

**Wanted:**  
**Algorithm for a non-random Pseudo Random sequence**

Background:

The WSPR.EXE software offers transmitter duty cycles of 20/25/33% with a randomised transmit pattern. Observing the 33% stream, it is rare to see two consecutive transmission periods and three consecutive ones have never been observed. Similarly, I have hardly ever seen gaps of more than four periods of receive. If the randomising sequence were truly random, then longer periods of each should be seen, and in fact are compulsory in a truly random stream

Conclusion: some clever algorithm is used to generate what is, in effect, a bandpass-filtered PRS.

I have built a WSPR beacon and so far have incorporated a PN sequence that generates transmission periods based on a 255 bit Linear feedback Shift Register. This gives a flat random number between 1 to 255 which is multiplied by D, (the duty cycle taking values of 2, 3, 4 or 5) and divided by 256 to give a random integer between 0 and D-1. When the returned value is zero, transmission is permitted.

Modeling the process does indeed give a set of transmission periods with the required duty cycle – in fact quite accurately so when measured over the entire 255 repeat period of the LFSR sequence.

BUT its not ideal. Being a truly (pseudo) random generation there are runs of several consecutive periods of transmission, and longer off periods than wanted.

All this is entirely what you would expect from a PN sequence generated this way, runs of consecutive ones up to the length of the sequence MUST occur occasionally for it to be truly (pseudo) random.

But this is not the best sequence for a WSPR beacon transmitter...

As I write this, the beacon set to 50% duty cycle has just generated the pattern 000001110010101010100001001 where 0 is Rx, 1 is Tx. And last night it sent 5 consecutive Tx periods at least once while being watched.

So the 50% value is only guaranteed over the complete 255 long LFSR repeat length.

I am looking for an algorithm that can be implemented in a PIC using 8 bit integer arithmetic that implements more closely the sequences of the WSPR.EXE software. Ie no more then two consecutive Tx periods (except possibly at 50% duty cycle where 3 may be allowed) , and no more than , say,  $3 \cdot D$  receive periods.

Of course, one straightforward cop-out would be to pre-compute tables with desired [non-random] properties – they wouldn't have to be very big and are easy to do in a PIC

**But that is cheating !**