CW Shaping in DSP Software

Getting into shape, via DSP filters.

Alex Shovkoplyas, VE3NEA

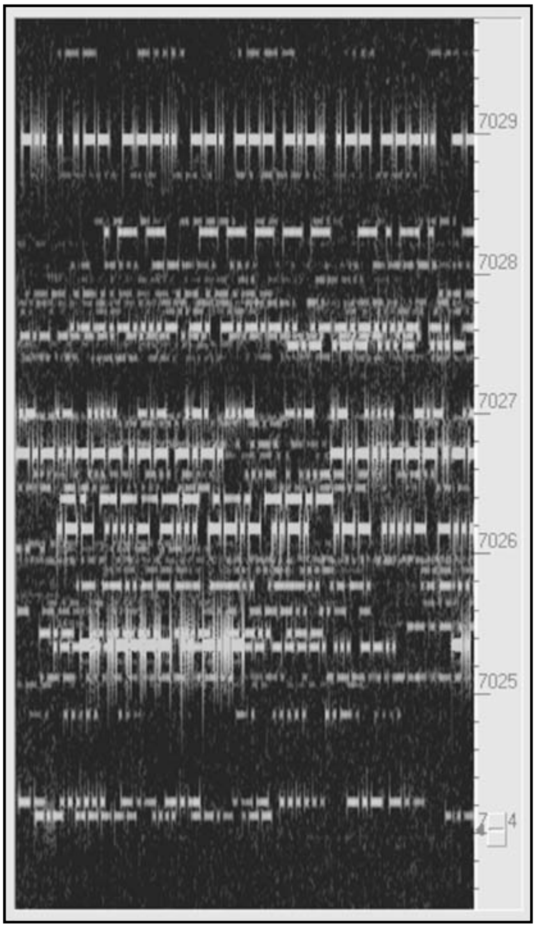

Figure 1 — K7C pileup on 40 m.

CW portions of amateur HF bands are very crowded at times. It is not uncommon to hear several stations within 100 Hz of each other, especially in the big pileups. Figure 1 shows a waterfall display of the K7C pileup on 40 m. The vertical axis is frequency, and the horizontal axis is time. You can see many stations calling the DX; the density of the pileup reaches 10 stations per kilohertz. Under such conditions, it is very important that each signal occupy as little bandwidth as possible.

Traditionally, the bandwidth of a CW signal is reduced by shaping its dots and dashes. In the classical, all-hardware transceivers, CW is shaped with RLC elements, and the choice of shapes that can be achieved is very limited. Now that firmware- and software-defined radios are becoming popular, the developer has much greater control over the CW shape: virtually any shape can be implemented with DSP methods. Among the popular shapes are sine, raised cosine, Gaussian, and the integral of SIN^4.[1-4]

[1]Notes appear on page 7.

85 Raintree Cres
Richmond Hill, ON L4E 3Y8
Canada
ve3nea@dxatlas.com

Usually the developer selects a smooth shape, in a hope that it will result in a narrow bandwidth. There is a better approach, though. CW shaping is in fact a filtering problem. When you replace rectangular edges of Morse elements, this is equivalent to low-pass-filtering the envelope, and there is a one-to-one relationship between the shape applied and the equivalent filter. Thus, instead of selecting a shape, we can select a prototype filter with known characteristics, and use its step response as a waveform to shape CW.

We will demonstrate the method by designing a keying shape that has characteristics similar to those commonly used in amateur transceivers. We will design a shape with the rise time of 5 ms, a value typically used in commercial radios, but with a higher level of spurious emission rejection.

The Blackman-Harris FIR filter will be used in this exercise because it has good characteristics both in the time and frequency domain. This filter has no overshooting, its stopband rejection is over 90 dB, it has a pretty good shape factor and no ripple in the passband. Also, the values of the Blackman-Harris kernel can be calculated with just a few lines of Delphi code:

```
function BlackmanHarrisKernel(x: Single): Single;
const
 a0 = 0.35875; a1 = 0.48829; a2 = 0.14128; a3 = 0.01168;
begin
 Result := a0 - a1*Cos(2*Pi*x) + a2*Cos(4*Pi*x) - a3*Cos(6*Pi*x);
end;
```

Code snippet 1 — The function that calculates a single point of the prototype filter's impulse response.

The step response of the filter is an integral of its kernel. We calculate it as follows:

```
type
 TSingleArray = array of Single;


function BlackmanHarrisStepResponse(Len: integer): TSingleArray;
var
 i: integer;
 Scale: Single;
begin
 SetLength(Result, Len);
 //generate kernel
 for i:=0 to High(Result) do Result[i] := BlackmanHarrisKernel(i/Len);
 //integrate
 for i:=1 to High(Result) do Result[i] := Result[i-1] + Result[i];
 //normalize
 Scale := 1 / Result[High(Result)];
 for i:=0 to High(Result) do Result[i] := Result[i] * Scale;
end;
```

Code snippet 2 — The function that calculates the step response of the prototype filter.

The Len parameter is the kernel length, in samples. Given the desired rise time, the required kernel length can be calculated as 2.7 × RiseTime × SamplingRate.

Now that we have a filter, we can apply it to the CW envelope and see how it works. Normally the filter is applied by convolving its kernel with the input signal, but in the case of CW shaping we do not need to calculate the convolution. A sequence of rectangular pulses with unit amplitude can be viewed as a sum of positive and negative step functions, one function per edge. Since our filter is linear, we can filter each step function separately and then add up the results. Moreover, we already have a filtered version of the step function; it is the step response of the filter that we have just calculated. In practice, assuming that the dot duration is greater than the kernel length, we do not even have to do additions; we just copy the step response to the output buffer at the locations where the pulse edges are:

```
var
 Rise, Fall: TSingleArray;


function ShapeCW(Data: TSingleArray): TSingleArray;
var
 i: integer;
begin
 Result := Copy(Data);
 SetLength(Result, Length(Result) + High(Rise));


 for i:=High(Data) downto 1 do
  if (Data[i-1] = 0) and (Data[i] = 1) then
   Move(Rise[0], Result[i], Length(Rise) * SizeOf(Single))


  else if (Data[i-1] = 1) and (Data[i] = 0) then
   Move(Fall[0], Result[i], Length(Fall) * SizeOf(Single));
end;
```

Code snippet 3 — The function that applies shape to the CW envelope.

The code above assumes that we have precalculated the step response of the filter with the BlackmanHarrisStepResponse function and stored it in the Rise array. The Fall array is a copy of Rise, mirrored left-to-right.

Note that filtering increases the data length by the number of points in the kernel minus 1. Use the overlap-add method to process multiple blocks of data.

The suggested method of CW shaping was tested on a sequence of dots sent at 40 WPM (30 ms dot length). The 5 ms rise time was selected.

The dot waveform after shaping, and the spectrum of the shaped signal are shown on the chart below (Figure 2). The grid steps on the spectrum display are 100 Hz horizontally and 10 dB vertically.

Figures 3 and 4 show the spectrums of the signals keyed at 20 WPM and 80 WPM respectively, shaped with the proposed method. The fine structure of the spectrum depends on the keying speed, while the spectrum envelope is a function of the shaping filter.

The waveforms and spectrums of several other CW shapes are shown on Figures 5-8 for comparison. All shaping filters are tuned to produce a 5-ms rise time, and the keying speed is 40 WPM.

The chart on Figure 9 shows all waveforms and spectra combined. All spectra have virtually the same bandwidth at the levels down to –40 dB, but beyond that point they differ significantly. At –100 dB, all shapes except Blackman-Harris have a very wide bandwidth, well over 1 kHz. The bandwidth of the Blackman-Harris shape at this level is less than 300 Hz. It is interesting to note that visually all tested shapes look almost the same, the differences are so small that they can be barely seen on the chart; but spurious emissions that we want to reject are also of low levels, between –40 to –100 dB, and small differences in the shape play a significant role in achieving a good rejection factor. As you can see from Table 1, these tiny differences can improve rejection by 30-50 dB!

Table 1 shows the level of spurious emissions at a 300 Hz offset for the keying speed of 40 WPM and the rise time of 5 ms for each of the discussed keying shapes.

Blackman-Harris has much better characteristics than all other shapes in the table.

Table 1
Spurious Emission Level at 300 Hz Offset

Shape	*Spurious Emissions at 300 Hz Offset*
Blackman-Harris	< –100 dB
Rectangular	–29 dB
Sine	–51 dB
Raised Cosine	–70 dB
Truncated Gaussian	–83 dB

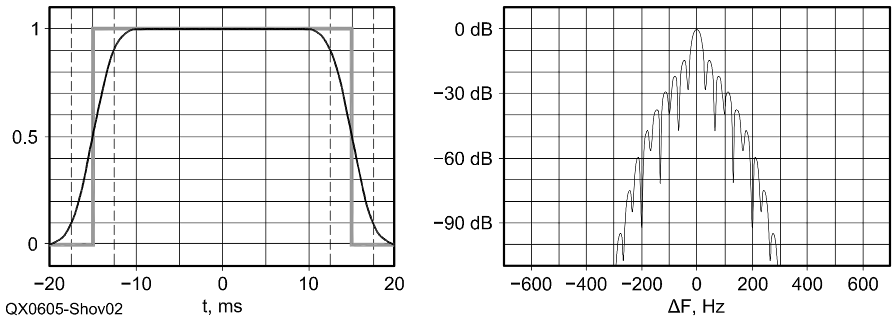

Figure 2 — Blackman-Harris CW shaping.

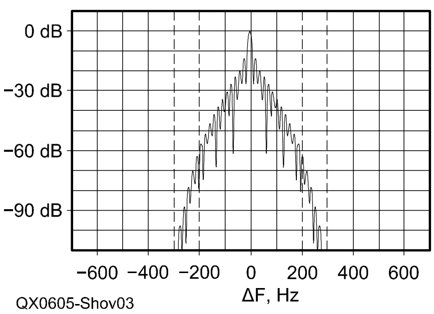

Figure 3 — Blackman-Harris shaping, 20 WPM

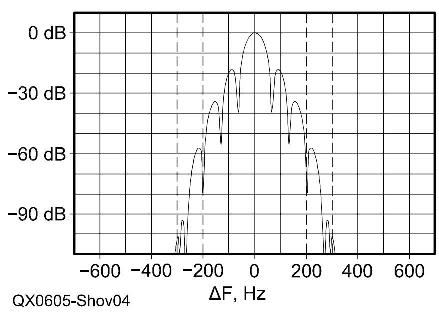

Figure 4 — Blackman-Harris shaping, 80 WPM

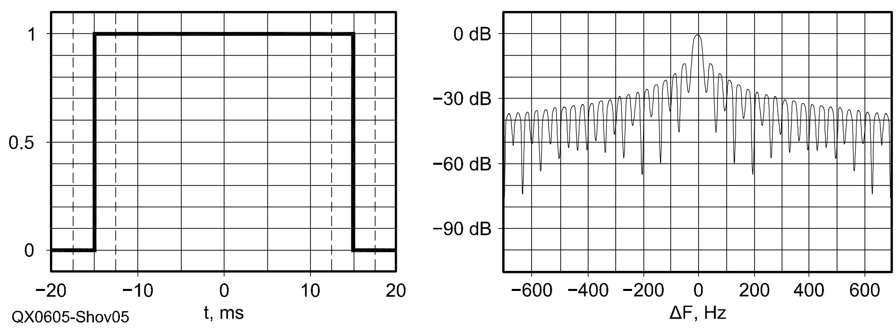

Figure 5 — No shaping.

Design Parameters

Now that we know how to design a keying shape that meets our specifications, we will see what parameters we should set as a design goal.

A good shape must meet several conflicting requirements. The rise time (and the corresponding bandwidth) is only one of those. In fact, it is the easiest parameter to achieve. Basically, we want the following from the signal shape:

1) the signal must be easy to copy, the distortion of the original shape should be minimal;

2) the power density of spurious emissions (key clicks) away from the operating frequency must be as low possible; and

3) the length of the keying shape (which is often many times longer than the rise time)

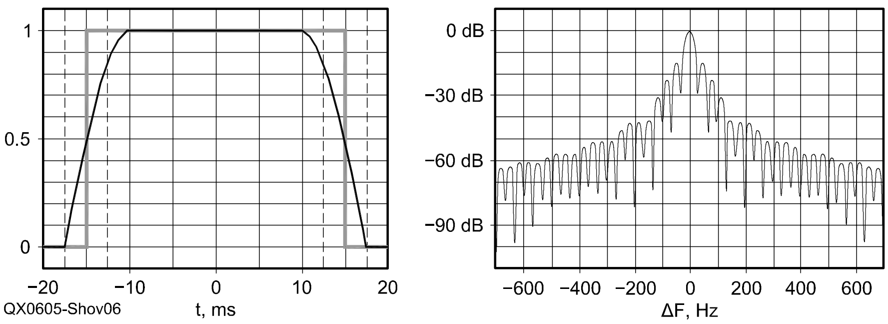

Figure 6 — Sine shaping.

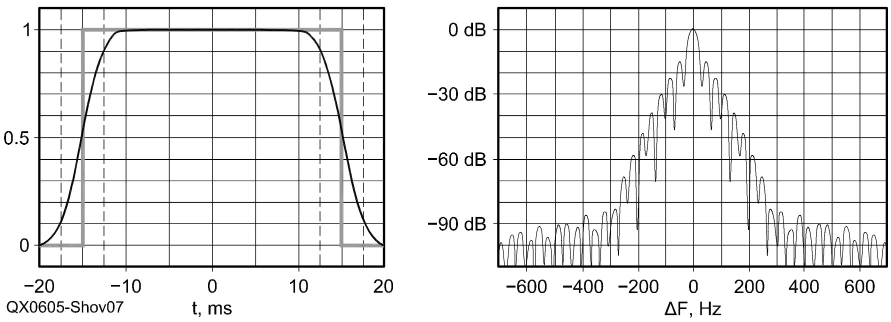

Figure 7 — Raised Cosine shaping.

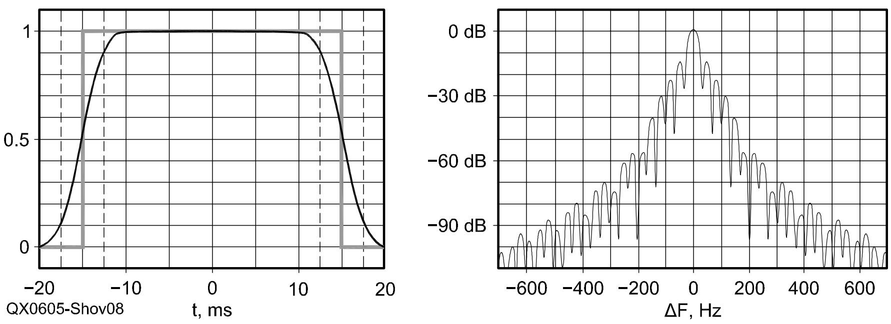

Figure 8 — Gaussian shaping, kernel truncated at 3 × sigma.

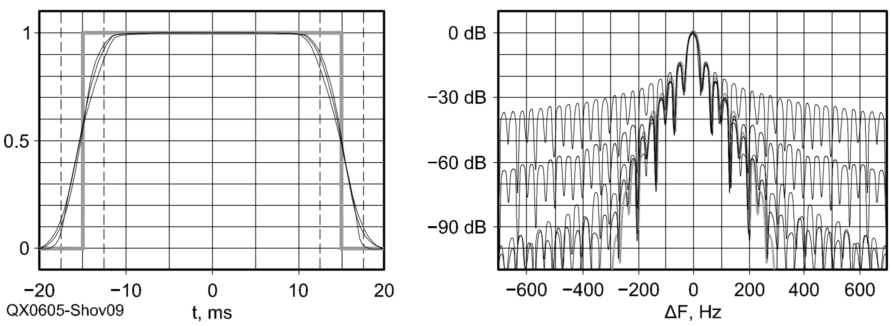

Figure 9 — All charts combined. The Blackman-Harris shaping chart is the bold line.

must be small since it affects the signal delay introduced by the shaper.

It follows from the above that the choice of the CW shape is not just a single parameter minimization problem. The developer must decide what level of spurious emissions he wants to achieve, at what frequency offset that level should be measured, and to what degree (1) and (3) can be relaxed to meet (2).

One possible strategy is to design the filter and resulting keying shape to achieve (1) and to use the remaining degrees of freedom to optimize if for (2) and (3). The design will then be reduced to choosing the three main parameters of the prototype filter: bandwidth, shape factor, and stopband rejection.

Obviously, we want the stopband rejection to be as high as possible, but what about the other two parameters? How do they affect the operator's ability to copy weak signals? To answer this question, I wrote a program called *CwBwTest*, and performed a few tests.

The program can be downloaded from **www.dxatlas.com/CwShaping/CwBwTest. zip**. It has three tabbed pages. On the first page, there are controls that play CW audio with specified keying speed, signal-to-noise ratio, bandwidth and shape factor.

Page 2 presents a weak signal reception test. The program plays random call signs in the noise at 20 WPM, prompts you to copy those call signs, and plots the copy accuracy on a chart as a function of the SNR for different shaping parameters. Filtering is applied to the CW signal only, the noise bandwidth is 500 Hz in all cases. This is equivalent to shaping the signal in the transmitter and receiving it with a typical CW receiver.

Page 3 in the program shows the envelope oscillogram of the signal being played. The results of the tests are shown on Figure 10. The tests were performed by three operators independently; 1500 call signs were copied in total. The results were within 0.5 dB of each other, and the curves had the same shape. The chart presents the combined results of all tests.

As can be seen from the chart, the optimal bandwidth of the 20 WPM signal is 30 Hz. The copy accuracy does not increase as the bandwidth increases beyond this point, but the accuracy drops at lower bandwidths. At the optimal bandwidth, the exact shape of the envelope does not affect the accuracy.

Based on the test results, we can conclude that the keying shape at 20 WPM should be selected to produce the 30 Hz bandwidth at –6 dB, or 1.5 × WPM. The shape factor of 2.7 is optimal — not because of the copy

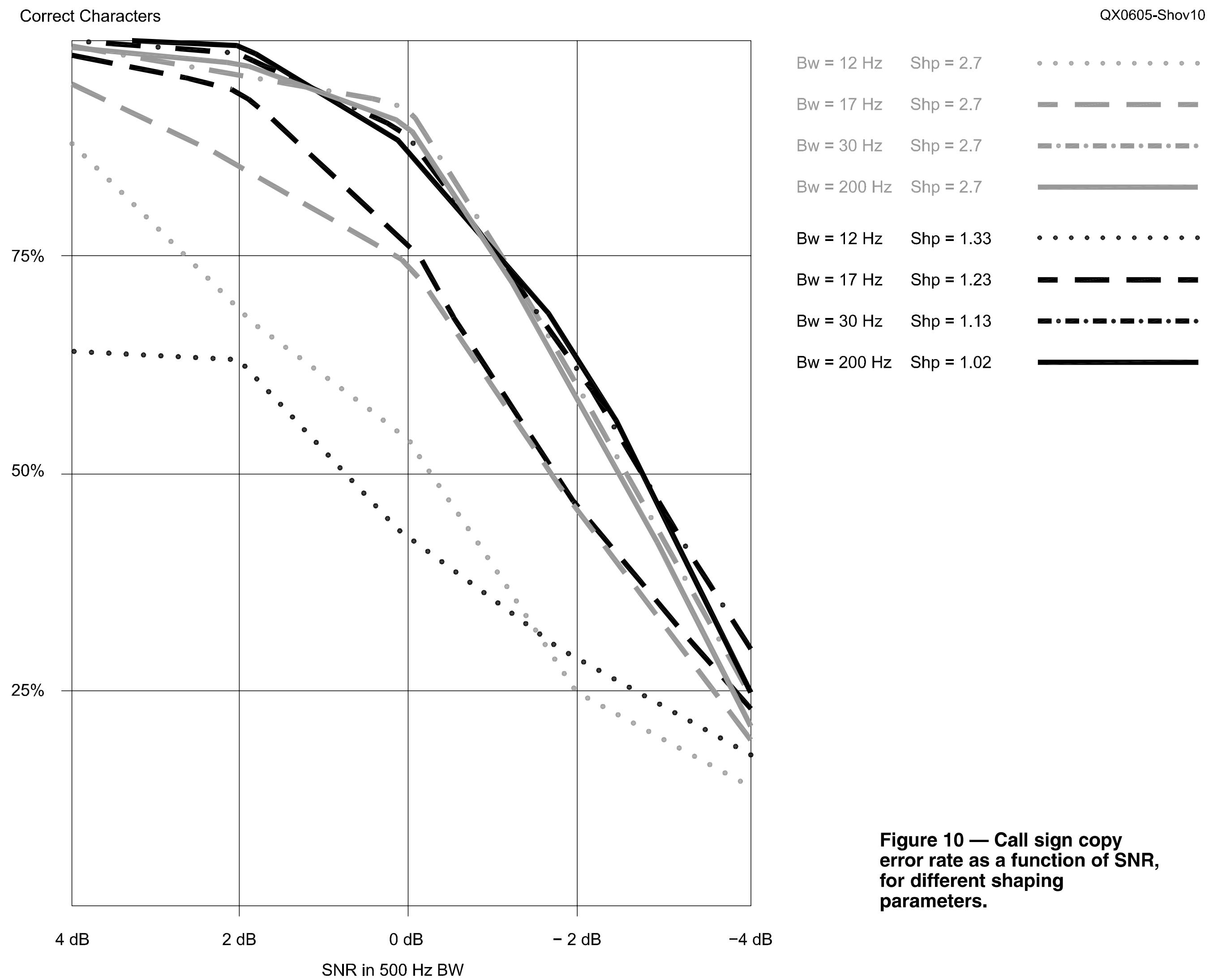

Figure 10 — Call sign copy error rate as a function of SNR, for different shaping parameters.

accuracy, but because it ensures no Gibbs effect that makes CW sound unpleasantly. With these parameters and the Blackman-Harris keying shape, the total bandwidth at the –100 dB level is about 80 Hz.

Notes

[1] *On the Occupied Bandwidth of CW Emissions*. Douglas T. Smith Editorial Services, 2004. Online publication: **www.dougsmith.net/cwbandwidth1.htm**.

[2] PowerSDR 1.4.4 Source Code: Flex-Radio Downloads. Online download: **www.flex-radio.com/download_files/**.

[3] *Spectral Analysis of a CW keying pulse.* Kevin Schmidt, W9CF. Online publication: **fermi.la.asu.edu/w9cf/articles/click/index.html**.

[4] *The T03DSP High Performance Transceiver with DSP IF processing.* Oleg Skydan, UR3IQO. Online publication: **skydan.in.ua/T03DSP/CWExciter.htm**.

Alex Shovkoplyas, VE3NEA (ex-UB5EMI and UR5EMI), has been licensed since 1980. He attended Dnipropetrovsk State University in the Ukraine, and majored in Radio-Physics and Electronics. Founder and owner of Afreet Software Inc, he is employed as a software developer with Resume Mirror Inc. He is the author of Morse Runner, HamCAP, DX Atlas *and other software. His Web sites are* **www.dxatlas.com** *and* **www.netvampire.com/Ham**. □□

Upgrade to Phase-Locked Performance

Model 1152
PLL for DEMI Transverters
Model 5112
PLL for DB6NT Transverters
Model 902
PLL for 902MHz
Model SEQ-1
Micro-Controlled Sequencer

jwm
ENGINEERING GROUP

949-713-6367 / http://www.jwmeng.com