PRQS Test Patterns for PAM4

IEEE 802.3bs 400GbE Task Force
Coconut Point, Florida
Sept. 14-16, 2015

Ilya Lyubomirsky, Finisar

Outline

- Objectives
- PRQS Generation Techniques
- Analysis and Discussion
- Conclusion

Objectives

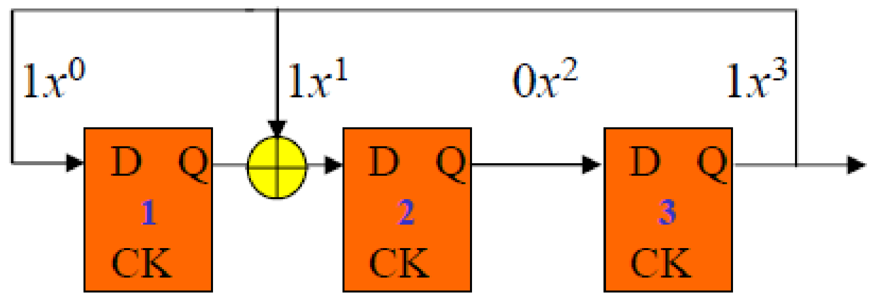
- Propose the pseudo random quaternary sequence (PRQS) as a test pattern for PAM4
- Describe efficient methods for generating PRQS
- Describe the statistical properties of PRQS

Why PRQS for PAM4?

- PRBS patterns mapped to PAM4 can result in poor baseline wander and clock content characteristics (see analysis of PRBS15 in anslow_3bs_03_0714)
- SSPR pattern has good baseline wander characteristics but may not posses ideal “random” characteristics
- PRQS patterns are a natural generalization of PRBS to quaternary sequences, having similar random properties
- PRQS patterns of modest length can provide good baseline wander and clock content characteristics for a test pattern (anslow_01_0915_smf)
- PRQS patterns are generated algorithmically using linear feedback shift registers, providing efficient implementation and a flexible design

The Algebra of PRBS Generation

$$P(x) = x^3 + x + 1$$



States cycle through all nonzero elements of GF(2^3) constructed from binary primitive polynomial P(x) = x^3+x+1

1	1	1	1
1	0	1	2
1	0	0	3
0	1	0	4
0	0	1	5
1	1	0	6
0	1	1	7
1	1	1	

PRBS generator is based on a primitive polynomial with coefficients in GF(2)
=> Similarly PRQS requires a primitive polynomial but with coefficients in GF(4)

GF(4) Arithmetic

Let $\mathrm{GF}(4) = \{0, 1, \beta, \delta\}$, where $0, 1$ are additive and multiplicative identities.

The field axioms allow only these operation tables:

$+$	0	1	β	δ
0	0	1	β	δ
1	1	0	δ	β
β	β	δ	0	1
δ	δ	β	1	0

$\times$	0	1	β	δ
0	0	0	0	0
1	0	1	β	δ
β	0	β	δ	1
δ	0	δ	1	β

Multiplication table for $\mathrm{GF}(4)$ in binary (lsb first) and 4-ary:

$\times$	00	10	01	11
00	00	00	00	00
10	00	10	01	11
01	00	01	11	10
11	00	11	10	01

$\equiv$

$\times$	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

Primitive Polynomials over GF(4)

TABLE 5

SOME PRIMITIVE POLYNOMIALS

GF(4)		GF(8)		GF(9)		GF(16)	
n	f(x)	n	f(x)	n	f(x)	n	f(x)
1	1A	1	1A	1	1A	1	1A
2	11A	2	1AA	2	11A	2	11G
3	111A	3	101A	3	101A	3	101G
4	101AB	4	1001C	4	1001E	4	101AB
5	10001A	5	10011C	5	10010A	5	1000AB
6	100011A	6	100001A	6	10001AA		
7	100001AB	7	100001AC	7	1000001A		
8	10000101A						
9	100000011A						
10	10000001AAA						
11	10000000001A						

$X^2 + X + 2$ → 2

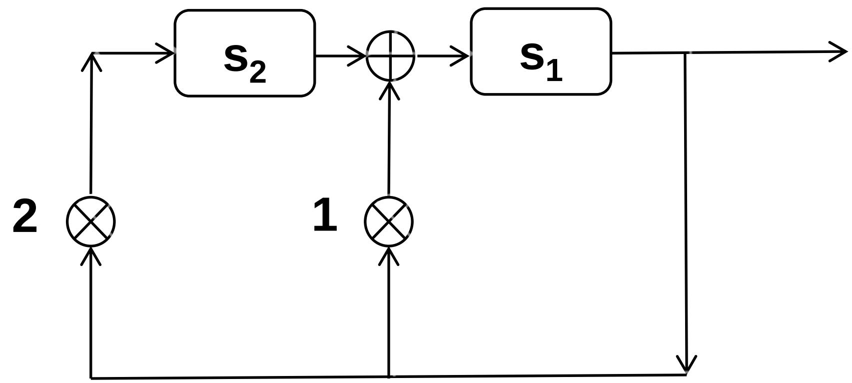
$X^7 + X^2 + 2X + 3$ → 7

$X^{10} + X^3 + 2X^2 + 2X + 2$ → 10

Note in the notation above GF(4)= {0, 1, A, B}

Source: D. H. Green, et. al. "Irreducible Polynomials over Composite Galois Fields and their Applications in Coding Techniques," ***Proc. IEE*****, vol. 121, no. 9, pp. 1935-1939,1974**

Example: PRQS2 Generator (length 4^2-1)



Output Sequence b_n

$P(x) = x^2+x+2$

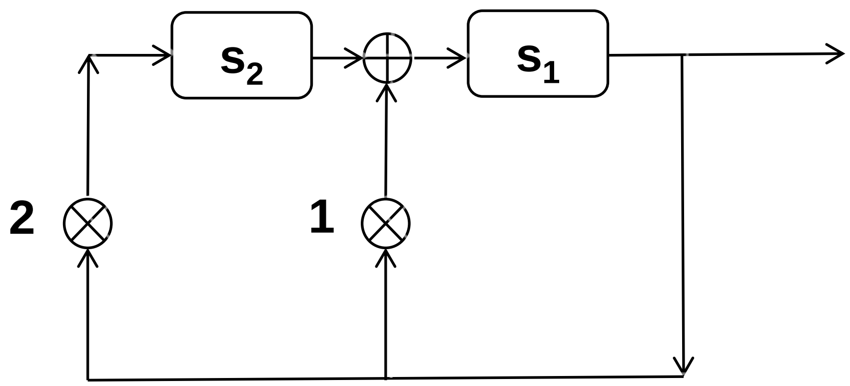
s_2	s_1	b_n
1	0	0
0	1	1
2	1	1
2	3	3
1	1	1
2	0	0
0	2	2
3	2	2
3	1	1
2	2	2
3	0	0
0	3	3
1	3	3
1	2	2
3	3	3
1	0	

GF(4) ARITHMETIC OPERATIONS OF ADDITION AND MULTIPLICATION

+	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

x	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

Connection Between PRQS and PRBS



$b(D) = (2D^2+D)b(D)+s_1+s_2D$

Note: D is the unit delay operator,
b(D) output sequence polynomial,
initial seed s_1, s_2

$\Rightarrow \; b(D) = (s_1+s_2D) / (2D^2+D+1)$

Factor PRBS4 generator polynomial $D^4+D^3+1 = (2D^2+D+1)(3D^2+D+1)$

$\Rightarrow b(D) = (s_1+s_2D)(3D^2+D+1) / (D^4+D^3+1)$

$\Rightarrow b(D) = [3s_2D^3+(3s_1+s_2)D^2+(s_2+s_1)D+s_1] / (D^4+D^3+1)$

Let initial seed $s_2 = 1, s_1 = 0$**, then** $b(D) = (3D^3+D^2+D) / (D^4+D^3+1)$

Go to binary notation for GF(4): 0 = [0 0], 1 = [0 1], 2 = [1 0], 3 = [1 1]

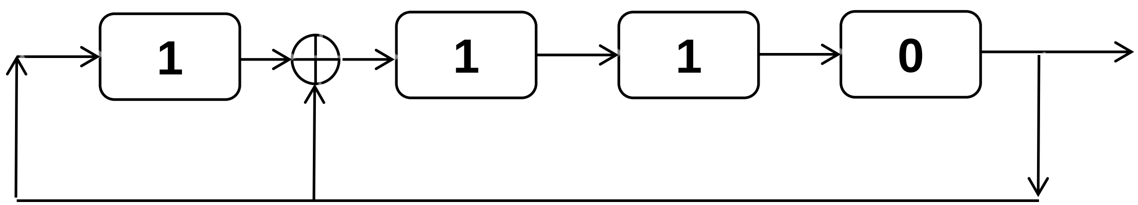
$\Rightarrow b(D) = ([1\ 1]D^3+[0\ 1]D^2+[0\ 1]D) / (D^4+D^3+1)$

Connection Between PRQS and PRBS

b(D) = ([1 1]D^3+[0 1]D^2+[0 1]D) / (D^4+D^3+1)

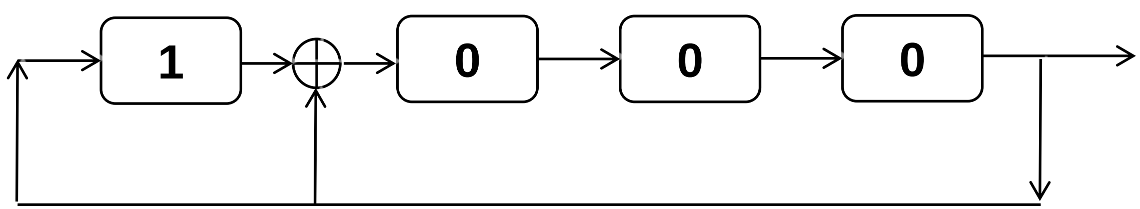
Separate b(D) into LSB and MSB

LSB: (D^3+D^2+D) / (D^4+D^3+1)



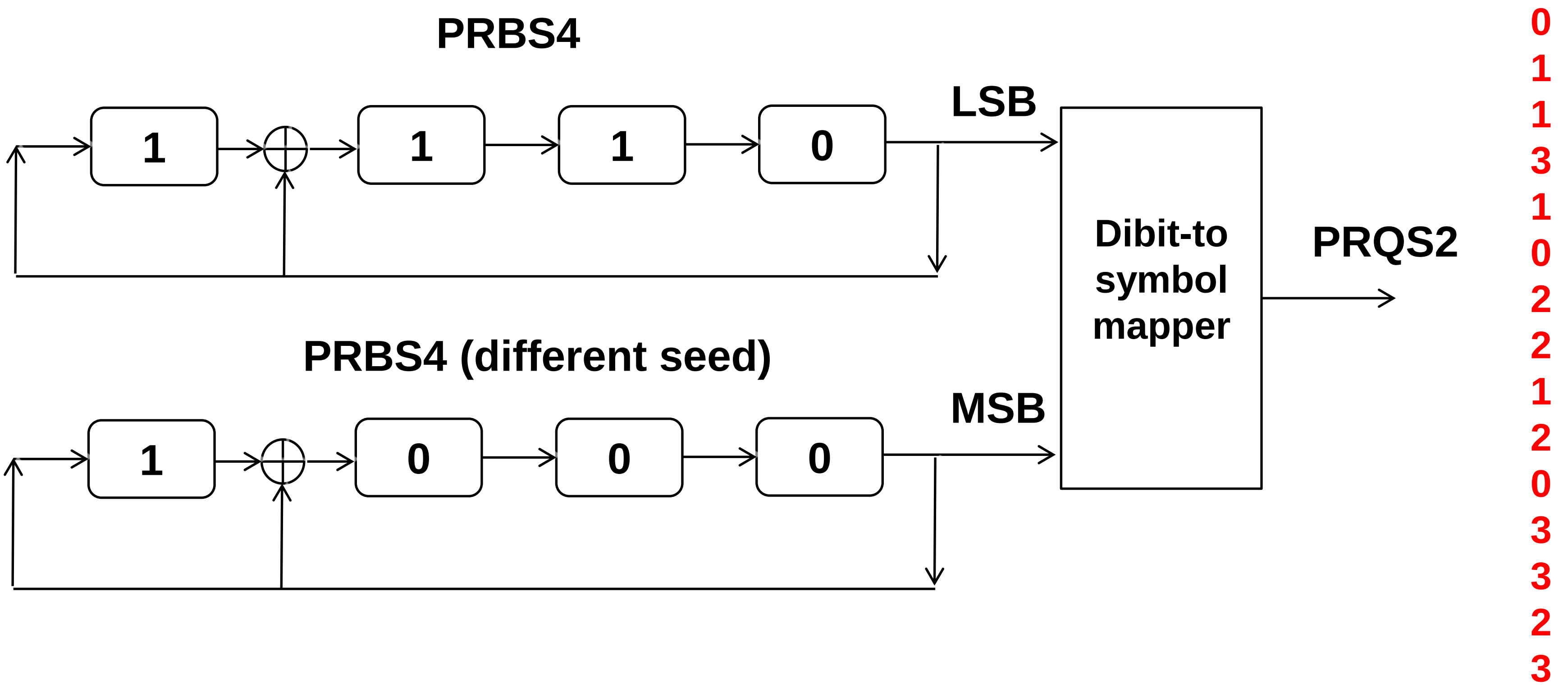
PRBS4

MSB: D^3 / (D^4+D^3+1)

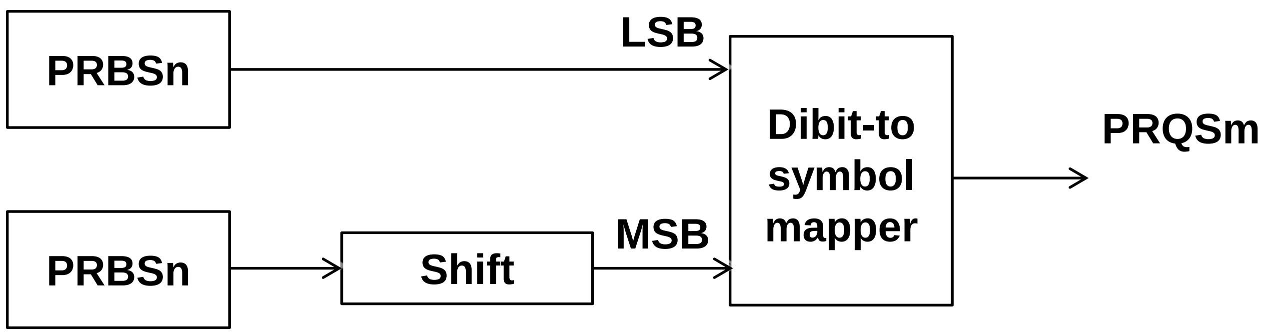


PRBS4
(shifted 10 bits)

PRQS2 Generator by PRBS4 Multiplexing



PRQSm Generator by PRBSn Multiplexing

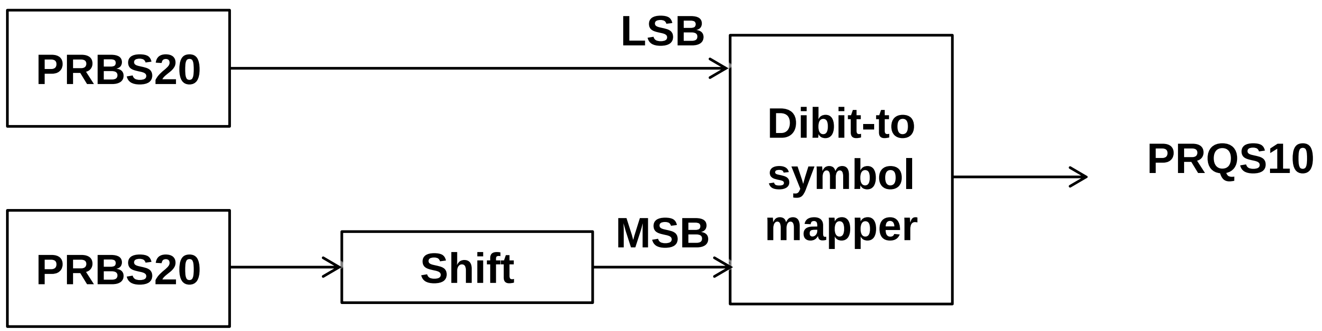


- To generate PRQSm of length 4^m-1, need PRBSn with n=2m
- Shift MSB PRBS by m or (4^m-1)/3 or (4^m-1)2/3
- Dibit-to-symbol mapper can use Natural or Gray Coding

Sources: 1. D. van den Borne, et. al., "Pseudo Random Sequences for Modeling of Quaternary Modulation Formats," proceedings *OECC/IOOC*, Pacifico Yokohama, July, 2007

2. J. J. Komo and M. S. Lam, "Maximal Length Sequences for QPSK," *Proceedings of the Twentieth Southeastern Symposium on System Theory* , Charlotte, March,1988

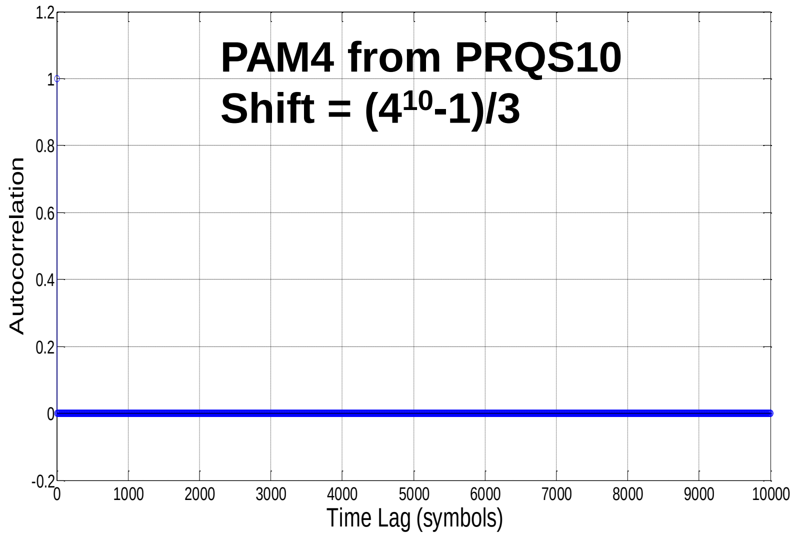
Proposed PAM4 Test Pattern: PRQS10

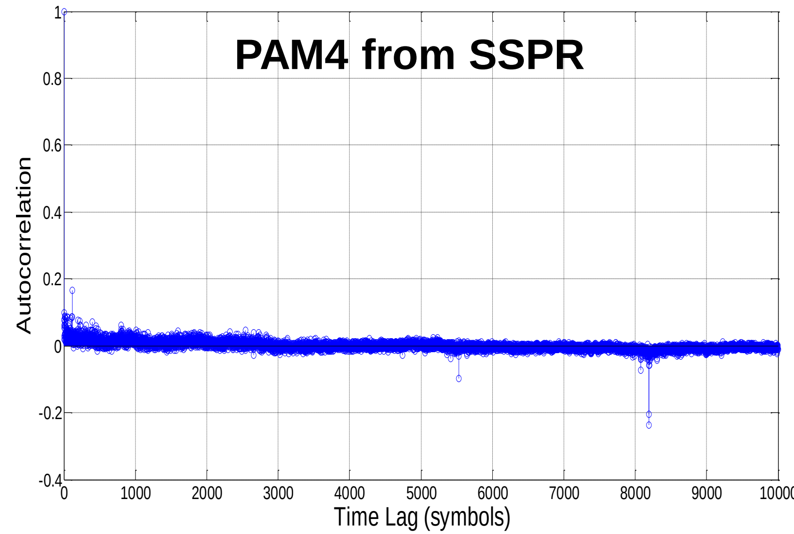


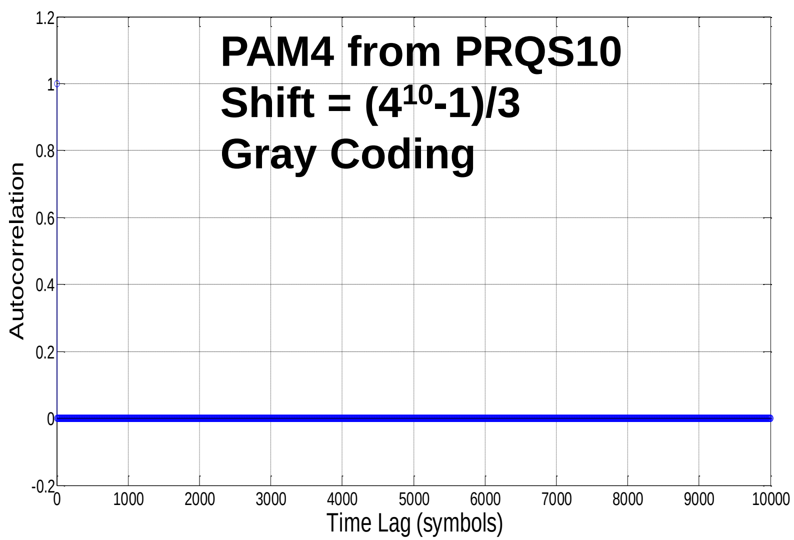
- Algorithmically generated based on multiplexing two PRBS20 patterns
- True maximum length quaternary sequence, i.e. contains all 10 length symbol patterns with equal probability (except for all 10 zeros)
- Pattern length = 4^{10}-1 = 1,048,575 ~ 1 M (short enough for DCAs to support)
- Good “random” statistical properties (see also next slide):

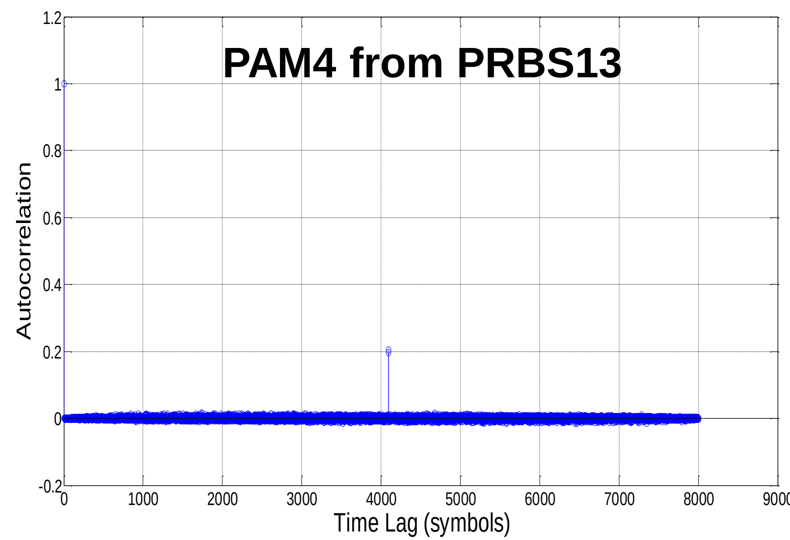
	P0	P1	P2	P3	Transition Density
PRQS10	**0.2500**	**0.2500**	**0.2500**	**0.2500**	**0.7500**
SSPR	0.2573	0.2279	0.2575	0.2573	0.7101
PRBS13	0.2499	0.2500	0.2500	0.2500	0.7501

Simulated Autocorrelation

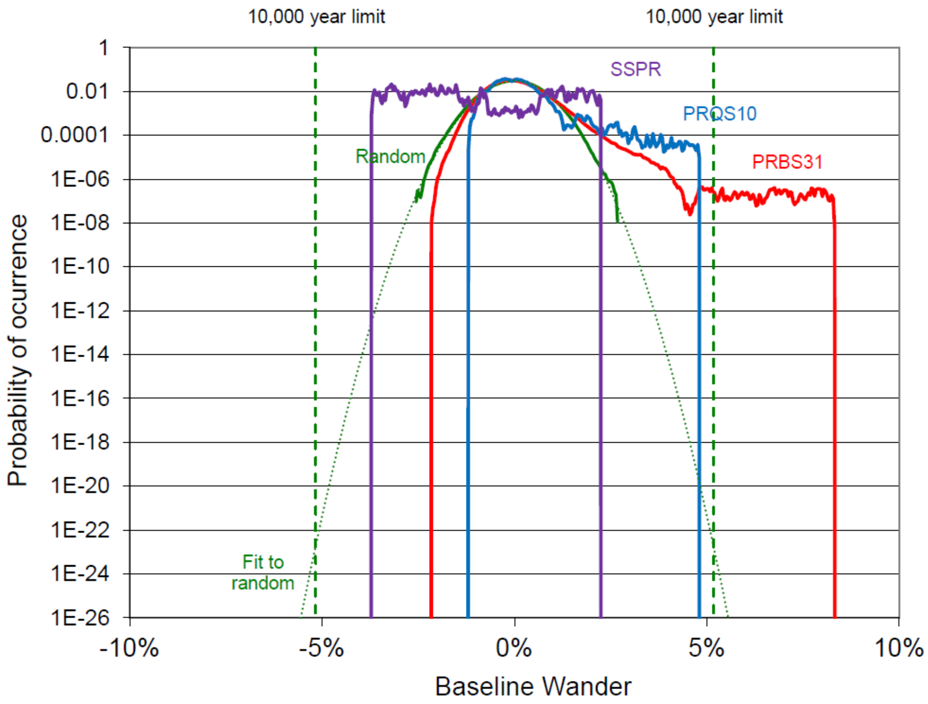




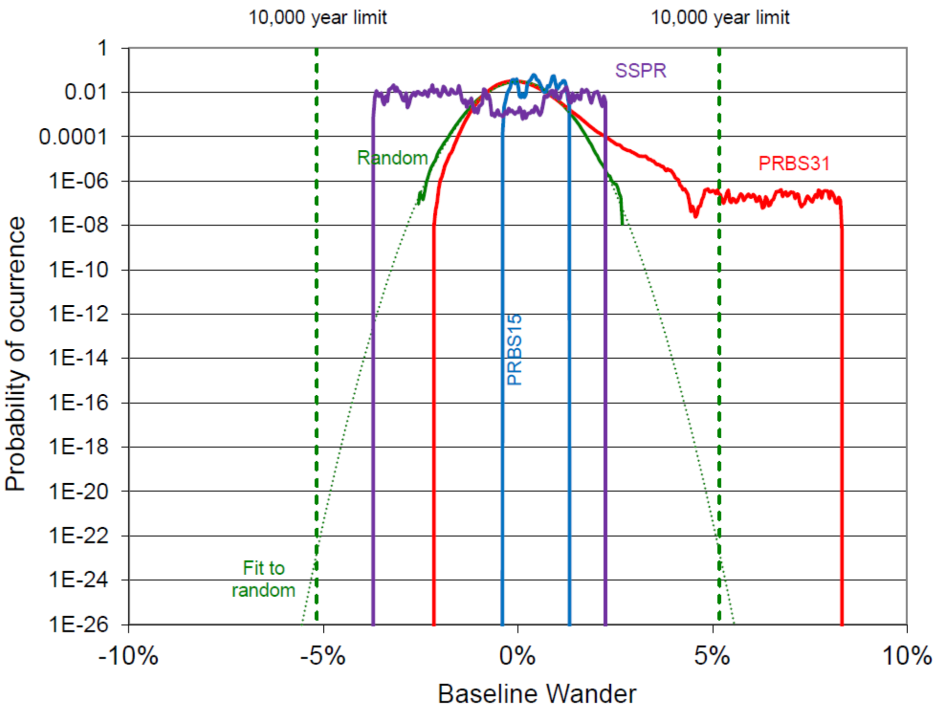




Baseline Wander Characteristics



Source: anslow_01_0915_smf



Source: anslow_3bs_03_0714

Conclusion

- We proposed a PRQS10 test pattern for PAM4 as a natural generalization of PRBS to quaternary sequences
- PRQS patterns can be generated algorithmically using either GF(4) arithmetic based LFSRs or by multiplexing 2 appropriate PRBS patterns
- The proposed PRQS10 pattern has desirable statistical properties for emulating random PAM4 data, provides good baseline wander characteristics, and has modest length ~ 1M symbols