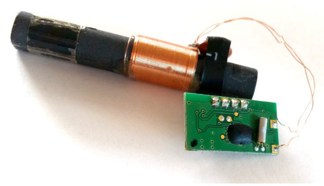
The EM2S 60kHz radio receiver module and the MSF time signal

Deirdre O'Byrne, 6th February 2018

The EM2S module

The EM2S module is a cheap[1] 60kHz AM radio receiver module. It is designed to receive the MSF time signal[2] transmitted by the National Physical Laboratory (NPL) from Anthorn in north-west England.

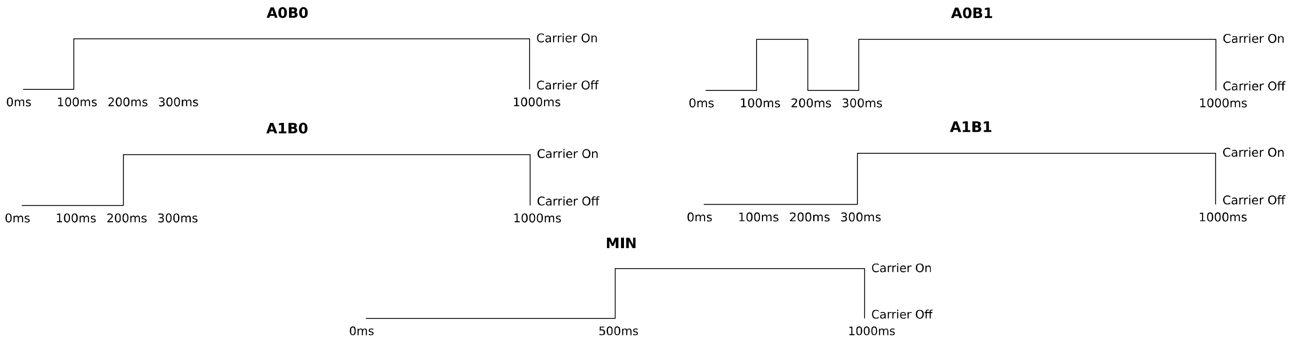


The MSF time signal consists of an accurate 60kHz carrier wave, which is turned on and off to broadcast time information. The NPL publishes[3] details of how to interpret the signal.

In this paper I investigate how well the EM2S module receives this signal, with a view towards designing a noise-tolerant decoding algorithm.

The 5 different MSF signal waveforms

Each MSF signal waveform lasts for one second. The waveform starts at the start of each UTC second with the 60kHz carrier being turned off for 100ms, 200ms, 300ms or 500ms. For the purposes of this paper, the 5 different MSF signals are called “A0B0”, “A0B1”, “A1B0”, “A1B1” and “MIN”, and the waveforms transmitted by NPL are -



The EM2S module has an open collector output which is connected to ground when the carrier signal is not detected. Most EM2S decoding algorithms seem to work by detecting the relative times of the falling and rising of the carrier, which appear as falling and rising edges of the output of the module. However, this approach has two problems -

1. The edges do not usually occur close to their expected locations. Algorithms tend to give up to 40ms leeway.
2. Signal noise appears in the output signal of the EM2S modules as short-lived spikes – usually of less than about 15ms duration each. Algorithms have to accommodate these spikes.

---

1 Approx €20 for a module plus antenna at the time of writing - https://www.hkw-shop.de/Empfangstechnik-AM/Empfangsmodul-EM2S-60-kHz.html?listtype=search&searchparam=em2s

2 http://www.npl.co.uk/science-technology/time-frequency/products-and-services/time/msf-radio-time-signal

3 http://www.npl.co.uk/upload/pdf/MSF_Time_Date_Code.pdf

Experimental setup

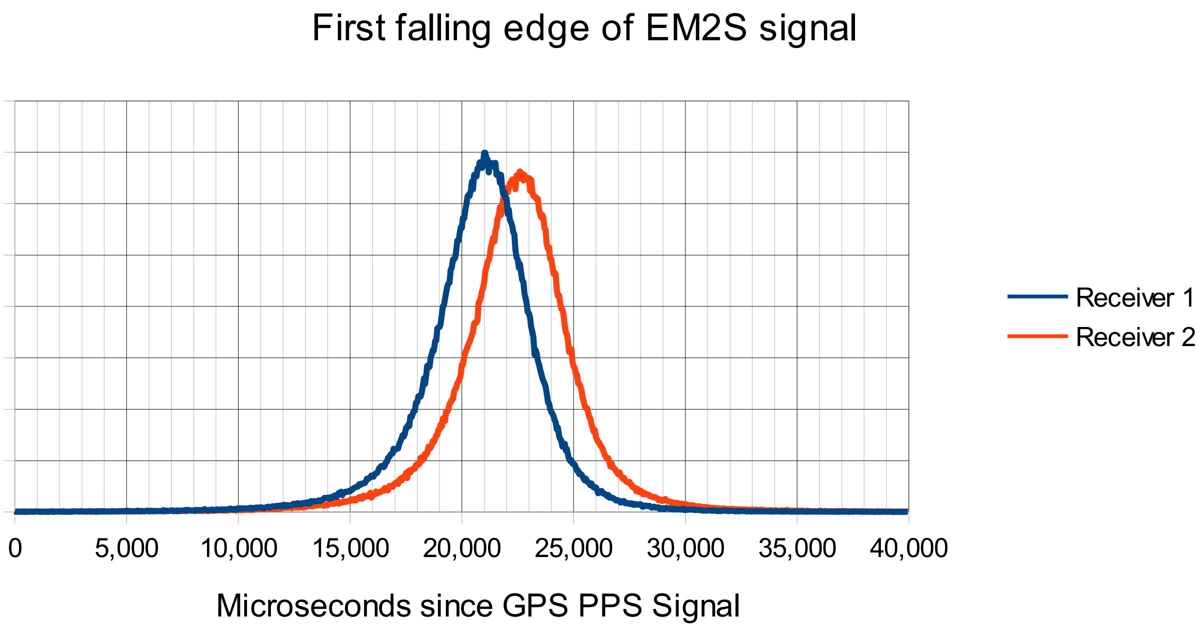
I attached the output of two EM2S modules, and the PPS output of a GPS receiver, to a logic analyser running at 20kHz. This logic analyser was connected to a computer, which logged the raw data to a disk. Analysis was done on the data with custom software.

The experiment was conducted in the midlands of Ireland, near 53.7°N 7.8°W, and lasted from approximately 2018/01/28 22:57:54.58 UTC for approximately 8 days, 4 hours, 17 minutes and 14.26 seconds.

The software was written with knowledge of which UTC second each GPS PPS signal represented, and was thus able to determine which of the 5 MSF signals should occur during each second.

Experiment 1 – EM2S timing signal accuracy

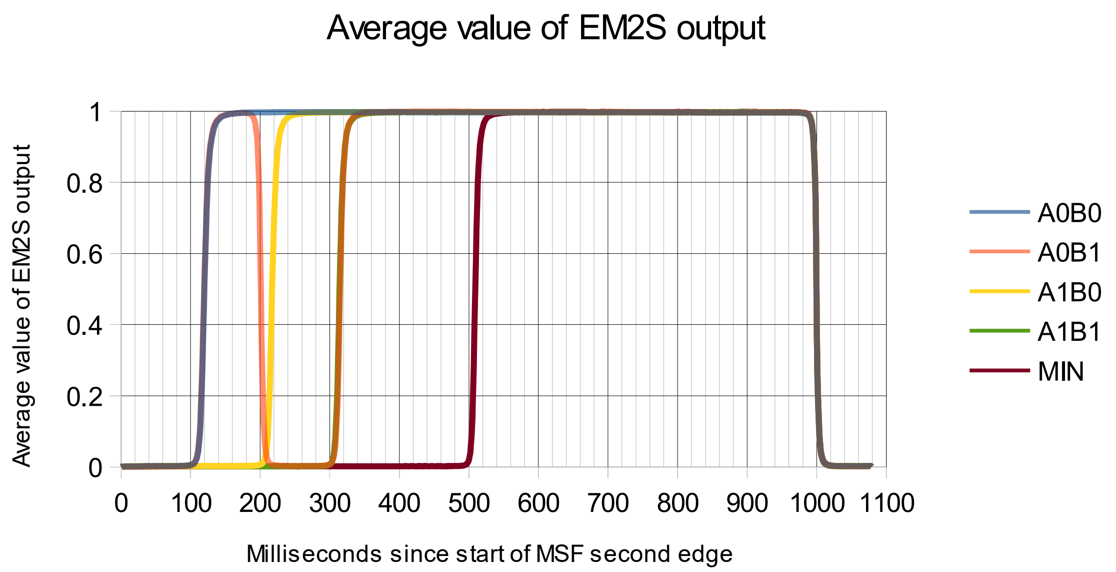
For the first experiement, I wanted to find out how accurately the EM2S receivers detected the loss of the MSF carrier at the start of each second. The analysis was to measure the delay between the GPS PPS signal and the first falling edge of the output of the EM2S receivers.



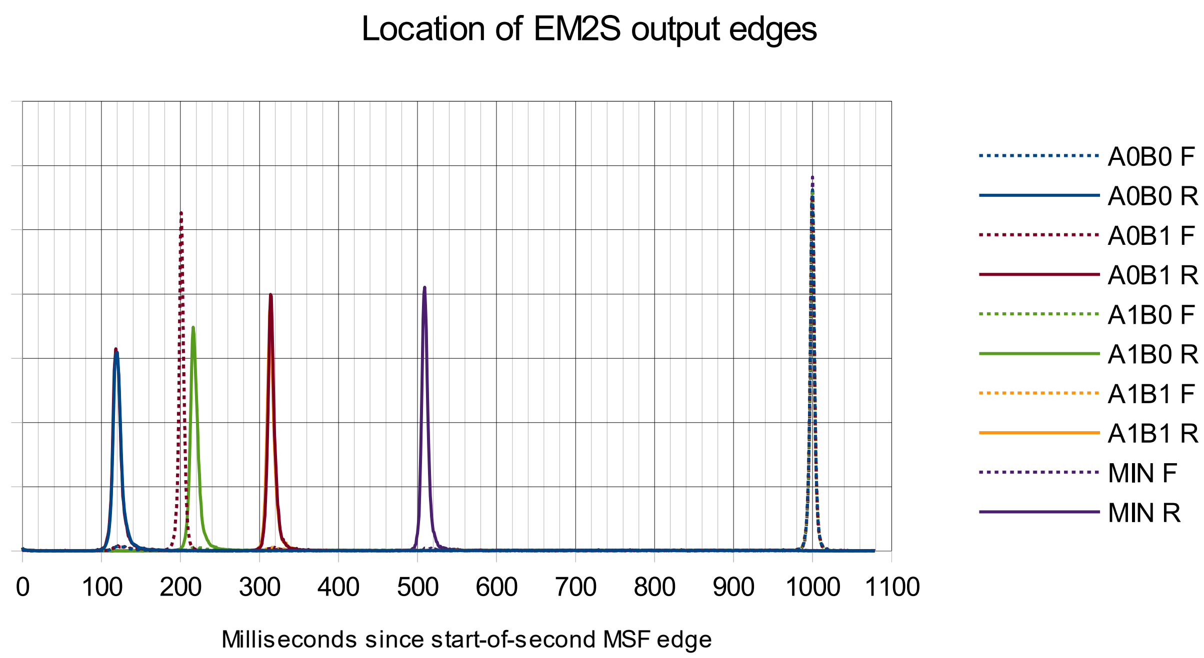
This graph shows a histogram of the locations within the UTC second of the first falling edge of the EM2S output signal. It is observed that receiver 1 detects the loss of the 60kHz carrier on average 20.7 milliseconds after the start of the UTC second, and receiver 2 detects it on average 22.3 milliseconds after. These two receivers were within 20cm of each other at all times.

Experiment 2 – average value of the signal

For the second experiment, I measured the value of the EM2S output for each of 1,080 millisecond intervals from the first detection of the falling edge of that output after the GPS PPS signal.



This graph shows the average value of the EM2S output signal during each of the 5 MSF second waveforms. It is observed that rising edges tend to be "late", whereas falling edges tend to be closer to where they are supposed to be. Also, there is about 20ms - 40ms of spread in the location of the edges. This can be seen more clearly in the following histogram -



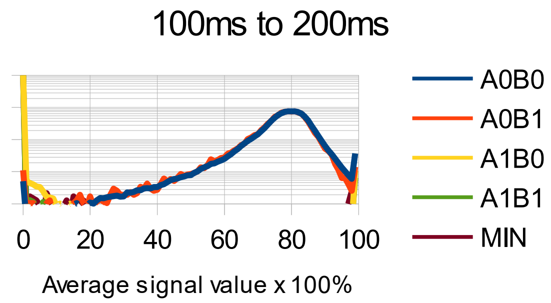
It is observed that each rising edge has noticable falling-edge noise associated with it (the dotted lines below each solid line). Indeed examination of the raw data shows that rising edges often have noise spikes close to them.

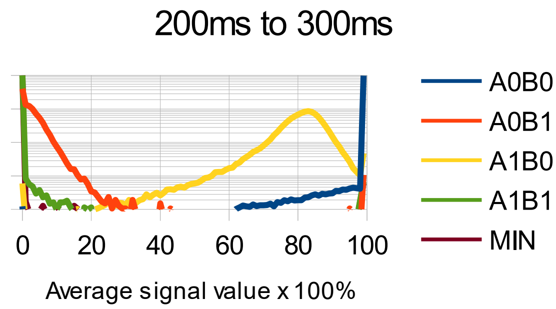
A decoding algorithm

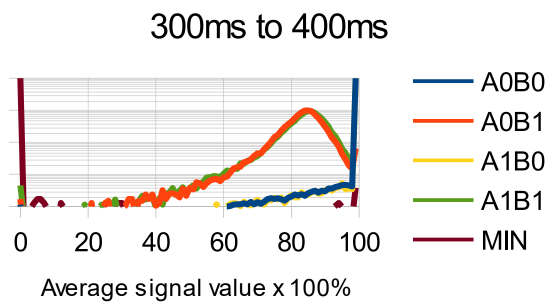
Decoding algorithms based on the location of rising and falling edges are problematic. The edges are not where they are supposed to be, there is significant spread in their observed location, and there is significant noise associated with rising edges in particular. Noise is also occasionally observed between the edges, and accounting for this noise is not trivial in a decoding algorithm.

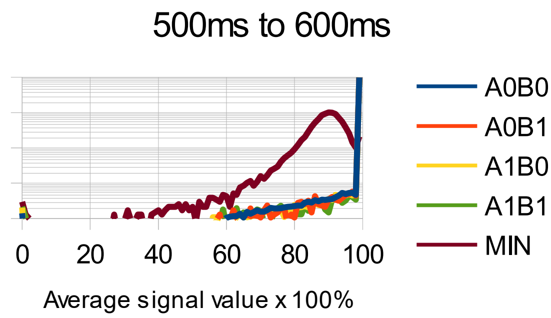
Decoding algorithms based on taking many samples of the signal at regular intervals seems to be a more resilient approach. The inherent inaccuracy of the EM2S modules is such that sampling at 1kHz is more than adequate, and even a sampling rate no faster than 500Hz should be sufficient. The question becomes – how many of the observed samples in each 100ms interval need to be of a given value for the overall signal to be deemed to have that value during that interval?

The key intervals are 100ms to 200ms after the start of MSF second edge, 200ms to 300ms, 300ms to 400ms, and 500ms to 600ms. The corresponding histograms are presented below -







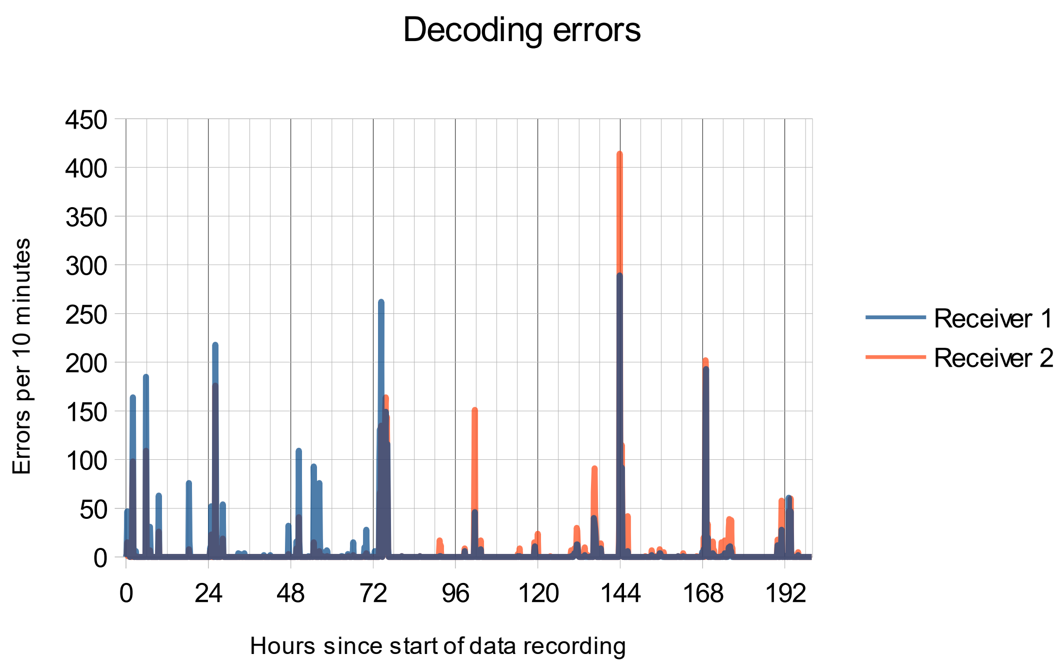


The trickiest problem is deciding between A0B1 and A1B0 in the 200ms to 300ms interval. As can be seen by the graphs, if the average signal level is less than about 25%, the carrier should be considered to be off, and if it is more than 25%, the carrier should be considered to be on. Using this algorithm, we get the following result -

	Decoded signal					
Actual signal	A0B0	A0B1	A1B0	A1B1	MIN	Unknown
A0B0	940,122	119	1,815	75	1	5,263
A0B1	88	46,663	5	108	0	248
A1B0	175	9	331,040	605	1	1,644
A1B1	7	10	21	60,956	6	353
MIN	5	0	5	4	23,416	116

There are relatively few unknown signals, and few misidentified signals.

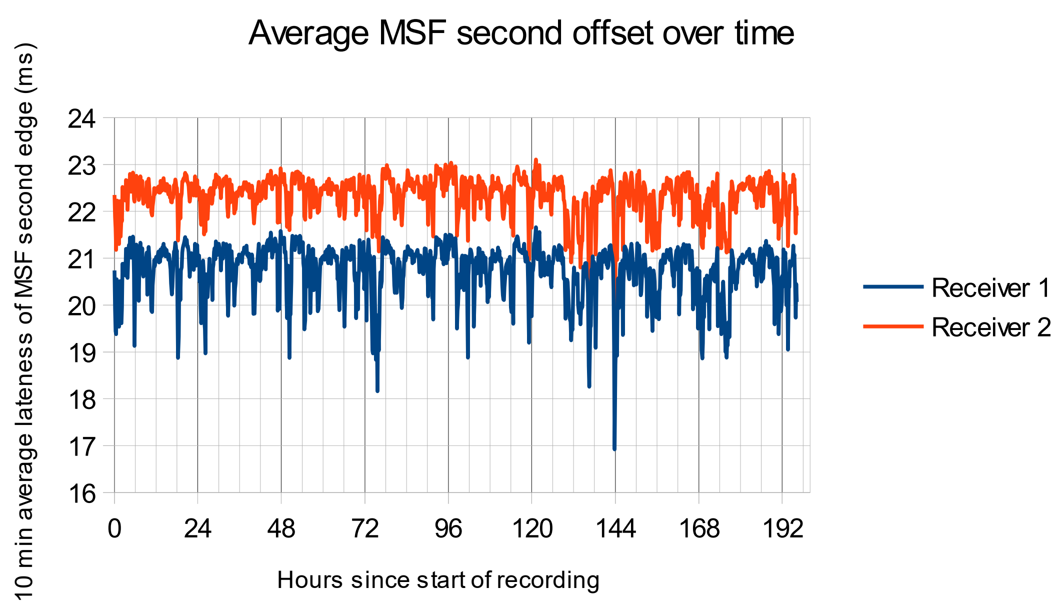
The unknown signals, and the signal errors, appear to come in groups as shown in the following histogram -



There also appears to be something of a 24-hour periodicity to some of the errors.

The EM2S modules and their antennas were placed in an upstairs bedroom near the floor and a 3D printer (which was turned off and powered down). Hence it is probably the case that a large number of the errors are due to poor antenna location.

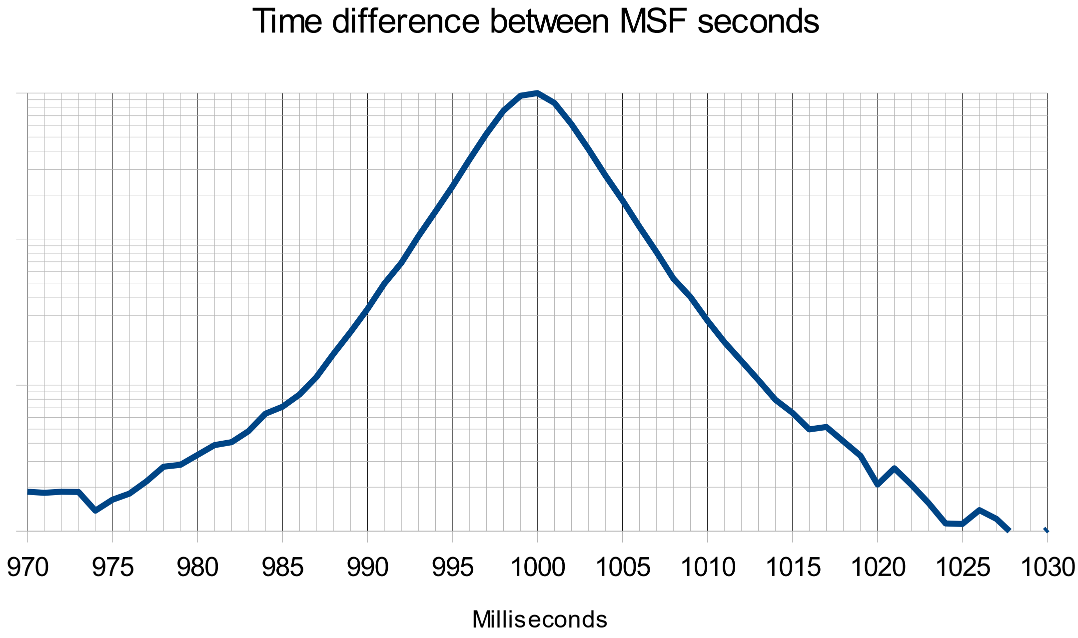
The graph of the difference between EM2S/MSF and UTC seconds over time is presented below. Note that the average difference is going to be massively affected when the EM2S is spitting out just noise -

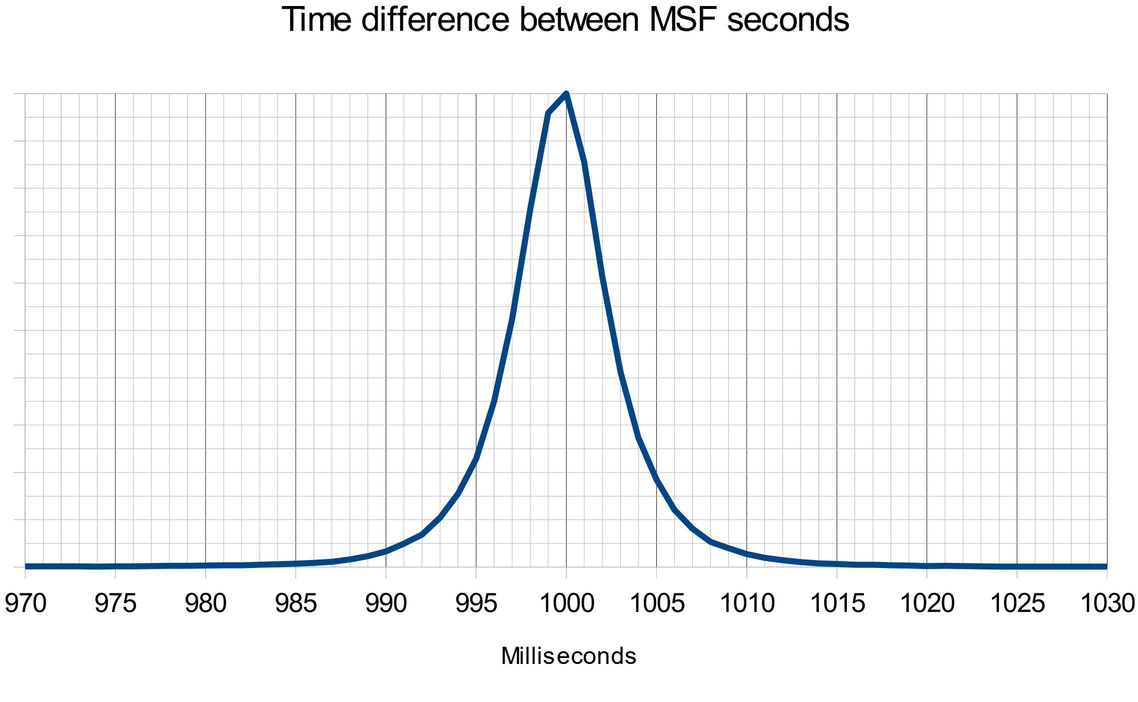


Start of second detection

Algorithms for detecting the start of the second are not considered by this paper, as the algorithms in use by my analysis have access to the start of the second through the GPS PPS signal.

Histograms of the time difference between successive start of MSF second edges are presented below in the hope they will be useful.





The first histogram has a logarithmic scale, and the second is the same data with a linear scale.

Other decoding algorithms

Research was also conducted on an algorithm which splits the second into intervals from 0ms – 80ms, 160ms – 180ms, 260ms – 280ms, 360ms – 400ms, 400ms – 480ms, 560ms – 600ms, 600ms – 700ms, 700ms – 800ms, 800ms – 900ms, 900ms – 980ms. This was found to not perform as well as the algorithm presented above.

A small improvement can be obtained if a threshold of 15% is used for every 100ms interval except the 200ms – 300ms interval, which should remain at 25%. When this algorithm is used, we get -

	Decoded signal					
Actual signal	A0B0	A0B1	A1B0	A1B1	MIN	Unknown
A0B0	941,301	150	984	68	1	4,891
A0B1	89	46,731	3	69	0	220
A1B0	264	17	331,108	621	0	1,464
A1B1	11	14	18	61,016	5	289
MIN	8	3	4	5	23,410	116

The data, and the analysis code, used in this project have been made available on github[4]. The reader is welcome to try their own ideas for a decoding algorithm, as well as to test their production algorithm against a large dataset.

Future developments

It is noted that there are many disallowed patterns in the signal. For starters, the waveform for each second should broadly correspond to one of the 5 nominal waveforms given above. The following table lists the actual decoded pattern which was seen in the top 20 misidentifications of each misidentified signal. The patterns are the carrier absent ("0") or present ("1") during each 100ms interval of the second -

Top mis-identified patterns

A0B0 (01111111)		A0B1 (01011111)		A1B0 (00111111)		A1B1 (00011111)		MIN (00001111)	
Pattern	Instances	Pattern	Instances	Pattern	Instances	Pattern	Instances	Pattern	Instances
1111111111	2105	0111111111	89	0001111111	621	0000111111	57	1111111111	22
0011111111	984	0001111111	69	1111111111	466	1111111111	54	0000111111	19
1111111110	467	1111111111	67	0111111111	264	1001111111	23	0111111111	8
0111111110	214	0111111101	33	1011111111	105	0011111111	18	0000001111	8
1111111101	197	1101111111	32	1111111101	79	1101111111	14	0100111111	5
0101111111	150	0100111111	12	0011111110	65	0101111111	14	0001111111	5
1111111011	131	1011111110	10	1111111100	48	0111111111	11	0011111111	4
0111101111	124	1111111110	8	1111111110	34	1111110011	11	1000011111	4
0111111011	116	0000111111	5	1111111011	33	1111111001	10	1101011111	4
0111110111	111	0101111110	5	1111111001	32	0001111110	9	0000011101	4
0111111101	105	0111111100	4	0011111011	32	0001111011	9	1101111111	3
1111110111	91	0101111011	4	0011110111	31	1011111111	7	0101111111	3
1111101111	84	0011111111	3	0011111101	30	1111111000	7	0001011111	3
0110111111	82	1001111111	3	0011101111	27	1111001111	6	1110101111	3
0111011111	81	0101101111	3	1111101111	25	0001011111	6	1001111111	2
1101111111	76	0101110111	3	1101111111	23	0001110111	5	1000111111	2
1111111100	76	1011111111	2	1001111111	23	0000011111	5	0100011111	2
0001111111	68	1110111111	2	1111110111	23	1111111100	4	1100011111	2
1011111111	62	0101111101	2	1110111111	22	0001101111	4	0011111101	2
1110111111	47	0001111101	2	1111110011	21	0011111110	3	1111101011	2

The NPL-published algorithm for determining the time from the received signal gives another way of detecting (and hence possibly correcting) received errors. There are many restraints which follow from that algorithm. For instance, the date and day-of-week should remain the same from minute to minute unless the time-of-day goes from 23:59 to 00:00. Such an algorithm was baptised the "blame algorithm" by Poul-Henning Kamp of the time-nuts mailing list[5]. A blame algorithm in conjunction with study of the table above may generate further improvements in an overall decoding algorithm. For instance, each pattern could be assigned a set of confidences that it should be decoded as a certain value, and the blame algorithm could be used to solidify those confidences into a most likely decoded signal. And the confidences themselves could be informed by the average signal level observed during the 100ms intervals – an observed average of 20%, for instance, could easily be either a "0" or a "1".

4 https://github.com/deirdreobyrne/MSF-EM2S

5 https://www.febo.com/pipermail/time-nuts/2018-February/108775.html

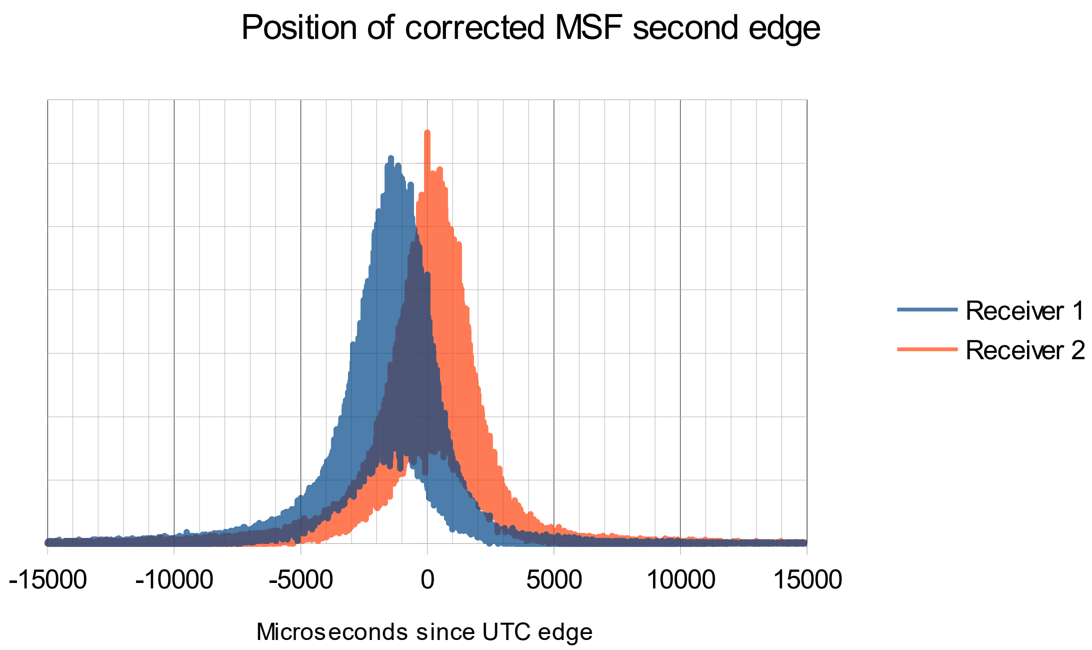
Towards more absolute accuracy

It turns out[6] that there is a correlation between the time interval between second markers and how late those markers are relative to UTC. Performing the calculations on the data set, we get -

	Slope	Constant	Correlation
Receiver 1	0.4871	-466,360us	0.671
Receiver 2	0.4829	-460,620us	0.675

i.e. the second edge is late by an amount of approximately 0.48 times the second length minus 465,000 microseconds, with a correlation of about 67%.

When we apply the *same* correction to *both* signals, we get -



6 https://www.febo.com/pipermail/time-nuts/2018-February/108804.html