. Agonisingly, I found that there is a delay of up to 25 ms in between sending the retune command and the receiver issuing the acknowledgement message. This means that less than 10% of the time is used in obtaining readings.

To try and improve the performance I switched to storing the frequencies to be scanned in

A Solar Spectrometer for the 38 MHz band

suffer unduly from narrow band interference.

The antenna



Figure 5 38 MHz Yagi-Uda antenna

Figure 5 shows the fixed, south-facing, 3-element Yagi used for the receiving antenna, selected on the basis of simplicity and available space. My initial concern was getting sufficient height above ground level, without it becoming too obtrusive for the neighbours. However when I tried modelling different arrangements I found that placing the antenna at 3 metres above ground level produced a single lobe for the vertical gain pattern, as opposed to having several lobes if it was at 10 metres AGL. Multiple lobes would modify the received signal level as the Sun rose and fell during the day. As well as not upsetting the neighbours, the lower height was attractive in being able to use a simple construction using standard garden fencing posts.

The results

I quickly realised that receiving solar radio bursts was not going to be a problem. As I was debugging the micro-controller code one morning, with the receiver on in the background, I heard the noise level gently rise from the depths to a roar, before subsiding again. At the same time I saw the traces from my 3 VLF receivers begin to rocket, and then I noticed a small but distinct change in the magnetometer readings. I knew immediately that an X-class flare had just erupted on the Sun – my only regret was that I didn't have a solar telescope to hand to check what was happening in the optical.

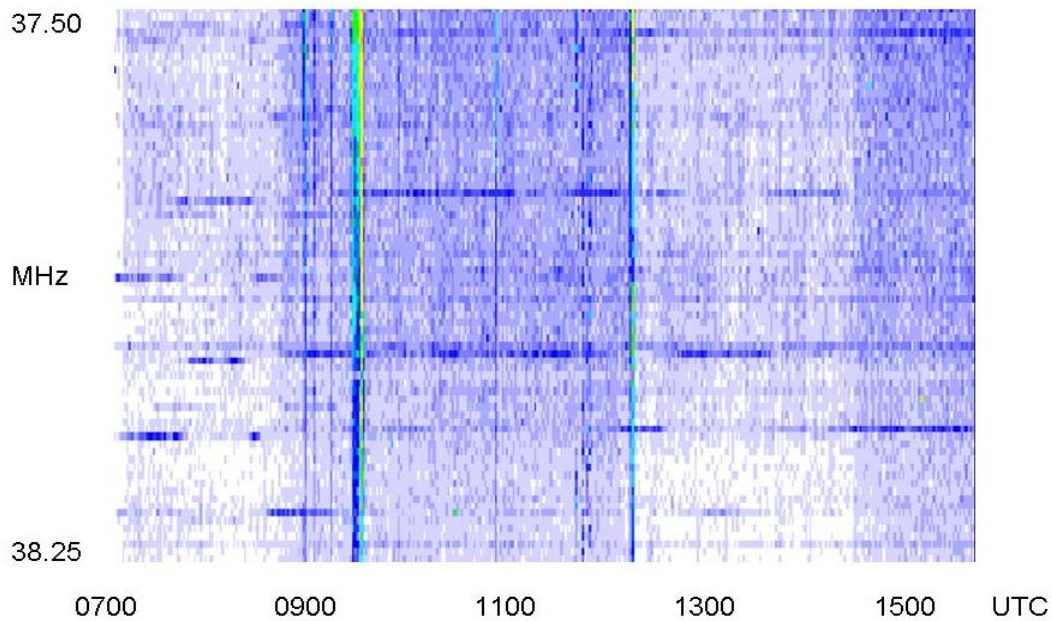


Figure 6 Observations for 2011 October 01

Figure 6 shows the first day's results using the Yagi antenna. The chart shows several terrestrial transmissions which appear as horizontal lines, i.e. at fixed frequencies. These are not a problem since small numbers of offending frequencies can simply be removed from the scanning list. More troublesome is the broadband interference that appears as vertical bands and which cannot be skipped over. The chart shows two types of broadband interference: low-level interference lasting for several hours at a time (e.g. 0800 - 1200) and short duration (less than a minute) but much more intense bursts of noise.

The origin of the low-level interference is unknown; it may be generated by TV IF strips which would explain the sudden onset of the noise and the equally sudden disappearance after a few hours. The high-level bursts could come from a variety of domestic and industrial sources. Further monitoring may narrow down the candidates, but the expectation is that this is something that will have to be lived with.

But is it all 'interference'? Referring to Fig. 6, the burst at about 1140 UTC certainly is, but the burst at 0945 UTC caught my attention, as the SWPC-NOAA daily event report⁴ listed a strong Type V burst at that time. Fig 7 shows this burst in more detail, along with the chart for the same period from the radio spectrometer system at Izmiran, Russia.⁵ This instrument performs daily patrols for the period up to 1200 UTC after which the equivalent Green Bank instrument takes over. I was skipping around the room when I saw this comparison!

4 See <http://www.swpc.noaa.gov/ftpmenu/indices/events.html>

5 See <http://www.izmiran.ru/stp/lars/>

A Solar Spectrometer for the 38 MHz band

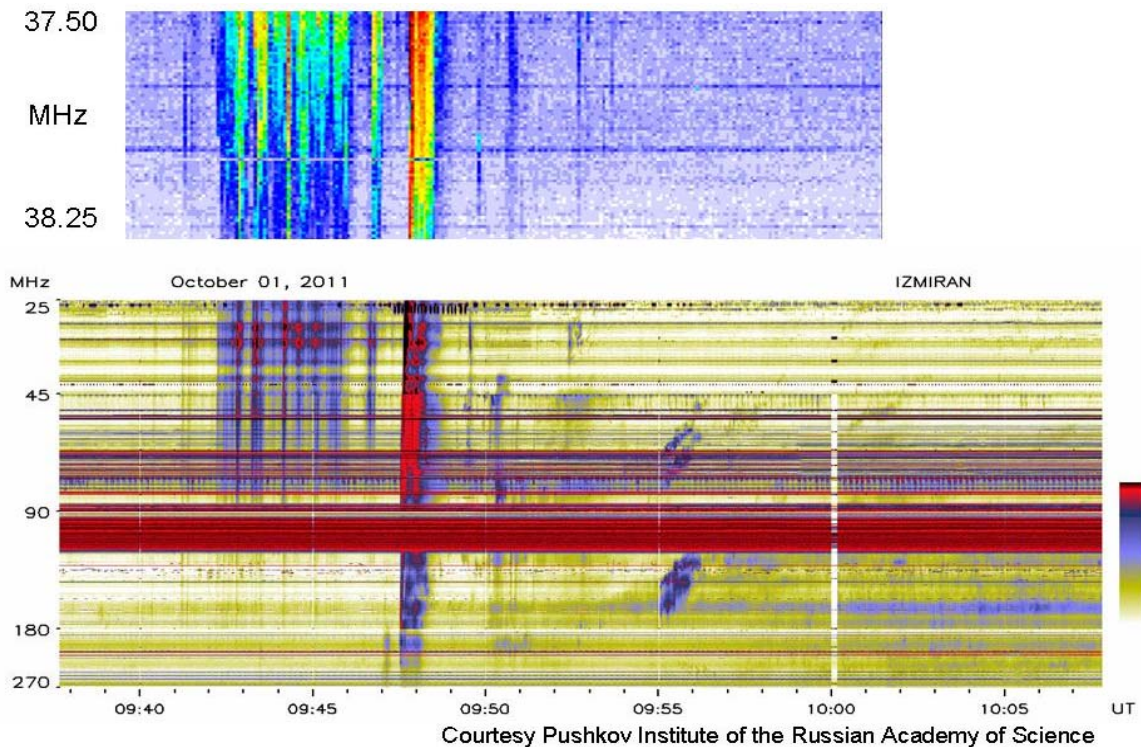


Figure 7 Detail from 2011 October 01 compared with professional instrument

Note that my chart only covers a small portion of the frequency range covered by the Izmiran instrument. The tail-off of signal levels above 38 MHz is believed to be due to my antenna performance, rather than lower noise levels. Note also the tremendous amount of detail that is visible at higher frequencies in the Izmiran plot, including harmonics and the splitting into two distinct bands of emission. If you have a sufficiently quiet environment there is a lot to observe here!

Further development and the FUNcube Dongle

So where to from here? I could install a pre-amplifier at the antenna to increase sensitivity and it would be easy to move from the current 8-bit to 10-bit resolution for the signal amplitude measurements. However I cannot improve the time resolution other than by reducing the number of tuning steps (i.e. reducing the frequency resolution) which is not an attractive trade-off.

A possible alternative would be to use the FUNcube Dongle SDR in place of the ICOM R75. The FCD is based around a commercial digital TV tuner chip, as is used in the e-CALLISTO instrument developed by Christian Monstein and team at ETH Zurich. The documentation for this instrument indicates that the scanning rate is of the order of 2.5 milliseconds per channel, or 15 times faster than the system described here.

A faster scanning rate would support a wider frequency scan at the same time resolution, provided the antenna also has a broad frequency response. Possible solutions are either Biconical or Log Periodic antennas.

The FCD is much cheaper than a communications receiver and its USB interface could make it easier to use, perhaps even with one of the new single-board computers such as the

A Solar Spectrometer for the 38 MHz band

Raspberry Pi. Using low power consumption hardware also raises the possibility of building the instrument as a stand-alone, solar powered remote installation to get away from domestic RF noise. Unfortunately this would require experience of high-level software languages such as Linux/Python, which I do not have.

Paul Hyde