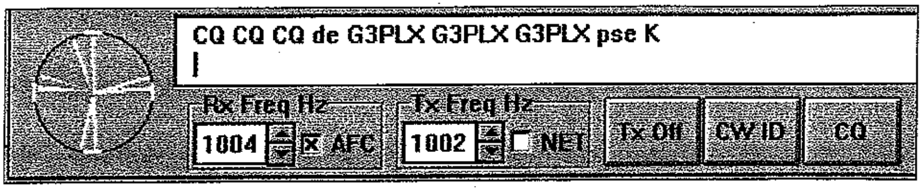

Screenshot of the control panel of the PSK31 Windows program, with the tuning display showing a slightly noisy QPSK signal, and showing the fine-tuning controls of the receive and transmit audio tones.

it will be a one. They all do their encoding to guess what the next phase-shift will be, and are given marks out of ten again which are added on to their earlier scores. The worst 16 encoders are killed-off again and the cycle repeats.

It's a bit like Darwin's theory of evolution, and eventually all the descendants of the encoders that made the right guesses earlier will be among the survivors and will all carry the same 'ancestral genes'. We therefore just keep a record of the family tree (the bit-guess sequence) of each survivor, and can trace back to find the transmitted bit-stream, although we have to wait at least 5 generations (bit periods) before all survivors have the same great great grandmother (who guessed right five bits ago). The whole point is that because the scoring system is based on the running total, the decoder always gives the most accurate guess, *even if the received pattern is corrupted,* although we might need to wait a bit longer than 5 bits for the best answer to become clear. In other words, the Viterbi decoder corrects errors.

The longer we wait, the more accurate it is. I chose a decoder delay of 4 times the time spread, or 20 bits. We now have a 25 bit delay from one end to the other, (800mS), giving a round-trip delay to a two-way contact of 1.6 seconds. I think this is about the limit before it becomes a nuisance. In any case, the decoder could change to trade performance for delay without incompatibility.

QPSK ON THE AIR

PSK31 OPERATORS find QPSK can be very good but is sometimes disappointing. In bench tests with white noise, it is actually *worse* than BPSK, confirming the simulation work mentioned earlier, but in conditions of burst noise, improvements of up to 5 times in the character error-rate have been recorded. This performance doesn't come free, however. Apart from the transmission delay, which can be a bit off-putting, QPSK is twice as critical in tuning as BPSK. A QPSK signal will start to decode wrong when the phase shift is 45° out, and that will be the case when the tuning error is only 3.9Hz. This could be a problem with some older radios. What tends to happen is that contacts start on BPSK and change to QPSK if it is worth doing and if there is no drift. There is one aspect of QPSK that has to be kept in mind - it is important for both stations to be using the correct sideband - on BPSK it doesn't matter.

EXTENDING THE ALPHABET

IN THE UK, our computer keyboards have a Pound sign above the figure 3, and many people will have noticed that they can't reliably send Pound signs over Internet, for example. This is because the Internet uses the 128-character ASCII alphabet, and the Pound sign is not part of that set but part of the ANSI character set which has an additional 128 characters and symbols. PSK31 as described so far is the same as Internet. It's a small problem in the UK, but much more of a nuisance in other parts of the world where characters like the German umlauts, French accents, and Spanish tildes are also missing from the ASCII character set. Luckily, with PSK31, the Varicode alphabet is very easy to extend without creating incompatibility with earlier versions. Since the Windows® operating system uses ANSI, and most PC programs are now written for Windows, I have recently extended the PSK31 alphabet in a Windows version.

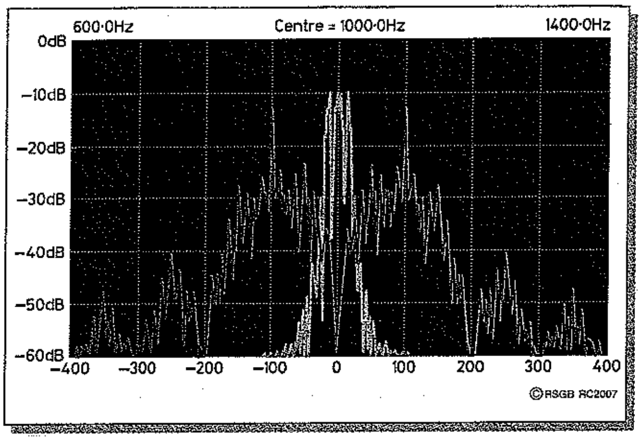

Fig 5: Comparison of PSK31 spectrum with 100 baud, 200Hz shift FSK.

In the basic PSK31, if there was no '00' pattern received 10-bits after the last '00', the decoder would simply ignore it as a corruption. If now I let the transmitter legally send codes longer than 10 bits, the standard decoder will just ignore them and the extended decoder can interpret them as extra characters. To get another 128 varicodes means adding more ten-bit codes, all the eleven-bit ones, and some twelve-bit codes. There seemed little reason to be clever with shorter common characters, so I chose to allocate them in numerical order, with code number 128 being 1110111101 and code number 255 being 101101011011. The vast majority of these will never be used. It would not be a good idea to transmit binary files this way!

SUMMARISING

THIS ARTICLE HAS tried to identify some of the characteristics of modern HF data-transmission modes that have contributed to the decline in 'live QSO' operation, unlike traditional RTTY which is still widely used. By concentrating on the special nature of live-QSO operation, a new RTTY mode has been devised which uses modern DSP techniques and takes advantage of the frequency stability of today's HF radios. The bandwidth is much narrower than any other telegraphy mode. **Fig 4** shows the spectrum occupied by PSK31 and **Fig 5** compares this to the bandwidth of standard FSK.

At the time of writing (Nov 1998) PSK31 is available for the Texas TMS320C50DSK written by G0TJZ, the Analog Devices ADSP21061 'SHARC' kit written by DL6IAK, and the Motorola DSP56002EVM written by myself. One or two people are making promising progress with sound cards, and there is scope for implementation of PSK31 with traditional hardware modulators and demodulators. The two tables contain sufficient information to define PSK31 for those that want to try it themselves. All the ready-to-run software and news of the latest developments and activity can be found on the PSK31 Internet web page at http://bipt106.bi.ehu.es/psk31.html

Referring to **Table 2**, the left column contains the 32 combinations of a run of five Varicode bits, transmitted left bit first. The right column is the corresponding phase shift to be applied to the carrier, with 0 meaning no shift, 1 meaning advance by 90, 2 meaning polarity reversal and 3 meaning retard by 90. A signal that is advancing in phase continuously is higher in radio frequency than the carrier.

As an example, the 'space' symbol, a single 1 preceded and followed by zeros, would be represented by successive run-of-five groups 00000, 00001, 00010, 00100, 01000, 10000, 00000, which results in the transmitter sending the QPSK pattern .. 2,1,3,3,0,1,2, ..

Note that a continuous sequence of zeros (the Varicode idle sequence) gives continuous reversals, the same as BPSK. ♦

00000 2	01000 0	10000 1	11000 3
00001 1	01001 3	10001 2	11001 0
00010 3	01010 1	10010 0	11010 2
00011 0	01011 2	10011 3	11011 1
00100 3	01100 1	10100 0	11100 2
00101 0	01101 2	10101 3	11101 1
00110 2	01110 0	10110 1	11110 3
00111 1	01111 3	10111 2	11111 0

Table 2: The convolutional code.