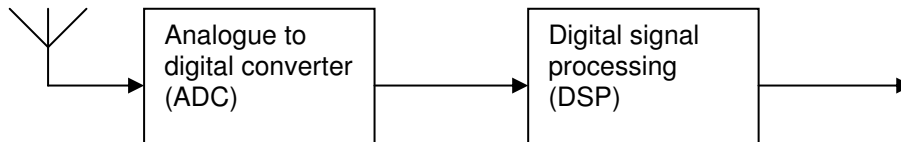


IMPLEMENTATION OF SOFTWARE DEFINED RADIO FOR THE 60kHz MSF TIME SIGNAL

Simon W Day Phasor Design www.phasor-design.uk.com

Introduction

A radio receiver consisting of just antenna, analogue to digital converter and software is possible. This is the notion of Software Defined Radio (SDR) in which filtering and demodulation etc. is implemented either in software or software configurable logic.



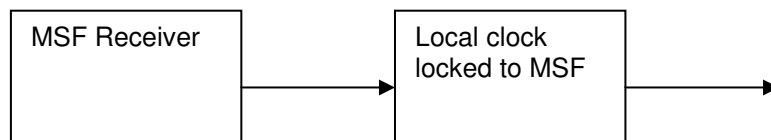
The two key components are shown above. The performance of the ADC is particularly important as this determines many receiver characteristics such as sensitivity (weak signal performance) and blocking (strong signal performance). Essentially, so long as a sufficiently large amount of digital processing power is available, a close to ideal receiver can be constructed.

SDR technology is becoming mature and applications are slowly appearing in, for example, cellular radio base stations. However portable battery powered equipment is a greater challenge due to the power hungry nature of DSP compared to conventional designs. This paper addresses the use of SDR for the MSF time signal at 60kHz. Most time signal receivers in “atomic controlled” clocks run for years on a pair of AA batteries so SDR would not be appropriate. There is a requirement, however, for a high quality MSF receiver to operate from mains power, which has led to the project described here.

The MSF Time Signal

The principal means of disseminating the UK national standards of time and frequency is the MSF signal, which is provided by the National Physical Laboratory. The transmission carrier, from Anthorn in Cumbria, is maintained at 60 kHz to within 2 parts in 10^{12} . Full details are provided by the NPL document CTM001v05 (June 2005) [1]. The propagation is vertically polarised ground wave, which is very reliable for 1000km or so, although at greater ranges multi-path fading, caused by the sky wave, may occur at night.

Analysis of this document shows that the basic symbol rate is 10 Symbol/s with digital ‘1’ represented by carrier off and ‘0’ for carrier on. The rise and fall times are not defined except that the document says that these are what result from the “combination of antenna and transmitter” - an oscilloscope shows fairly short rise and fall times. The time information is transmitted in a frame over one minute and relates to the following minute. There is a large amount of redundant information so that it is straightforward to detect corrupted frames and reject them. The implication is that a MSF receiver contains a local clock, which is locked to the MSF signal when it is available and free runs at other times. This situation is illustrated below. Interruptions to MSF may be caused by local noise or by planned maintenance of the transmitter.



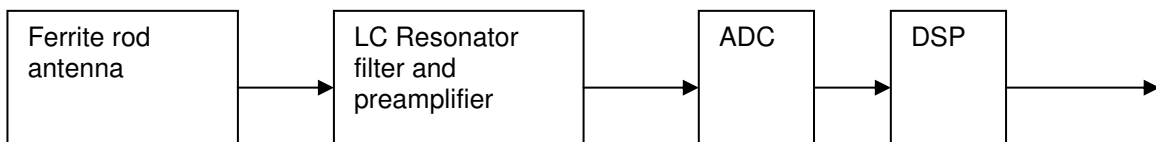
A symbol rate of 10 Symbol/s suggests that the receiver bandwidth should be in the range 10 to 20Hz. In low cost units these bandwidths are obtained by using a crystal as in the CMMR-8D-MF [2] where a single pole resonator is created by tuning out the parallel capacitance with an anti-phase capacitor. It was felt that a high quality receiver should have a more sophisticated filter, with better rejection and pulse shape characteristics. In the absence of any other viable filter technology SDR became the method of choice.

Software Defined Radio: ADC

A DSP processor including an on-chip dual 12-bit ADC, is available in the form of the DSPIC33FJ250GP710 [3]. This is a 16-bit wide data device capable of 40MIPS. The 12-bit ADC specification appears to provide a dynamic range of 72 dB, but in practice a number of factors affect this:

- Internal digital filtering increases the effective number of bits;
- External band limiting filtering can be applied to lower the total signal power applied to the ADC;
- Truncation of the data during digital processing was found to be necessary to maintain 16-bit fixed point arithmetic.

The MSF signal is fixed in frequency and so variable tuning was not a requirement. The simple solution was therefore to restrict the bandwidth applied to the ADC through the use of a LC resonator achieving a bandwidth of about 800Hz. The usual form of antenna for 60kHz is a ferrite rod coil, which is resonated with a capacitance of about 6800pF. The resulting bandwidth of around 600Hz also provides narrow band filtering.



This arrangement also allows for an external vertical monopole type antenna to be used where a ferrite rod is unsuitable. The use of a pre-selector filter could be labelled as an Assisted Software Defined Radio (ASDR) where the input frequency range is limited to avoid the need for a very high specification ADC. The ASDR closely parallels the very traditional TRF (Tuned Radio Frequency) receiver.

Software Defined Radio: Tuning

Straightforward generation of a quadrature local oscillator (LO) in DSP form is achieved by selecting a sample rate, which has a very simple arithmetic relationship with the received frequency. This logic suggests a sample rate of 240kS/s where the LO sequences are as follows:

I	0	+1	0	-1	0	+1	0	-1	0	+1	0
Q	-1	0	+1	0	-1	0	+1	0	-1	0	+1

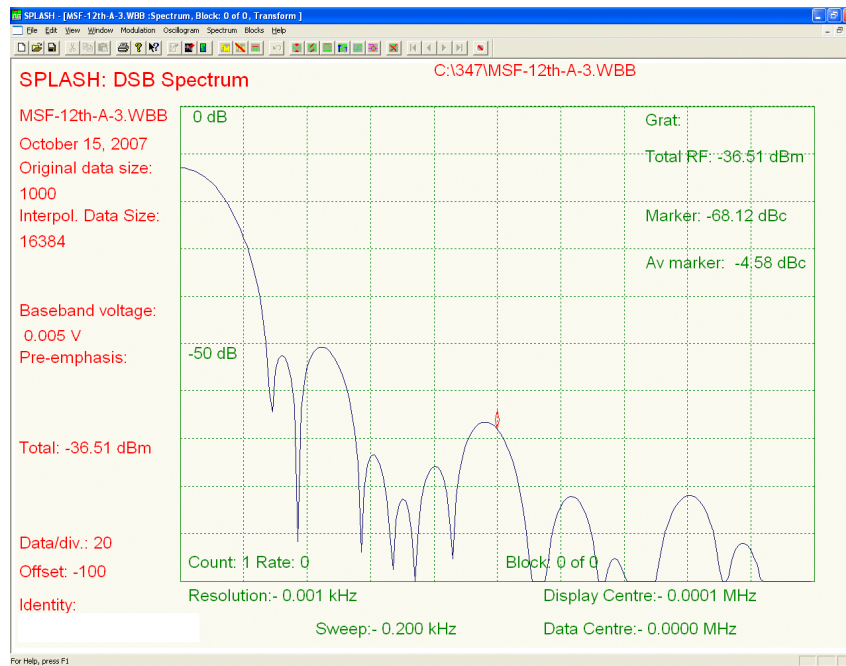
In practice this notion was found not to work well as the processor time to service the interrupt associated with the ADC consumed a large proportion of its capacity. A simple alternative was to sub-sample at 48kS/s giving an aliased frequency of 12 kHz which again requires the LO sequence tabled above. The use of the pre-selector filter makes this option feasible.

Software Defined Radio: Filter characteristics

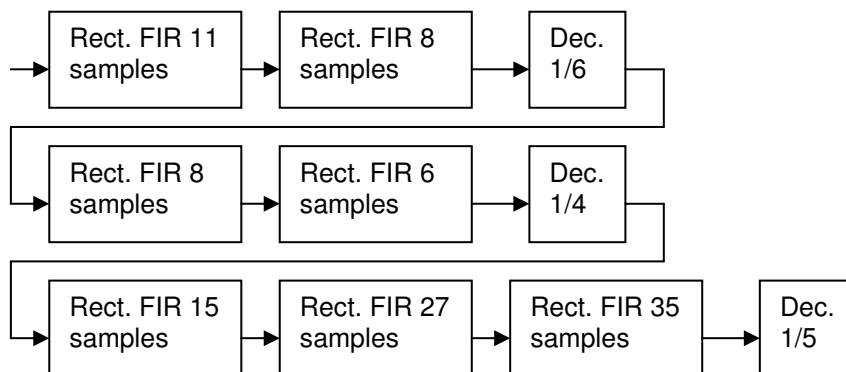
Some time was spent evaluating the various filter methods. The issues are:

- Type of DSP arithmetic – floating point – fixed point – bit widths – effect of truncation and overflow;
- Processing power/ time available;
- Pulse shape desired;
- Stop band performance – leakage at weak frequencies;
- Decimation strategy.

To model these issues use was made of the SPLASH software [4].

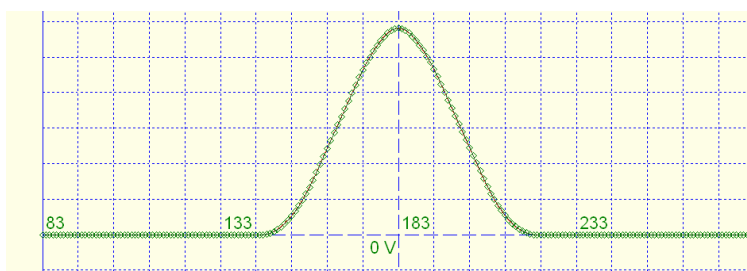


Eventually it was decided to use the structure shown below where “Rect. FIR” means rectangular finite impulse filter (that is equal value weights of unity) and “Dec.” means decimate. Filters of this type are essentially moving average filters.



Rectangular finite impulse filters can be implemented very efficiently in software using an algorithm that adds the new sample and subtracts the oldest sample each time an output is needed. Multiply-accumulate (MAC) units are not required.

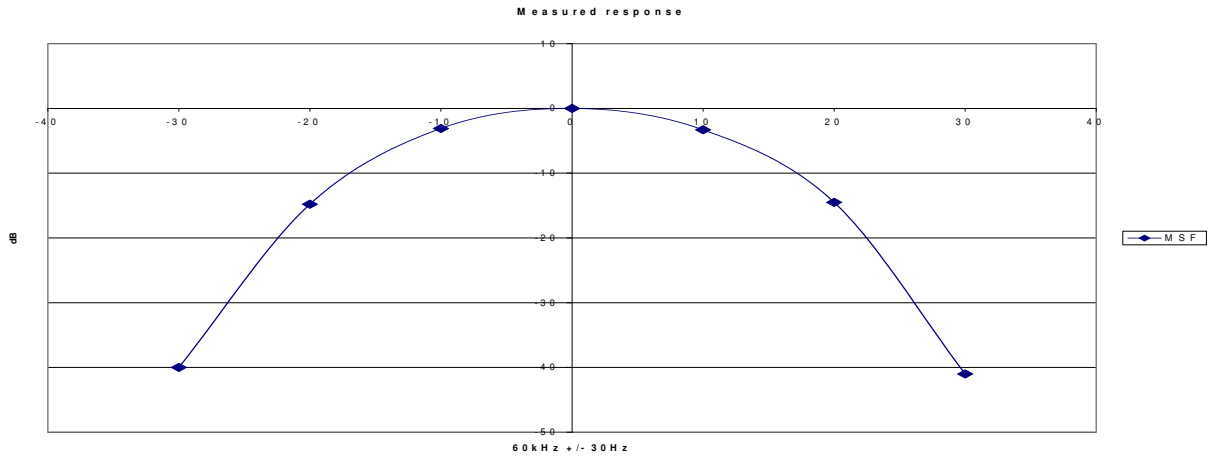
The effect of cascading moving average filters in this way is to produce a close approximation to a Gaussian pulse response in the time domain; in the frequency domain the zeros can be distributed to best effect, as illustrated in the plot above. This method was used in preference to the conventional Cascaded Integrator-Comb (CIC) technique (also known as the "Hogenauer filter").



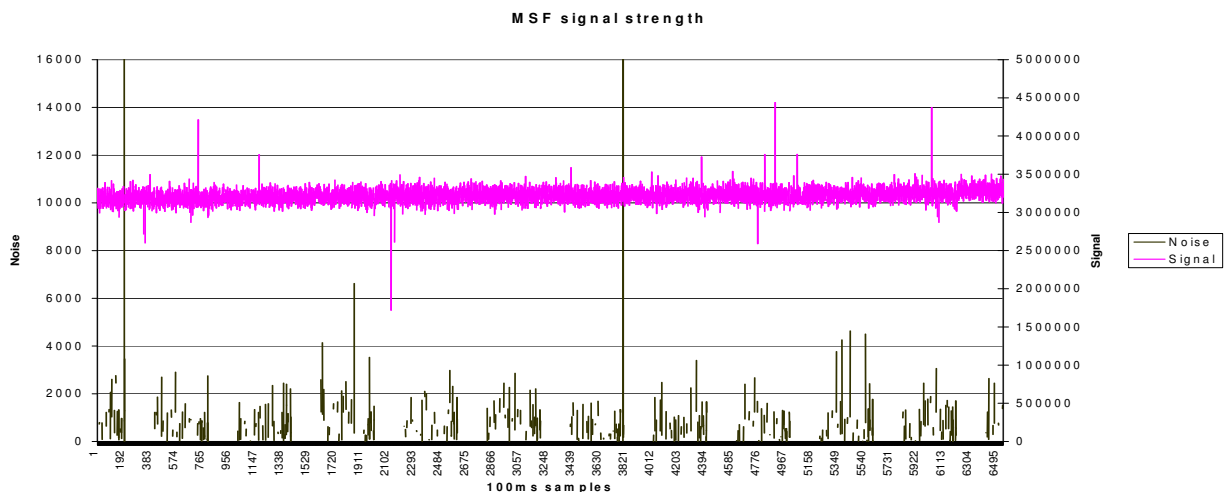
The simulated impulse response, showing the close to Gaussian shape, is plotted here. The absence of inter-symbol interference can be checked in this plot.

Software Defined Radio: Measured results

To verify the performance of the software filters, a signal generator was used to stimulate the receiver at 10Hz intervals. The software itself measured the power level of the filter output and calculated the dB level using floating point arithmetic in relation to an arbitrary reference point.



The ability to include code to measure and log parameters of the received signal was found to be very useful. The on-off keyed nature of the MSF signal allows direct measurement of the background noise during the 'off' symbols. In the following plot the signal and noise levels in terms of a power count are recorded over about ten minutes. The occasional noise spike can be seen.



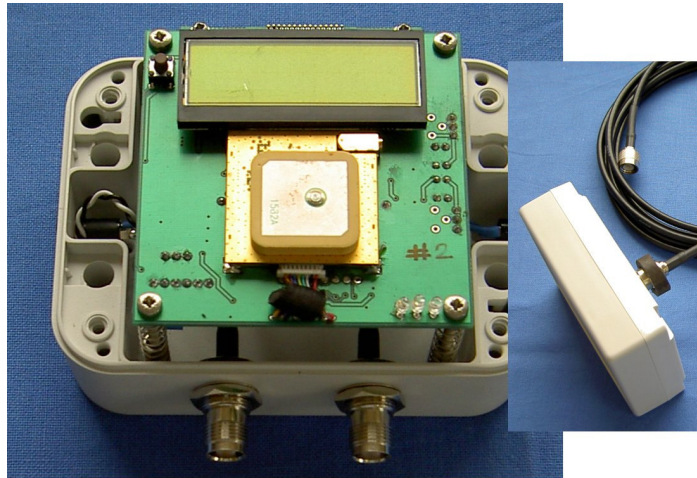
Processor time arrangement

The basic resource being manipulated is processor time. It was found necessary to carefully plan the use of processor time so that all tasks were completed in a timely way and a 'slotted' approach was found to be useful. In experimental terms, spare processor pins were used as flags for various tasks and the waveforms at these pins, when viewed on an oscilloscope, formed a way of visualising the loading of the processor.

Interface to GPS

As part of the overall system the MSF receiver accepted a second source of precise timing in the form of the GPS-1PPS signal. The internal TCXO of the MSF receiver was locked to drive the phase of the received MSF signal to a selected point. It was therefore possible to compare the phase of MSF with the time information in the GPS-1PPS. We were reassured to find that MSF phase was completely stable under these conditions.

The MSF output pulse was also found to be in time alignment with the GPS-1PPS signal once the 160ms of delay within the MSF receiver was accounted for. As a result of this investigation the MSF receiver timing output was corrected for its own delay.



The dual channel receiver with separate antenna is shown above. The GPS antenna is visible in the centre.

Conclusions

Software defined radio technique has been applied to the case of a 60kHz time signal receiver. The low radio frequency and narrow bandwidth made the hardware requirement fairly straightforward and implementation in a single chip 16-bit processor with on-chip ADC was achieved. In fact the receiver implemented was a dual channel type with two independent receivers for diversity reception.

The expected benefits of SDR were gained and specifically the filtering was largely performed in software. The absence of any suitable alternative filtering technology made this critical. An additional unexpected benefit was the capability to code measurement facilities; these were arranged to monitor a wide range of receiver parameters and log these over time.

The requirement to carefully plan the processor time usage became fully appreciated as the project progressed.

Acknowledgements

This work was carried out under contract to Navtech Systems Limited [5] and their kind permission to present the work is gratefully acknowledged.

References

- [1] CTM001v05 (June 2005) <http://www.npl.co.uk/time/msf/ctm001v05.pdf>
- [2] CMMR-8D-MF http://www.c-max.com.hk/docs/cme8000/CMMR-8D-A5_1.pdf
- [3] DSPIC33FJ250GP710 <http://ww1.microchip.com/downloads/en/DeviceDoc/70286A.pdf>
- [4] SPLASH Version 6.4 <http://www.phasor-design.uk.com/software/splash.htm>
- [5] Navtech Systems <http://www.navtechsystems.co.uk>