An Efficient Correlator for Implementations of BBC Jam Resistance

Leemon C. Baird III
William L. Bahn

Academy Center for Cyberspace Research
United States Air Force Academy

USAFA-TR-2009-ACCR-02

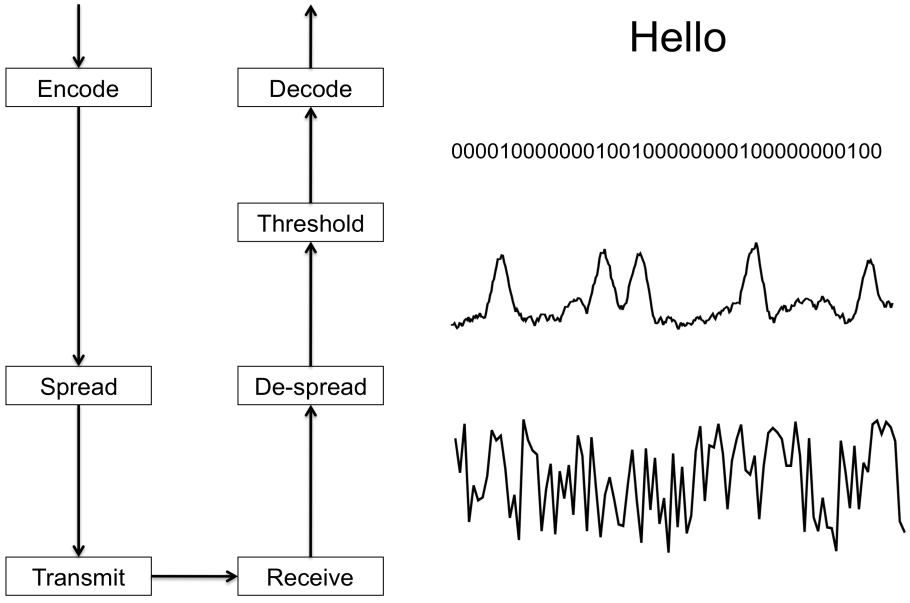
November, 2009

Abstract

This paper describes the construction of an efficient matched filter designed as part of a BBC jam resistant system. It is an Efficient Golay Correlator (EGC) with very long impulse response (thousands or millions of chips), using random parameters. Simulation results show very good autocorrelation and power spectrum, and suggest this could be used in building a complete BBC system.[1]

Background

A wireless, jam-resistant, communication system based on BBC (Baird, Bahn, Collins) coding [1] could be organized in this way:

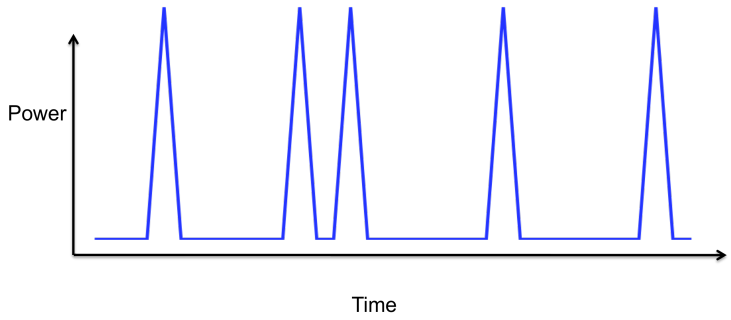


[1] This work was sponsored in part by the Air Force Information Operations Center (AFIOC), Lackland AFB, TX, and was performed at the Academy Center for Cyberspace Resaerch (ACCR) at the United States Air Force Academy.

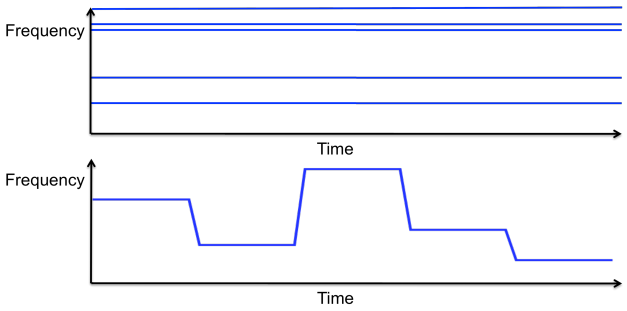
An original message ("Hello") is BBC encoded to give a codeword that is mostly 0 bits with a few 1 bits. A spreading operation converts this codeword into a waveform that spreads each 1 bit in both time and frequency. The waveform is transmitted through a wireless channel, and received at the other end. The receiver can then de-spread the signal, apply a threshold to the result, and recover the codeword. Finally, the codeword is decoded using the BBC algorithm.

For any given implementation, there are many choices for spreading and de-spreading the codeword. For example, a simple pulse-based system could transmit a short burst of high-power radio noise for each 1 in the packet, and transmit nothing for each 0.

0000100000001001000000001000000000100



However, this gives an enormous peak-to-mean power ratio, which requires amplifiers that are expensive, complex, or both. An alternative is to encode the packet as a set of pure sine waves, either transmitted simultaneously or sequentially:

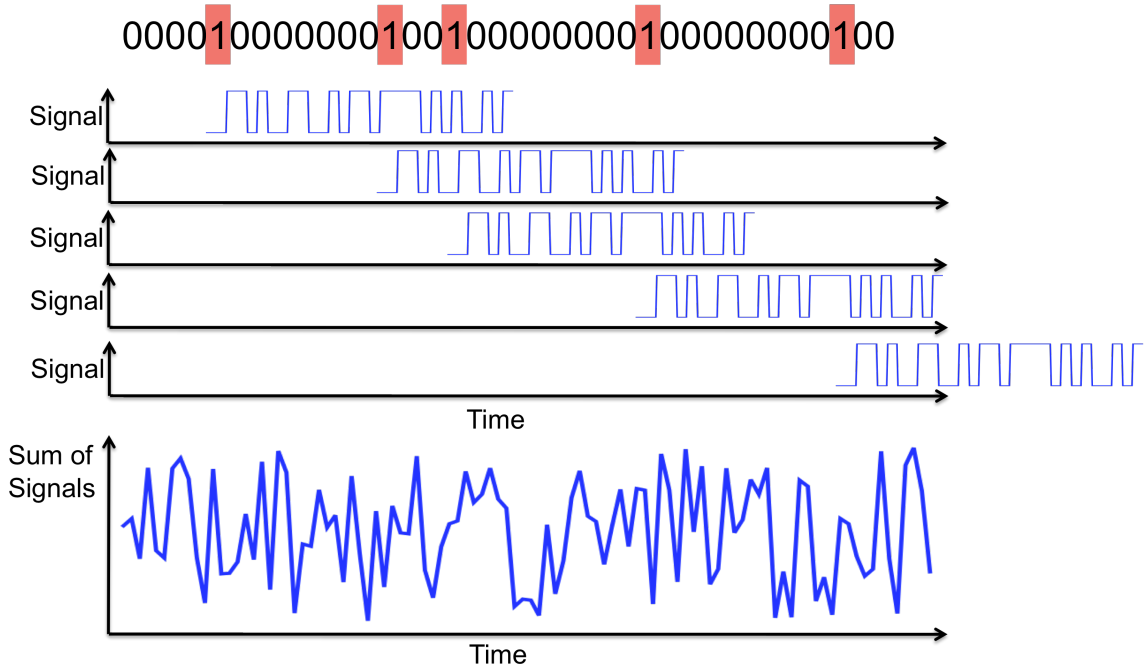


In the first case, there must be a frequency for each bit in the packet. A pure sine wave is broadcast at a given frequency if and only if its corresponding bit in the packet is 1. In the second case, if there are F frequencies and T time slots, then the "packet" contains FT bits. The

hash function would then be modified to ensure that no more than one bit is set to 1 in each time slot.

Both of these approaches solve the high peak-to-mean energy problem. However, they require a large number of frequencies to be distinguishable. This makes this impractical in most situations.

The best solution is one of this form:



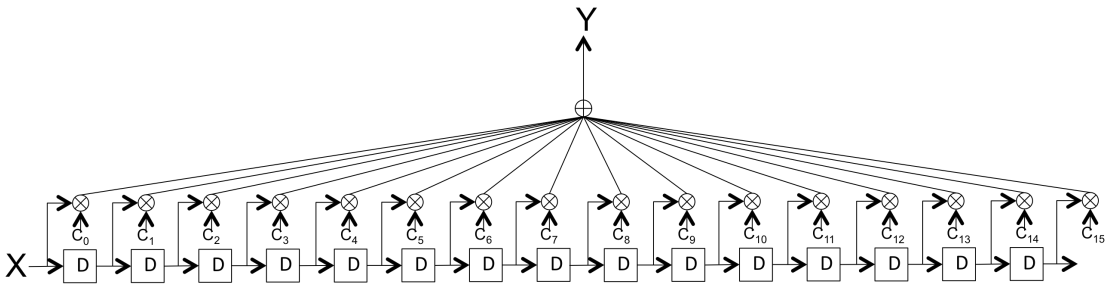
A single chip sequence is transmitted multiple times, with adjacent sequences overlapping. Each bit of the packet corresponds to a point in time, and a new copy of the chip sequence is transmitted starting at each time corresponding to a 1 bit in the packet. The actual, transmitted signal is then the sum of all these chip sequences.

Note that this differs from current spread spectrum systems in that the chip sequences are not modulated to encode data. Rather, all sequences are identical, but they are shifted in time relative to each other, and the set of shifts encodes the data. To ensure a low peak-to-mean power ratio, the chip sequences must be long. For example, the sequence length might be 10 times the average distance between successive 1 bits in the packet. For very high levels of jam resistance, that distance might be hundreds, thousands, or even millions of time steps, so the chip sequence must be thousands or millions of chips long.

This differs from existing spread spectrum chip sequences in being very long, and in not being modulated to encode data. One way to do the de-spreading would be use calculate a correlation between a long period of received signal and the entire chip sequence. Unfortunately, this would require the calculation of an enormous FFT on thousands or millions of elements. And this FFT

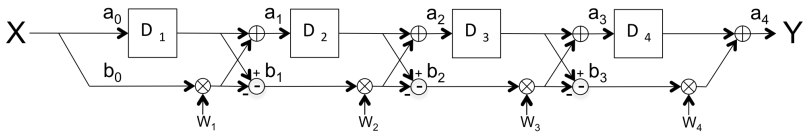
would have to be calculated for every chip, whose duration could be some small number of nanoseconds. This would be difficult.

A faster alternative would be matched Finite Impulse Response (FIR) filters. The filter would be organized as this:



For the sender, the chip sequence is $\{c_0,c_1,\ldots c_{15}\}$, where each c_i is +1 or -1. For the receiver, the chip sequence is reversed. This is a simple solution, but for a sequence of thousands or millions of chips, it requires thousands or millions of parallel multiplies and adds. This is impractical.

However, there is a system that is mathematically equivalent to this, although it is less general in the types of chip sequences that it can handle. It is the Efficient Golay Correlator:



$a_4(t)$ is the impulse response when the input $a_0(t)$ is 1 only at time 0:

$W_n \quad \{-1, +1\}$

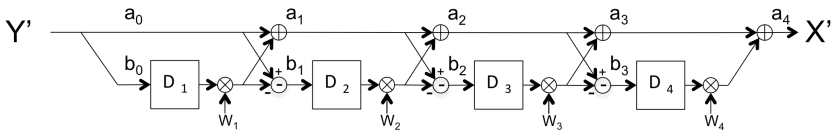
$D_n \quad \{2^0, 2^1, 2^2, 2^3\}$

$a_0(t) = \delta(t)$

$b_0(t) = \delta(t)$

$a_n(t) = a_{n-1}(t-D_n) + w_n\, b_{n-1}(t)$

$b_n(t) = a_{n-1}(t-D_n) - w_n\, b_{n-1}(t)$



$a_n(t) = a_{n-1}(t) + w_n\, b_{n-1}(t-D_n)$

$b_n(t) = a_{n-1}(t) - w_n\, b_{n-1}(t-D_n)$

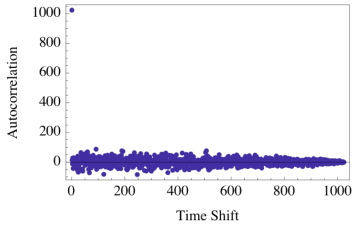
In this system, the w_i parameters are each +1 or -1, and the D_i parameters are some permutation of $\{2^4, 2^4, 2^4, 2^4, 2^4\}$. In the diagram, each square containing a Di is a delay line, whose output is equal to the value of its input D_i time steps ago. The diagram above gives the two matched filters. Either could be used to transmit, and the other would be used to receive. In this case, a chip sequence of a thousand or a million chips can be implemented with just 20 or 40 binary adders, respectively. Since the number of adders is logarithmic in the sequence length, and the amount of RAM in the delay line is linear, this is practical to implement. In fact, the delay line could be implemented as SRAM on the same chip as the adders.

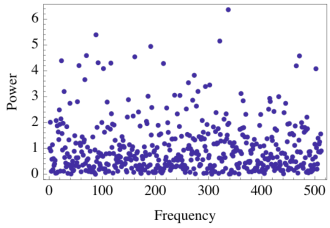
Unlike current spread spectrum systems, the received signal is correlated with a very long chip sequence, rather than just a portion of it. In an ideal system, a single 1 in the sender's codeword will result in a single spike in the correlation value for the receiver's codeword. In practice, slight amounts of clock jitter between the sender and receiver's clocks will result in that energy being smeared across several bit positions. However, this is acceptable, since the correlation is over a very long period, so there is a strong signal before smearing, and because the BBC decoding algorithm can be made to accept any 1 bit as a mark as long as it's within some small distance of the location that the hash function chooses. We also note that unlike current spread spectrum systems, there is no need for the transmitter and receiver to worry about synchronization. The transmitter can transmit at any time, and the receiver will receive the message. And multiple transmitters can transmit messages that overlap in time, and the receiver will receive all of them.

There is an obvious question at this point. The correlator is efficient, but it is constrained in what sequences it can represent. Can it actually produce sequences that have the low autocorrelation and uniform power spectrum that is needed for this application? Several researchers have published results on optimizing the w_i and D_i parameters to achieve these goals. But the results presented here will be for randomly-chosen parameters. In other words, with some effort, the actual results for a fielded system can be even better than those shown here.

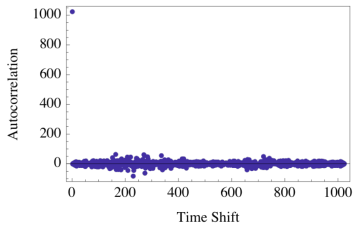
It is known that for long sequences, a random chip sequence will have low autocorrelation and uniform power spectrum. The following diagram compares the proposed system to random chip sequences for a chip sequence of 1000 chips:

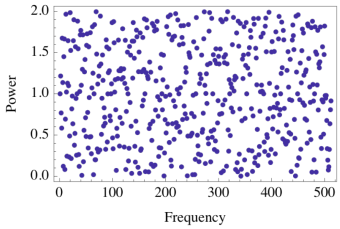
Random:





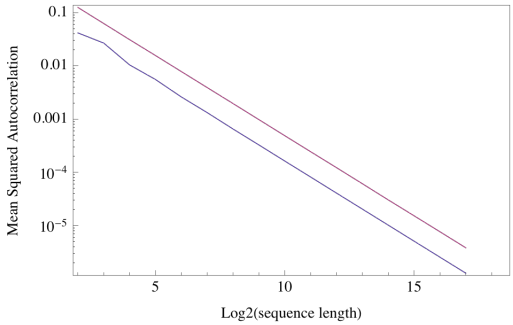
Golay:

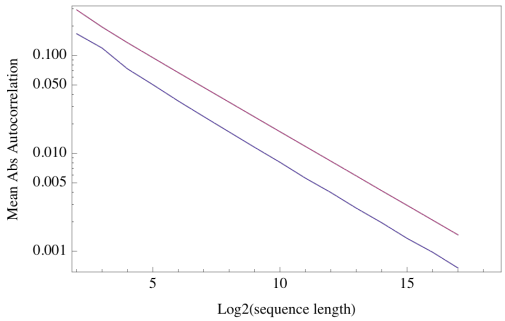




Note that the Golay system has even lower autocorrelation than the random system. It also has a uniform power spectrum. It is known that in the limit for a long random sequence, the power spectrum over any constant period will on average be a constant. But for any particular period, it will deviate randomly from that. Note that the random chip sequence has a power spectrum that is the same for low, medium, and high frequencies, but has many random outliers that reach up to values as high as 6. In contrast, the Golay power spectrum is also the same for low, medium, and high frequencies, but it has no outliers. The power is never more than 2, and is uniformly distributed between 0 and 2. So it appears that the proposed system with random parameters is better than a random chip sequence. And as others have shown, should be further improved by computer searches for good parameters.

The following two graphs compare the random chip sequence (top) to the Golay chips sequence (bottom) on a log-log graph.

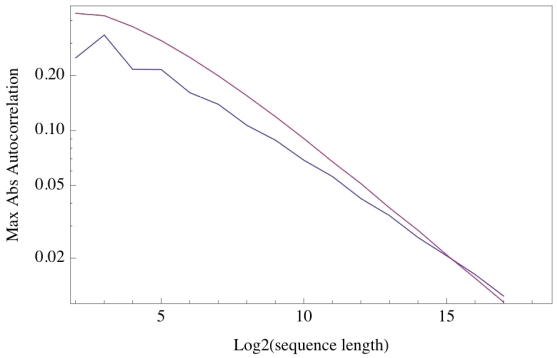




The graph on the left gives the mean squared autocorrelation, where the mean is taken over all nonzero shifts. For these graphs, “autocorrelation” refers to the dot product of the chip sequence (padded with zeros on both sides) with a shifted version of itself, divided by the dot product with no shift. The graph on the right gives the mean absolute autocorrelation. In all cases, the

proposed sequence is better than the random sequence by a constant percentage, for lengths ranging from 8 chips to about 128,000 chips.

The following graph shows the max absolute autocorrelation, where the max is taken over all nonzero shifts. Again, the random sequence is the top line, and the Golay is the bottom. In this case, they are more similar, and become even more similar as the sequence length increases to 128,000. But in both cases, the performance continues to improve as the sequence becomes longer.



.

Conclusions

The proposed implementation of BBC coding for jam resistance appears to be effective and efficient. Further analysis will be done to examine its behavior in the presence of nonlinear channels with fading, and to analyze various attacks that might be used to jam it. But preliminary results suggest that this can be implemented on current hardware with good performance.

References

[1] Baird, Leemon C. III, Bahn, William L. & Collins, Michael D. (2007) Jam-Resistant Communication Without Shared Secrets Through the Use of Concurrent Codes, Technical Report, U. S. Air Force Academy, USAFA-TR-2007-01, Feb 14.