20th IMEKO TC4 International Symposium and
18th International Workshop on ADC Modelling and Testing
Research on Electric and Electronic Measurement for the Economic Upturn
Benevento, Italy, September 15-17, 2014

NOISE WITH COLORED POWER SPECTRUM DERIVED FROM A SINGLE BIT WHITE NOISE INPUT

E. Napoli, G. Lopez, A.G.M. Strollo

DIETI – University of Napoli Federico II, Napoli, Italy, Contact author: ettore.napoli@unina.it

***Abstract*—Noise signals are needed for test and validation of electronic systems and communication channels. Available noise sources are however limited to wide band white noise sources while the arbitrary waveform generators are poorly suited for the generation of long random or pseudo random noise sequences.**
The paper explores the possibility of using a single bit pseudo random sequence, filtered with a FIR filter, for the generation of an analogue noise signal with programmable spectral features.
The results show that the technique is suitable for real time applications. When implemented on a Field Programmable Gate Array device, the technique uses less resources compared with the state of the art.

I. INTRODUCTION

The test of the electronics systems largely depends on the availability of a wide range of precise and reproducible analog signals. As a consequence, the number and the complexity of the electronic systems that generate the test signals increases together with development of the electronic systems.

Noise signals are needed in important applications, such as the test of communication channels, and the test of noise sensitive electronic systems. Unfortunately the available noise source on the market are limited to white noise sources which cannot be controlled in terms of occupied bandwidth and power spectral density. Typically, these noise sources are sold as auxiliary modules of Noise Figure Analyzers or Performance Spectrum Analyzers.

Most of the noise sources proposed in the scientific literature focus on the generation of white noise with Gaussian distribution. With reference to the actual implementation, FPGA (Field Programmable Gate Array), DSP (Digital Signal Processors) solutions, and software implementations have been proposed. Many contributions are oriented to hardware simulation, thus the noise is only synthesized in digital form [1]-[6]. For these contributions the emphasis is on the accurate reproduction of the Gaussian distribution or on the maximum sample rate in the generation. Some contributions, [7]-[10], address the reproduction of digital noise in the analog domain for the development of test equipment and noise generators.

Colored noise signals are needed to model physical and human related mechanisms that take place in modern systems, [8]-[11]. A band pass Gaussian noise generator designed to test wireless receivers is presented in [8]. The proposed system synthesizes the digital noise, which is stored in a flash memory, and then played. The sample rate is very low, e.g. 100 kHz, moreover the duration of the generated noise is limited by the finite flash memory size to a few seconds causing undesired periodicity in the noise. In [9] a 10 MS/s impulsive noise generator, capable of producing arbitrary pulses is proposed and implemented on FPGA. The papers [10],[11] present low speed DSP based systems for the generation of pseudorandom impulsive noise and binary noise, respectively. In [5] a digital circuit for the generation of comb filtered noise implemented using small scale integration logic is proposed. The solutions proposed in [9],[10],[11],[5] do not allow the generation of arbitrarily colored noise.

Recently, a design technique and a real time implementation of an electronic system able to generate non repetitive colored noise with programmable power spectrum has been proposed in [12], [13]. The proposed generator includes a digital section implemented on FPGA and a digital to analog converter (DAC) to play in analogue form the colored noise. The generation frequency is up to 240 MHz producing one noise sample per clock cycle. The Nyquist bandwidth (B) of the generated noise is therefore equal to 120 MHz but can be lowered to B=120 MHz/N, where N is an integer selected by the user in the [1,255] range. The user can also define the spectral pattern of the noise in the Nyquist bandwidth by configuring the digital FIR filter. Upsampling and interpolation techniques reject the spurs that are outside the bandwidth in which the analogue noise is confined. Within the chosen bandwidth, B, the power spectral density of the noise can be defined by the user with a frequency resolution of B/512 Hz (0.2%).

The digital section of the system proposed in [13] exploits a pseudo random number generator with

ISBN-14: 978-92-990073-2-7

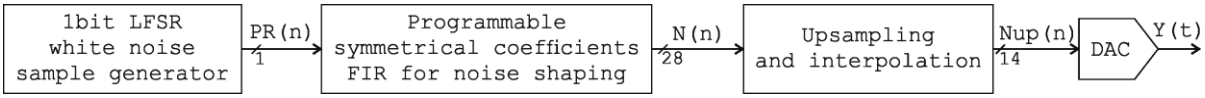

Fig. 1. Block diagram of the proposed programmable noise generator.

repetition period equal to 2^{63}-1 samples that assures non repetitive noise sequences in all practical applications. The pseudo random number generator produces a white noise signal represented on 12 bit. The white noise signal is then filtered using a 1024 tap FIR filter to obtain the sequence with the desired spectral features. Experimental data provided in [13] demonstrate the complete functionality of the designed system.

The circuits proposed in [12], [13], while providing real time operation and good flexibility, require a significant (25%) amount of logic resources of the selected FPGA thus impacting on the cost of the system. In order to reduce the complexity and the logic occupation of the system, in this paper the possibility of using as a white noise source a single bit pseudo random signal is addressed with the emphasis on the spectral features of the generated noise. The schematic block view of the proposed noise generation system is show in Fig. 1. System analysis and simulated results show that the significant improvement in circuit size are possible without impairing the quality of the output digital sequence.

II. WHITE NOISE SEQUENCE.

A. Previous art - PR(n) composed by 12 bit samples

The white noise generator in [13] is composed by 12 Linear Feedback Shift Registers (LFSR), and a Finite State Machine for the boot strap phase that initializes the status of the 12 LFSR sampling the least significant bit of an Analog to Digital Converter.

Each LFSR generates a pseudo random bit and is composed by 63 flip flops and a linear feedback network designed with XOR gates that permits to have a repetition period of 2^{63}-1 samples. The use of 12 LFSR, each one generating one bit of the white noise signal, allows to minimize the correlation between successive samples and ensures maximum randomness of the noise sequence.

The white noise generator produces a sequence of samples where every sample can be seen as an unsigned number in binary format with the weight of the Most Significant Bit (MSB) and the Least Significant Bit (LSB) are 2^0 and 2^{-11}, respectively. As a consequence, the maximum value of a sample is $2\text{-}2^{-11}$ while the minimum value is 0.

Due to the random properties of the generated sequence, the average value of an input samples is very close to 1.

B. Proposed - PR(n) composed by 1 bit samples

In this paper the PR(n) sequence is produced by a single 63 tap LFSR that produces a single bit per clock cycle. The resulting *PR(n)* sequence is demonstrated to have: repetition period of 2^{63}-1 samples; autocorrelation function that is unity for zero delay, and -2^{-63} for any delay greater that one bit. The spectral properties are hence equivalent to those of the 12 bit sequence and correspond to a white noise source. This is shown in Fig. 2 where the average of the spectrum of 10^3 sub sequences, each 10^4 samples long, is reported for both the 12 bit sequence and the single bit sequence. In both cases the resulting spectrum is flat.

When the proposed generator uses the single bit sequence for the generation of the colored noise, the white noise generator produces a sequence of bits that are seen as unsigned numbers that can be either 1 or 0. The average value of the input sequence is then 0.5, a value halved with respect to the average value of the 12 bit sequence. This is expected to have an impact on the magnitude of the generated noise that will be lower when the single bit input is considered.

III. PROPOSED CIRCUIT

The block diagram of the proposed circuit is shown in Fig. 1. The circuit is composed, from left to right, of: the 'White noise generator' that generates a pseudorandom sequence with flat spectrum; the 'Programmable FIR' that confers the desired PSD to the noise sequence; the 'Interpolator' that attenuates the spurs outside the desired bandwidth; the DAC that plays in analogue form the resulting colored noise.

A. White noise generator

It is composed by a single LFSR with 63 taps and a

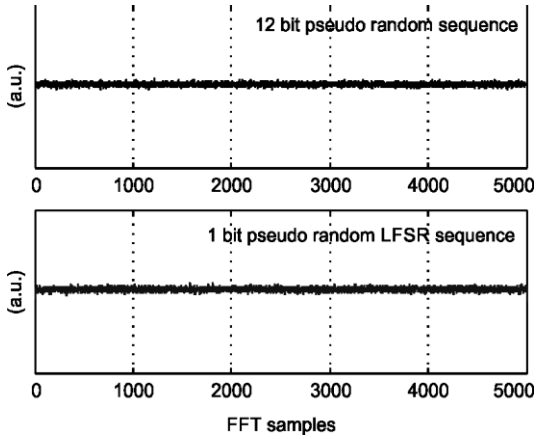

Fig. 2. Averaged spectrum of 10^3 sub sequences taken from the generated white noise. Each sub sequence is 10^4 samples long. The 1 bit pseudo random sequence exhibits a white noise spectrum similar to that of the 12 bit sequence.

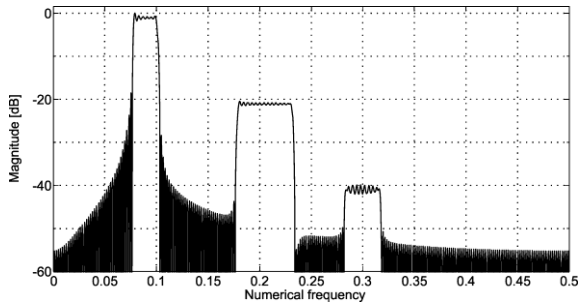

Fig. 3. Spectral response of the FIR filter selected for the test of the system. The spectral response has been calculated in software. The simulated results of the system that refer to this spectral response are shown in Section IV.

Finite State Machine similar to the one described in Section II.A. With respect to the circuit proposed in [13], the white noise generator exploited in this paper requires almost 12 times less logic resources.

The randomness of the seed, in conjunction with the huge duration of the LFSR pseudo random sequences, makes the signal generated by the system, for most of the practical purposes, indistinguishable from a true random noise.

B. Symmetrical coefficients shaping FIR filter

The filter that produces the noise sequence characterized by the assigned power spectral density is a fully pipelined FIR filter with 1024 taps. The coefficients of the FIR filter are calculated using the algorithm described in [12] and are, by construction real and symmetrical. The FIR filter exploits the symmetry implementing a folded structure that firstly sums the symmetrical samples and executes only one multiplication for every symmetrical couple of samples.

The input of the FIR filter is the single bit digital white noise sequence, *PR[n]*, while the output sequence is a digital version of the colored noise named *N[n]*.

The filter coefficients are real valued and represented on 1 bit. This largely simplifies the required logic for the FIR as the binary multiplication of the samples and the coefficients instead of requiring a binary multiplier with 12 bit inputs, is simply implemented with a Boolean AND gate. Other significant improvement in terms of logic resources comes from the logic required to store the 1024 input samples required for the FIR. In [13] this section required 1024*12 = 12288 flip flops, while for the proposed circuit with single bit input only 1024 flip flops are required.

As in [13], the FIR filter of the proposed system is activated once every N clock cycles (with N selectable by the user). In this way it samples the *PR[n]* signal and produces a new sample of the *N[n]* signal once every N clock cycles. The effect of the N input signal is hence the selection of the Nyquist bandwidth of the generated noise. With N=1, the FIR filter works at its maximum speed, f_{Clk}, and the Nyquist bandwidth of the noise is $B=f_{Clk}/2$. With $N>1$ the FIR works at reduced frequency and the Nyquist bandwidth of the noise is $B=f_{Clk}/2N$.

The selectable N value is hence useful to determine the Nyquist bandwidth of the generated noise that can be scaled down at will. In this way a more precise shaping of the noise, exactly for the Nyquist bandwidth required by the user is possible.

C. Interpolator

When the selected N value is higher than 1, the circuit is actually working at a lower frequency with respect to its maximum. It is then possible to take advantage of the spared computational power, to further improve the features of the generated noise. In fact the analogue version of the colored noise is obtained by feeding the digital data stream to a DAC which produces the waveform by changing the output voltage with step variations. Compared with the target smooth waveform, the one played by the DAC is characterized by spurious components in the spectrum originated by the aforementioned step variations [14]-[16]. The spurious components appear as replica of the desired spectrum at multiple values of the Nyquist bandwidth.

In the proposed circuit the *N[n]* sequence is upsampled by the integer factor N and a first order interpolation is carried out thus allowing the attenuation of the aforementioned spurs.

D. Field Programmable Gate Array Implementation data

The proposed noise generator has been implemented on a StratixIV FPGA device (EP4SGX230). FPGA implementation, while not granting the performances of a full custom VLSI implementation, is a fast and low cost technique for the implementation of digital circuits. Furthermore, the performances of modern FPGA devices are good enough to be selected for demanding streaming applications, [17]-[19]. Preliminary results for the logic resource occupation of the circuit is 11597 ALUTs (182400 available) and 18219 flip flop (182400 available). The complete generator occupies no more than 6% of the whole FPGA with a more than fourfold reduction compared to the circuit proposed in [13].

IV. SIMULATION RESULTS

The circuit has been tested using a FIR filter that confines the noise power within the frequency bands with normalized frequencies [0.08,0.10], [0.18,0.23], and [0.28,0.32], assigning relative power levels of 0dB, -20dB, and -40dB, respectively. Fig. 3 shows the filter response calculated in software.

In Fig. 4 the simulated results obtained by setting N=1 and with a clock frequency of 100 MHz are shown. The data are obtained calculating the FFT of a simulated output sequence composed by $2x10^5$ samples. Both the circuit with the 12 bit input and the proposed circuit with the single bit input provide an output noise with a spectrum very close to what is expected and is shown in Fig. 3. Note that, due to the clock frequency of 100 MHz

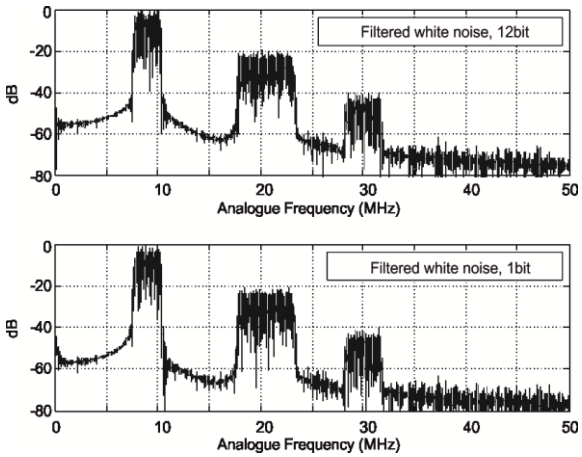

Fig. 4. Spectrum of the generated noise with N=1 and the FIR filter of Fig. 3. Top plot shows the output of the circuit with 12 bit white noise input while the bottom plot shows the output of the circuit with 1 bit white noise input. In both cases the output is in very good agreement with the expected result.

and N=1, the Nyquist bandwidth of the analogue noise signal is 50 MHz and the pass bands for the noise are located at [8 MHz, 10 MHz], [18 MHz, 23 MHz], and [28 MHz, 32 MHz].

A further simulation has been carried out imposing N=10 while keeping the clock frequency to 100 MHz. The Nyquist bandwidth of the generated noise is now equal to 5 MHz and the interpolator circuit can be activated to upsample the output sequence by a factor of 10.

Figure 5 shows the spectrum of the generated sequences in the [0 MHz, 30 MHz] frequency band. The spectrum is in good agreement with the imposed spectral density in the [0 MHz, 5 MHz] band, while attenuated replicas are shown at multiples of the Nyquist bandwidth. The top and the middle plots in Fig. 5 refer to the output of the circuit with the 12 bit input and the proposed circuit with the single bit input, respectively. The two plots are in very good agreement showing that the proposed approach is feasible. In fact the use of a single bit white noise input has a negligible effect on the spectral features of the generated noise but is able to largely reduce the logic resource occupation of the noise generator. The bottom plot shows the effect of the interpolation applied to the output of the proposed circuit. The spurs outside the Nyquist band are attenuated while no effect is visible for the Nyquist bandwidth.

With reference to the amplitude of the generated noise, the use of a single bit, having an average value that is halved with respect to the 12 bit input case, provides an output signal with reduced amplitude. Taking as reference the output sequence generated by the circuit with the 12 bits input, the output sequence produced by the circuit with the single bit input has a maximum absolute value that is reduced by 15%. Such reduction will obviously have an impact on the dynamic range of the sequence that is however minimal if compared with the potential advantages that the single bit input can provide in terms of performances and cost of the final system.

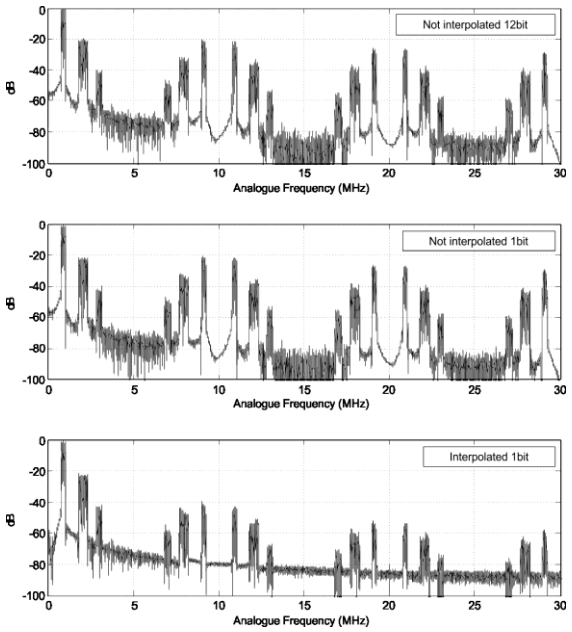

Fig. 5. Spectrum of the generated noise with N=10 and the FIR filter of Fig. 3. The top plot refers to the output of the circuit with 12 bit white noise input without activating the upsampling and interpolation block. The middle plot refers to the output of the circuit with 1 bit white noise input without activating upsampling and interpolation block. The bottom plot refers to the output of the circuit with 1 bit white noise input after the activation of the upsampling and interpolation block.

V. CONCLUSIONS

In this paper the possibility of using as a single bit white noise source to generate a noise with determined power spectral density is addressed. Simulated results show that the spectral features of the generated noise can be very well controlled by the user. The proposed system allows to select the Nyquist bandwidth of the generated noise and implements the operations of upsampling and interpolation in order to attenuate the spurs that reside outside the Nyquist bandwidth.

REFERENCES

[1] D.U. Lee, et al., "A Gaussian noise generator for hardware based simulations", IEEE Trans. on Computers, vol. 53, No. 12, Dec. 2004, pp. 1523-1534.

[2] D.U. Lee, et al., "A hardware Gaussian noise generator using the Box-Muller method and its error analysis", IEEE Trans. on Computers, vol.55, no.6, pp.659-671, June 2006, doi: 10.1109/TC.2006.81

[3] D.U. Lee, Wayne Luk, John D. Villasenor, Guanglie Zang, Philip H. W. Leong, "A hardware Gaussian noise generator using the Wallace method", IEEE Trans. on Very Large Scale Integration (VLSI), vol. 13, No. 8, August 2005, pp. 911-920.

[4] Yu Jihan Zhao, Guojing Zhang Zhang, "Design and FPGA implementation of digital noise generator based on superposition of Gaussian process," IEEE 13th Int. Conf. on Communication Technology (ICCT), 2011, pp. 277-280, 25-28 Sept. 2011.

[5] I. Paraskevakos, V. Paliouras, "A flexible high-throughput hardware architecture for a gaussian noise generator," 2011 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), pp. 1673-1676, 22-27 May 2011, doi:10.1109/ICASSP.2011.5946821.

[6] "Additive White Gaussian Noise (AWGN) Core v1.0," Xilinx Product Specification, 2002.

[7] C. Imperiale, "Programmable uniform-Gaussian, quasi-white noise generator and data acquisition-processing system for electrophysiology," Measurement, Vol.20, n.3, March 1997, pp. 149-163.

[8] Y. Linn, "Generation of bandpass Gaussian noise with applications to complete Built In Self Test in wireless communications receivers," IEEE Latin-American Conference on Communications, LATINCOM'09, pp.1-6, 10-11 Sept. 2009.

[9] K. Yamane, T. Shinozuka, K. Ohnuma, "Pseudo-noise generator with arbitrary APD, PDD and PSD," IEEE Int. Symp. on Electromagnetic Compatibility, 2000. vol.1, pp.471-476, 2000.

[10] R.M. Rodriguez-Osorio, et al., "A DSP-Based Impulsive Noise Generator for Test Applications", IEEE Trans. on Industrial Electronics, vol.54, no.6, pp.3397-3401, Dec. 2007, doi:10.1109/TIE.2007.903956

[11] Z. Dokoupil, K. Bartusek, E. Gescheidtova, "Noise Generator with DSP96002," Third Int. Conf. on Systems, 2008. ICONS 08, pp. 296-299, 13-18 April 2008, doi:10.1109/ICONS.2008.18

[12] L. Angrisani, M. D'Arco, E. Napoli, "Programmable power spectral density noise source," 19th IMEKO T4 Symp. and 17th IWADC Workshop Advances in Instrumentation and Sensors Interoperability, pp.759-764, July 18-19, 2013, Barcelona, Spain.

[13] E. Napoli, M. D'Arco, P. Di Cosmo, M. Genovese, A.G.M. Strollo, "FPGA Based System for the Generation of Noise with Programmable Power Spectrum," Proc. 2014 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1300-1303, Melbourne, 1-5 June, 2014, doi: 0.1109/ISCAS.2014.6865381

[14] L. Angrisani, M. D'Apuzzo, M. D'Arco, "Modelling timing jitter effects in digital to analog converters", IEEE Trans. on Instrumentation and Measurement, Vol.58, n.2, Febr. 2009, pp. 330-336.

[15] M. D'Arco, E. Napoli, M. Vadursi, "Synthesis and generation of critical waveforms by means of AWG," 2012 IEEE I2MTC - International Instrumentation and Measurement Technology Conference, Proceedings, art. no. 6229558, pp. 2543-2547, doi:10.1109/I2MTC.2012.6229558

[16] M. D'Apuzzo, M. D'Arco, L. Liccardo, M. Vadursi, "Modelling DAC output waveforms", IEEE Trans. on Instrumentation and Measurement, vol.59, n.11, Nov. 2010, pp. 2854-2862

[17] E. Napoli, M. Genovese, "FPGA-based architecture for real time segmentation and denoising of HD video", Jour. of Real Time Image Processing, 2011, pp. 1-13, doi:10.1007/s11554-011-0238-1

[18] M. Genovese, E. Napoli, D. De Caro, N. Petra, A.G.M. Strollo, "Analysis and comparison of Direct Digital Frequency Synthesizers implemented on FPGA", Integration, the VLSI journal, vol.47, n.2, March 2014, pp. 261-271, doi:10.1016/j.vlsi.2013.09.001

[19] M. Genovese, E. Napoli, "ASIC and FPGA Implementation of the Gaussian Mixture Model Algorithm for Real-Time Segmentation of High Definition Video", IEEE Trans. on Very Large Scale Integration (VLSI) Systems, vol.22, no.3, pp.537-547, March 2014, doi:10.1109/TVLSI.2013.2249295