# NONUNIFORM SAMPLING DRIVER DESIGN FOR OPTIMAL ADC UTILIZATION

*F. Papenfuß* [1], *Y. Artyukh* [2], *E. Boole* [2], *D. Timmermann* [1]

[1] Inst. of Applied Microelectronics and Computer Science, University of Rostock, Germany,
EMail: frank.papenfuss@etechnik.uni-rostock.de, dirk.timmermann@etechnik.uni-rostock.de
[2] Institute of Electronics and Computer Science (IECS), Riga, Latvia,
EMail: artyukh@edi.lv, buls@edi.lv

## ABSTRACT

Deliberate nonuniform sampling promises increased equivalent sampling rates with reduced overall hardware costs of the DSP system. The equivalent sampling rate is the sampling rate that a uniform sampling device would require in order to achieve the same processing bandwidth. Equivalent bandwidths of realizable systems may well extend into the GHz range while the mean sampling rate stays in the MHz range. Current prototype systems (IECS) have an equivalent bandwidth of 1.6GHz at a mean sampling rate of 80MHz, achieving 40 times the bandwidth of a classic DSP system that would operate uniformly at 80MHz (cf. [1]). Throughout the literature on nonuniform sampling (e. g. [2] and [3]) different sampling schemes have been investigated. This paper focuses on nonuniform sampling schemes optimized for fast and efficient hardware implementations. To our knowledge this is the first proposal of an efficient nonuniform sampling driver (SD) design in the open literature.

## 1. INTRODUCTION

Nonuniform sampling circumvents traditional sampling limitations requiring a sampling rate of at least twice the input signal bandwidth. A special unit, the SD, generates the sampling pulse train used to digitize the analog signal. To realize a SD in digital circuits obviously a synchronous design is desirable keeping the design process simple. According to sampling theory a straightforward implementation of a SD produces sampling instances deliberately jittered around a fixed system clock. A pseudo random number generator (PRNG) generates numbers passed to a digitally controllable delay line (DCDL) delaying pulses produced by a central controlling unit. Though each digital circuit driving an ADC performs, strictly speaking, periodic sampling with jitter (due to phase noise) a simple SD realization depicted in Fig. 1 does it deliberately. One can consider the time axis being separated into time slots having system clock duration $T_{clk}$. Inside each slot a sampling instance $t_k$ is produced. For the SD design to be successful it must realize a sampling instance with equal probability anywhere in the *k*-th time slot in order to achieve a constant probability to produce sampling points anywhere at the time axis. Failure to do so will result in an undesired spectrum of the sampled signal containing spurious frequencies (cf. [4]). Therefore, sampling algorithm, architecture and SD hardware implementation have to be carefully aligned to obtain maximum benefit from nonuniform sampling.

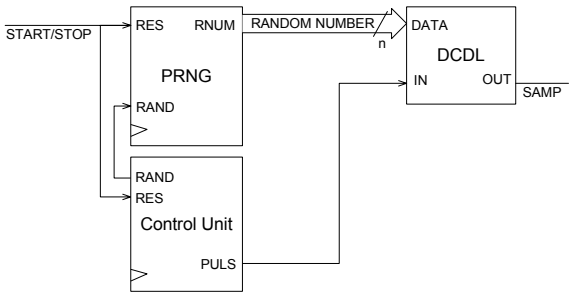


Fig. 1: General synchronous SD building block.

This is due to the convolution of sampling process spectrum and signal spectrum. The process is illustrated in Fig. 3 showing the spectrum of a signal with one component. The figure is added to stress the importance of matching the probability density function (PDF) of a sampling instance to the SD system clock period.

Real circuits will not produce sampling points with infinite accuracy but will realize time increments of so called time quantum size $T_Q$. This renders the sampling instance PDF discrete (see Fig. 2). The equivalent sampling rate is given by the inverse time quantum. The limited amount of time increments in a matched time slot is expressed by the system clock period to time quantum ratio *M*

$$M = \frac{T_{clk}}{T_Q}. \qquad (1)$$

This is a key parameter of a sampling driver since it represents the factor by which the processing bandwidth of the digital system is increased. It is convenient to keep

0-7803-7762-1/03/$17.00 ©2003 IEEE

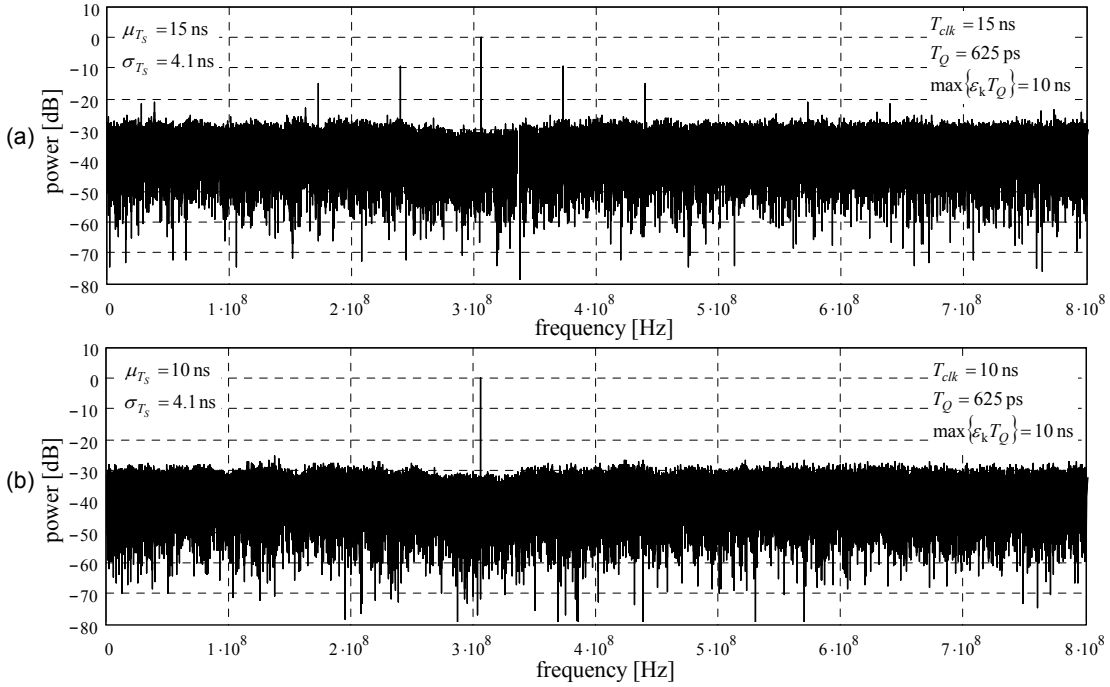


Fig. 3: Power density spectra (via DFT) of a signal containing exactly one frequency at 305MHz. The PDF of a sampling instance is (a) not matched and (b) matched to SD system clock period.

$M$ at a power of two to fully utilize the bits of the data vector entering the DCDL. The process of sampling instant generation is well known as periodic sampling with jitter (cf. [1]) and can be described by

$$t_k = kT_{clk} + \varepsilon_k T_Q \qquad k, \varepsilon_k \in \mathbf{N} \qquad 0 \le \varepsilon_k < M \,, \tag{2}$$

where $\varepsilon_k$ is a pseudo random number produced by the PRNG at the $k$-th time slot. Unfortunately equation (2)

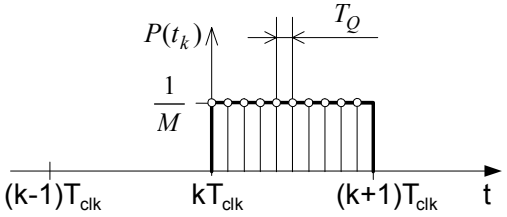


Fig. 2: PDF of sampling instances at k-th time slot.

has a bad property. Two successive samples may be separated by only the time quantum $T_Q$. It therefore seems to be desirable to define a setup of a random experiment that will serve to assess the quality of generated sampling sequences. Let $T_s$ be the time between consecutive samples, the intersample time

$$T_s = t_k - t_{k-1} \,. \tag{3}$$

Thus, the intersample time is a derived random variable. For convenience we define the Laplacian random experiment $E_0$

$$\begin{aligned}
&\Omega_0 = \{\omega_0^{(0)}, \omega_1^{(0)}, ..., \omega_n^{(0)}, ..., \omega_{M^2-1}^{(0)}\} \\
&\Omega_0 = \{(0,0),(0,1),...(i_n, j_n),...,(M-1, M-2),(M-1, M-1)\} \\
&i_n, j_n, n \in \mathbf{N}; \quad 0 \le i_n, j_n, n < M \\
&\omega_n^{(0)} \equiv (i_n, j_n) \equiv \quad (\varepsilon_{k-1} = i_n \quad and \quad \varepsilon_k = j_n) \\
&P(\omega_n^{(0)}) = \frac{1}{M^2}
\end{aligned} \,, \tag{4}$$

where $(i_n, j_n)$ denotes the event that $\varepsilon_{k-1}$ takes on value $i_n$ and $\varepsilon_k$ takes on value $j_n$. It is easy to see that there are $M^2$ such events. Assuming that both $\varepsilon_{k-1}$ and $\varepsilon_k$ have uniform distribution and are statistically independent, it immediately follows that the events $(i_n, j_n)$ have equal probability $1/M^2$. Observing that, given (2) and (3) $T_s$ will never become larger than $2M$ one can define a different random experiment $E_1$ with a set $\Omega_1$ of $2M$ elementary events

$$\begin{aligned}
&\Omega_1 = \{\omega_0^{(1)}, \omega_1^{(1)}, ..., \omega_l^{(1)}, ..., \omega_{2M-1}^{(1)}\} \\
&\Omega_1 = \{0, T_Q, 2T_Q, ..., lT_Q, ..., (2M-1)T_Q\} \qquad l \in \mathbf{N}, 0 \le l < 2M \\
&\omega_l^{(1)} \equiv T_s = lT_Q
\end{aligned} \,, \tag{5}$$

where the $l$-th event in $\Omega_1$ denotes the event that $T_s$ takes on value $lT_Q$. Unlike the events in $\Omega_0$ the events in $\Omega_1$ do not occur with equal probability. However, these probabilities can be obtained from events in $E_0$ by

$$P(\omega_l^{(1)}) = \sum_{\omega_n^{(0)} \in \omega_l^{(1)}} P(\omega_n^{(0)}) = \frac{1}{M^2} \sum_{\omega_n^{(0)} \in \omega_l^{(1)}} 1 \,. \tag{6}$$

$$n, l \in \mathbf{N}; \quad 0 \le n < M^2; \quad 0 \le l < 2M$$

Using (1), (2) and (3) we can say when an event in $\Omega_0$ is said to be a favorable event in terms of an event in $\Omega_1$

$$\omega_n^{(0)} \in \omega_l^{(1)} \quad if \quad l = j_n + M - i_n \,. \tag{7}$$

Applying (6) and (7) the probabilities for all events in $\Omega_1$ and hence the discrete PDF of $T_s$ can be estimated. It is sketched in Fig. 4.

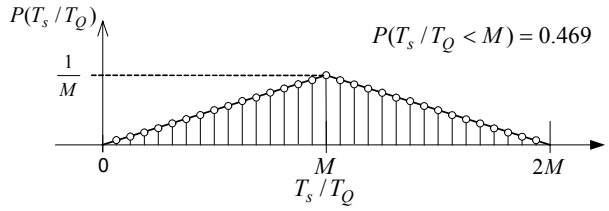


Fig. 4: PDF of intersample time.

In an optimal SD design for full ADC utilization the system clock period $T_{clk}$ is usually matched to the minimum conversion time of the attached ADC

$$T_{clk} = \min\{T_{ENCODE}\} \,. \tag{8}$$

This is justified by the design decision to operate the sampling driver also in a uniform mode ($\varepsilon_k$ constant) in which case the ADC should be fully utilized. Hence the intersample time constraint

$$\begin{aligned} t_k - t_{k-1} &\geq T_{clk} \\ T_s &\geq MT_Q \end{aligned} \tag{9}$$

must always be met. Given (6) and (7) we can calculate the probability that (9) is not met in case of this straightforward design. It is about 47% and we conclude that such a straightforward design is not usable as a sampling driver.

## 2. PHASE SHIFTING

To avoid too short intersample times we propose a different sampling scheme that introduces phase shifts at times when consecutive samples occur too close for the ADC to handle. The modified sampling scheme can be described recursively as described in (10). Only the control unit of the design shown in Fig. 1 needs to be changed. A phase shift of the sampling pulse means deferring it one SD system clock period (i. e. 360°).

$$t_k = t_{k-1} - \varepsilon_{k-1}T_Q + T_{clk} + \left\{ \begin{matrix} 0 & if \quad \varepsilon_{k-1} < \dfrac{M}{2} \\ T_{clk} & otherwise \end{matrix} \right\} + \varepsilon_k T_Q \,. \tag{10}$$

$$k, \varepsilon_k \in \mathbf{N}; \quad 0 \leq \varepsilon_k < M$$

In the modified design the control unit of the sampling driver constantly checks the random numbers that have been and are produced by the PRNG and introduces phase shifts described by (10). Deliberate phase shifting fundamentally changes the sampling scheme. Periodic sampling with deliberate jitter becomes additive random sampling. A single sampling instance has still an evenly distributed PDF but is now stretched over two SD system clock periods because of the introduced phase shift.

A well-known property of the additive random sampling scheme is that it produces a constant valued sampling point density function (SPDF) after a transient phase. This property (based on the central limit theorem) is extensively treated in [1]. The PDF of the derived random variable $T_s$ looks different than in the previous Section 1. Using (3) and (10) one can write

$$T_s = \left( M - \varepsilon_{k-1} + \left\{ \begin{matrix} 0 & if \quad \varepsilon_{k-1} < \dfrac{M}{2} \\ M & otherwise \end{matrix} \right\} + \varepsilon_k \right) T_Q \,. \tag{11}$$

$$k, \varepsilon_k \in \mathbf{N}; \quad 0 \leq \varepsilon_k < M$$

We use the same method as in Section 1 to determine the probabilities $P(T_s/T_Q = l)$ a priori. The result is depicted in Fig. 5. The probability to produce intersample times less than $MT_Q$ is around 11% and thus non-zero.

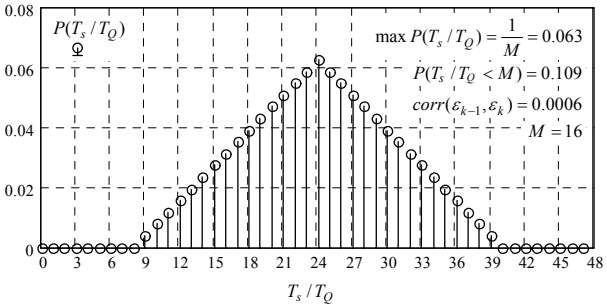


Fig. 5: PDF of intersample time with phase shift.

The property of such a sampling driver is certainly better but would still be too demanding for an attached ADC being operated at its limits as described above. A solution will be presented in the next Section.

## 3. RANDOM NUMBER CORRELATION

When generating pseudo random numbers maximum length linear feedback shift registers (LFSR) are commonly used (see [5] and [6]). Using a slice of bits from a longer LFSR one can write for consecutive random numbers $\varepsilon_{k-1}$ and $\varepsilon_k$

$$\varepsilon_k = (2\varepsilon_{k-1} + \tau_k) \bmod 2^n \qquad k, \varepsilon_k \in \mathbf{N} \qquad \tau_k \in \{0,1\}, \tag{12}$$

where $\tau_k$ is a binary random number assumed to be evenly distributed and $n$ is the dimension of the vector passed as random number to the DCDL. It is important to note that, given (12), the probabilities for events in $\Omega_0$ are no longer evenly distributed. Through computer simulation the distribution of $T_s$ was determined given that two successive random numbers are correlated as defined in

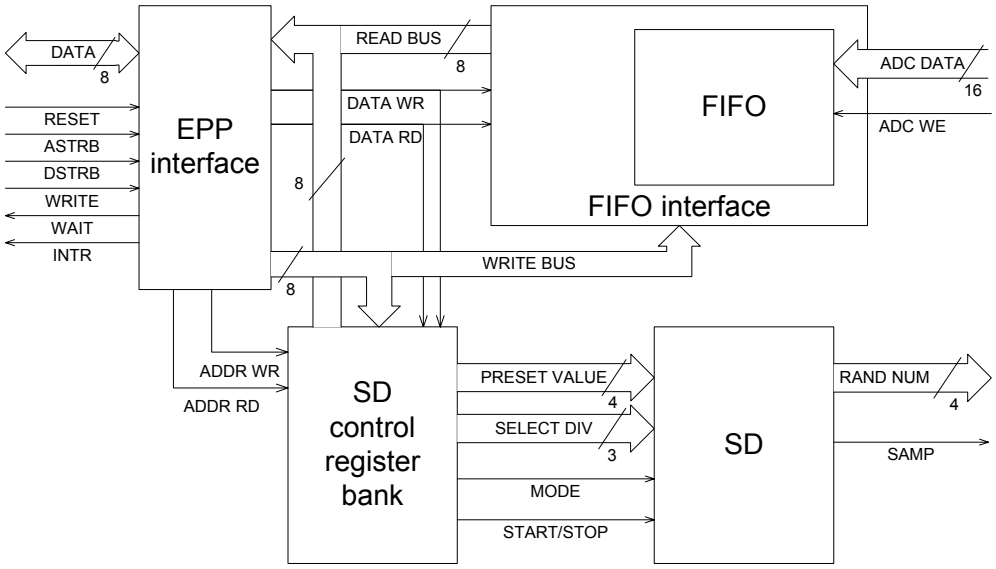


Fig. 6: Overall sampling driver architecture (main units).

(12). The simulation results clearly reveal that the PDF of the intersample time now satisfies the constraints (Fig. 7).

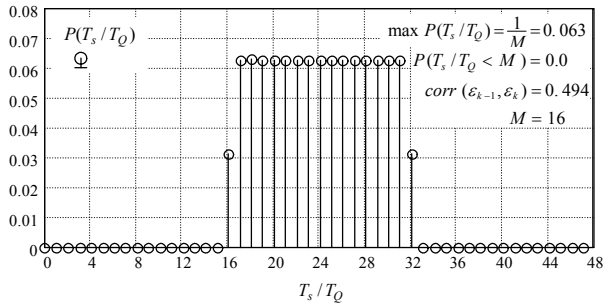


Fig. 7: PDF of intersample time with phase shift and correlated random numbers.

## 4. SD INTEGRATION

Obviously the SD cannot stand alone. It has to be integrated into a larger architecture. In Fig. 6 an integration of a SD design into a sample recording architecture is presented. The extended parallel port (EPP) interface is used to program the SD. Functions such as starting and stopping as well as uniform and nonuniform sampling are realized. Sampled data is buffered in the FIFO and read via the EPP. The design was tested and implemented in a FPGA using VHDL.

## 5. CONCLUSIONS

In this paper we have derived an efficient sampling algorithm for deliberate additive random sampling. The algorithm is well suited to fully utilize the minimum conversion time of an ADC. Cost reduction is achieved because cheaper ADCs can be used instead of expensive ones while processing the same or even higher bandwidth than the comparable traditional system. Alternatively it is possible that a system utilizing the suggested design is used to process GHz signals fully digital with higher bandwidth and/or resolution than possible today in a traditional design using a cutting edge ADC.

It was shown that introducing deliberate correlation into the random number generation process is beneficial. It will create exactly the sampling pulse train that best utilizes the ADC. In our case the introduced correlation coefficient of consecutive random numbers is 0.5.

## 6. REFERENCES

[1] Y. Artyukh, A. Ribakov, V. Vedin, “Evaluation of Pseudorandom Sampling Processes”. In *Proceedings of the 1997 Workshop on Sampling Theory and Applications,* pp. 361-363, SampTA ‘97.

[2] I. Bilinskis and A. Mikelsons, *Randomized Signal Processing*, Prentice Hall International, 1992, ISBN 0137510748.

[3] J. J. Wojtiuk, *Randomised Sampling for Radio Design*, PhD Thesis, University of South Autralia, 2000

[4] D. M. Bland and A. Tarczynski, "The effect of sampling jitter in a digitized signal". In *Proceedings of the 1997 IEEE International Symposium on Circuits and Systems,* Vol. 4, pp. 2685-2688, ISCAS ‘97.

[5] M. Abamovici, M. A. Breuer and A. D. Friedman, *Digital Systems Testing and Testable Design*, IEEE Press, 1990, ISBN 0780310624.

[6] P. Alfke, “Efficient Shift Registers, LFSR Counters, and Long Pseudo-Random Sequence Generators”, *XILINX XAPP 052,* July 7, 1996 (Version 1.1)