

Re: 8 Bit Random Numbers

Source: <http://sci.tech-archive.net/Archive/sci.electronics.basics/2009-02/msg00061.html>

- *From:* Bill Bowden <wrongaddress@xxxxxxx>
 - *Date:* Tue, 3 Feb 2009 18:36:48 -0800 (PST)
-

On Feb 3, 9:59 am, nos...@xxxxxxxxxxx wrote:

Bill Bowden wrote:

Nobody wrote:

John Fields wrote:

Well, we could argue forever about what constitutes "[E]XOR-based", but it's not an LFSR, and I thought it was clear that's what "nosspam" was referring to by "XOR-based PRNG".

He was, and you're both wrong.

What you're saying, in effect, is that if an 8 cylinder ICE fitted with a 7 cylinder ignition system was fitted with a system winch allowed all 8 cylinders to work it would no longer be an ICE, which is total nonsense.

Re: 8 Bit Random Numbers

No, I'm saying that if you modify an LFSR so that its input is no longer an N-input XOR of (some of) its outputs, it's no longer an LFSR.

I don't think that anyone would contest that there exist *other* circuits which will cycle through all 256 8-bit values, or at least don't have the lock-up state.

There are circuits which don't have the lockup state, but an LFSR isn't one of them.

The output of the 8 bit LFSR that Fields did is exactly the same as the standard version using 4 taps. The only addition is the zero state appears between the values hex 80 and 01. The new sequence goes 80,00,01.....

What part of

"A linear feedback shift register (LFSR) is a shift register whose input bit is a linear function of its previous state. The only linear functions of single bits are xor and inverse-xor..."

are you having trouble understanding?

I'm having trouble with the linear part. Linear suggests to me an equal amount of "1s" and "0s" over a long period, which should be 1024 of each for an 8 bit counter. The LFSR without Fields mod produces 1024 "1s" and only 1023 zeros which is slightly non-linear.

Using Fields approach produces equal amounts of "1" and "0" or 1024 of each, which seems closer to linear.

What am I missing?

-Bill

Re: 8 Bit Random Numbers

Re: 8 Bit Random Numbers

What Fields posted was *NOT* an 8 bit LFSR. It was *NOT* XOR based. (BTW, why does Fields keep calling XOR "EXOR?" Does he have some sort of problem using standard terminology?) Simply saying "you are both wrong" does not change reality. If the two of you would simply stop calling whatever his circuit is a "LFSR" or a "XOR Based PRNG", then everyone here could agree. There is nothing wrong with his design. You are simply calling it by the wrong name. Just because someone is an engineer, that doesn't mean that he has to be stubborn and refuse to admit any error. The world will not end if you admit to using the wrong terminology.

BTW, for most applications a standard Galois LFSR is superior to what Fields posted. When implemented using logic gates, The XOR gates are run in parallel rather than in serial, reducing propagation delay and allowing for faster cycling. When implemented in software, it is more efficient because the XOR can be computed a word at a time. Code it or breadboard it and see.

Yes, I know about that, but haven't figured out the 8 bit number to xor with the random register to get the desired result.

Maybe you know what it is?

-Bill

.