Behavioral Modeling of Time-Interleaved ADCs using MATLAB

Conference Paper · October 2003

Source: OAI

2 authors:

Christian Vogel
FH JOANNEUM University of Applied Sciences
96 PUBLICATIONS **1,755** CITATIONS
SEE PROFILE

Heinz Koeppl
Technical University of Darmstadt
302 PUBLICATIONS **3,351** CITATIONS
SEE PROFILE

Behavioral Modeling of Time-Interleaved ADCs using MATLAB

Christian Vogel[1] and Heinz Koeppl[2]

Christian Doppler Laboratory for Nonlinear Signal Processing

Institute of Communications and Wave Propagation

Graz University of Technology, Austria

[1]Phone: (+43) 316 873-4443, email: vogel@inw.tugraz.at [2]Phone: (+43) 316 873-4434, email: koeppl@inw.tugraz.at

***Abstract*—We introduce a behavioral MATLAB simulation of a time-interleaved ADC. The simulation includes the main error sources of a time-interleaved ADC and works with continuous-time input signals and continuous-time input filters. We clarify how to implement the signal generation and how to connect the continuous-time and the discrete-time part of the time-interleaved ADC with high calculation precision. By using the simulation results, we treat the problem of channel mismatches, demonstrate the problems of bandwidth mismatches, and show the impact of nonlinearities.**

I. Introduction

New telecommunication technologies need high-speed analog-to-digital converters (ADCs). The employment of a time-interleaved architecture is a well-known strategy to achieve this goal.

Such an architecture consists of M channel ADCs, which have the same sampling rate but different phases, as if they were a single converter operating at an M times higher sampling rate [1]. The time-interleaved architecture is illustrated in Fig. 1, where each channel ADC operates with a sampling frequency of f_s/M.

The basic concept of a time-interleaved ADC is independent from the channel ADC technology. In principle, all known ADC technologies are applicable as channel ADCs.

II. System Modeling

In this section we introduce our behavioral simulation model. We consider the main error sources of a time-interleaved ADC, the signal generation, and the channel model.

A. Time-Interleaved ADC Model

We decided to use MATLAB as a simulation environment due to the wide range of toolboxes, which allow a fast and flexible programming of required functionality.

The simulation model should include the most important signal processing error sources of a time-interleaved ADC. The three main error sources degrading the spurious free-dynamic range (SFDR) and the signal-to-noise and distortion ratio (SINAD) of a time-interleaved ADC are offset, gain, and timing mismatches [2]. Gain mismatches are the differences among the gains and offset mismatches are the differences among the offsets of the channel ADCs. Timing mismatches are the deterministic deviations between the nominal sampling time and the real sampling time for each channel. Another important error source is timing jitter, which is, however, not restricted to time-interleaved ADCs. In contrast to the timing mismatch, it is the stochastic deviation from the nominal sampling period for each sample [3]. Through analysis of multirate systems we also know that bandwidth mismatches lower the SINAD and the SFDR [4]. Conventional ADCs have static and dynamic nonlinearities, which have to be considered in time-interleaved ADCs as well.

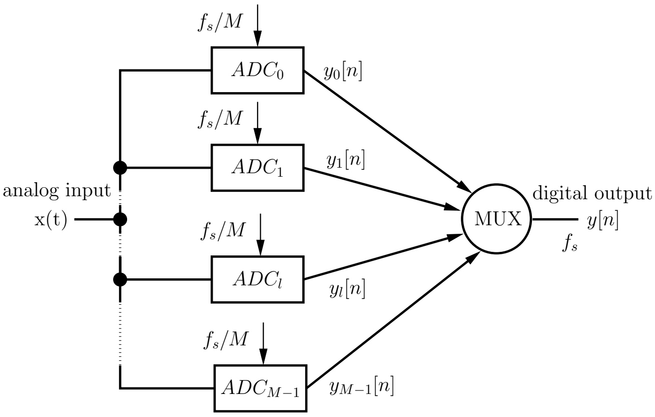

Fig. 1. Time-interleaved ADC with M channels.

B. Signal Generation

The first problem, which has to be solved, is the input signal generation. It is a non-trivial task, since we want to simulate the sampling of an arbitrary continuous-time signal with a discrete environment. To define a general set of P input signals $x_p(t)$ we use the coefficients of the Fourier series to define each input signal as

$$x_p(t) = a_{p,0} + \sum_{k=1}^{+\infty} a_{p,k} \sin(2\pi f_p k t + \varphi_{p,k}), \qquad (1)$$

where the total input signal is

$$x(t) = \sum_{p=0}^{P-1} x_p(t). \qquad (2)$$

This signal representation spans a sufficiently large signal space that allows us to simulate a broad class of possible input signals and can easily be applied to continuous-time filter operations. Moreover, the simulation can imitate the sampling process with high calculation precision.

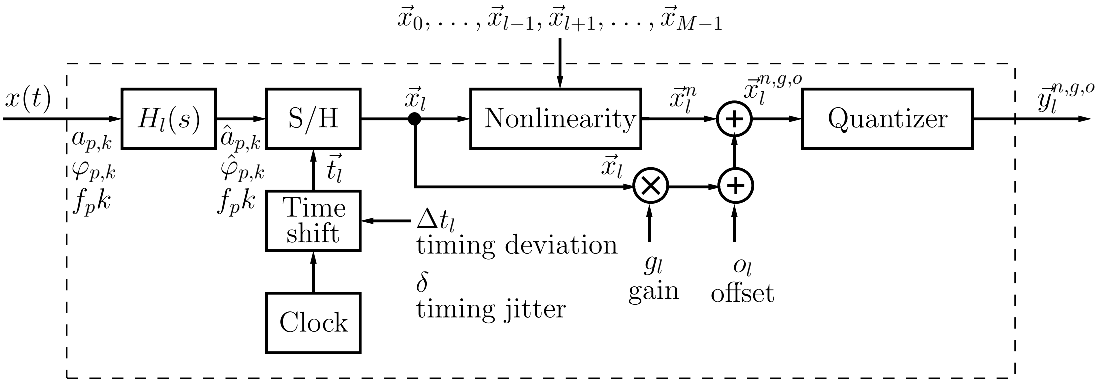

Fig. 2. Behavioral simulation model of the lth channel ADC.

C. Channel ADC Model

In order to achieve the described error effects from Sec. A we developed a behavioral channel model, which is presented in Fig. 2.

C.1 Input Filter

The input signal $x(t)$ is sent through a continuous-time filter $H_l(s)$. The transfer function of the filter can be specified by its Laplace transform. An insightful example is the first-order RC-low pass. Its transfer function is

$$H_l(s) = \frac{1}{1 + s\alpha_l} \tag{3}$$

where $\alpha_l = R_l C_l$ to get the connection to a real world device. The filter operates on the amplitude coefficients $a_{p,k}$ and phase coefficients $\varphi_{p,k}$ of the Fourier series. For each given frequency component f_pk these two coefficients are modified according to the transfer function and become $\hat{a}_{p,k}$ and $\hat{\varphi}_{p,k}$.

C.2 Sample and Hold

In the S/H block the analog signal described by the Fourier coefficients is converted to discrete time values. Therefore, sampling instants must be generated. Through error analyses we know that all instants possess deterministic timing mismatches Δt_l and stochastic timing jitter δ. Hence, the time vector for the lth channel is

$$\begin{aligned}\vec{t}_l = [\ & lT_s + \Delta t_l + \delta_{l,0}, (M+l)T_s + \Delta t_l + \delta_{l,1}, \ldots, \\ & (kM+l)T_s + \Delta t_l + \delta_{l,k}, \ldots, \\ & ((K-1)M+l)T_s + \Delta t_l + \delta_{l,K-1}]^T \end{aligned} \tag{4}$$

where M is the number of ADCs and K the number of samples. The sampling process itself is the evaluation of the Fourier series, given by the coefficients $\hat{a}_{p,k}$, $\hat{\varphi}_{p,k}$, and f_pk, and the time vector $\vec{t}_l$

$$\vec{x}_l = x(\vec{t}_l). \tag{5}$$

C.3 Nonlinearity

In the next block, the sampled signal runs through a static or dynamic nonlinearity. The nonlinearity is realized as an additive component of an arbitrary nonlinear mapping $\mathcal{F}_l$

$$\vec{x}_l^n = \vec{x}_l + \mathcal{F}_l(\vec{x}_0, ..., \vec{x}_l, ..., \vec{x}_{M-1}). \tag{6}$$

We have to consider samples from all channels to be able to simulate nonlinearities of the S/H circuits. With this approach we can easily simulate given static integral nonlinearity (INL) specifications, since the maximum of $\mathcal{F}$ is at the same time the INL as defined in [5].

In order to implement the nonlinearity computationally efficient we have to reduce the complexity of the nonlinear mapping $\mathcal{F}_l$. First, we rewrite the input of the mapping in a more convenient form

$$\vec{x} = \mathbf{X}(:), \tag{7}$$

where

$$\mathbf{X} = [\vec{x}_0, ..., \vec{x}_l, ..., \vec{x}_{M-1}]^T. \tag{8}$$

Afterwards, we assume causality and a finite memory length of J samples and therefore obtain the input sequence

$$\vec{x}[n]_J = (x[n], x[n-1], \ldots, x[n-J+1]). \tag{9}$$

To further reduce the complexity we utilize two different approaches. The first approach quantizes the input sequence, whereas the second one approximates the nonlinear mapping $\mathcal{F}_l$ with radial basis functions (RBF). The effect of the nonlinear mapping $\mathcal{F}_l^Q$ is illustrated in Fig. 3. There we have used a finite memory of $J = 2$ and a quantization resolution of $Q = 8$bits. In general, we need Q^J entries to implement the mapping with the help of a J-dimensional table.

In the second approach the input sequences $\vec{x}[n]_J$ are mapped to scalar output samples using a weighted sum

$$\mathcal{F}_l^{RBF}(\vec{x}[n]_J) = \sum_{k=1}^{N_b} w_k \phi_k(\vec{x}[n]_J), \tag{10}$$

where

$$\phi_k(\vec{x}[n]_J) = \exp\left[-\frac{1}{2\sigma^2}||\vec{x}[n]_J - \vec{c}_k||^2\right] \tag{11}$$

and $\vec{c}_k$ is the constant center of the kth basis function $\phi_k(\vec{x}[n])$. An RBF network approximation is shown in Fig. 4, where the relevance vector machine algorithm [6] has been used to estimate the weights w_k of the RBF network.

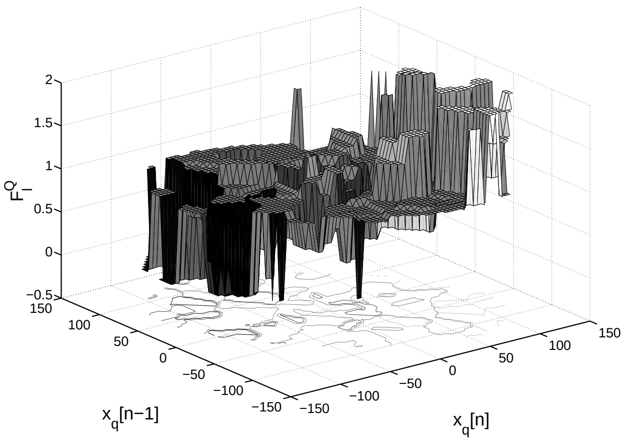

Fig. 3. Illustration of the nonlinear mapping $\mathcal{F}_l^Q$ with a memory of $J = 2$ and a quantization resolution of $Q = 8$bits.

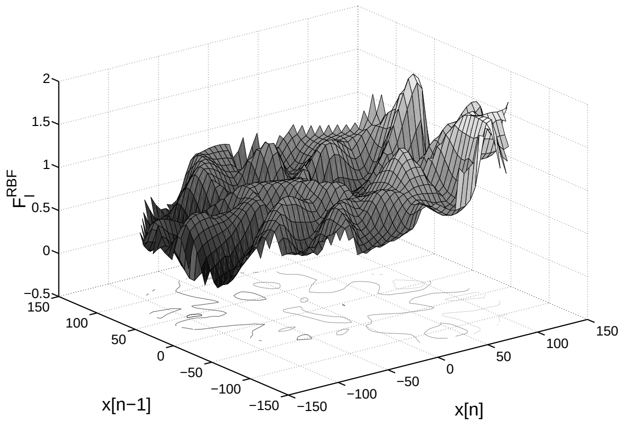

Fig. 4. RBF network approximation of the nonlinear mapping $\mathcal{F}_l$ with a memory of $J = 2$.

C.4 Gain and Offset

The block parallel to the nonlinearity multiplies the signal $\vec{x}_l$ with the gain g_l. Thereby a static gain mismatch can be simulated. After that, the offset o_l is added to obtain offset mismatches. To sum up, we modify the signal as

$$\vec{x}_l^{n,g,o} = \vec{x}_l \cdot g_l + o_l + \vec{x}_l^{n}. \tag{12}$$

Because of the nonlinearity the gain and the offset block are redundant, however, since the mismatches of these parameters are often investigated, we used extra blocks.

C.5 Quantizer and Coder

In the next step, the signal is quantized in order to get the digital signal, i.e. a discrete-time and discrete-amplitude signal,

$$\vec{y}_l^{n,g,o} = \mathcal{Q}\{\vec{x}_l^{n,g,o}\}. \tag{13}$$

Finally, all channel outputs are merged into one output signal in a time-interleaved way and coded afterwards.

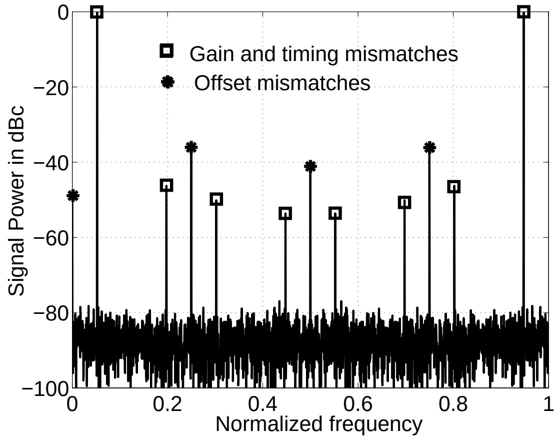

Fig. 5. Simulation with offset, gain and timing mismatches.

III. Simulation Results

In the following we will show simulation results of the most important error sources, which show the abilities of the environment. In Fig. 5 we see the output spectrum of a time-interleaved ADC consisting of four channel ADCs with offset, gain, and timing mismatches. Additionally, the time-interleaved ADC produces timing jitter. Beside the input signal, a sine wave given as $x(t) = \sin(2\pi f_0 t)$, we see additional spurious lines in the spectrum. The spectral lines appear at $k\Omega_s/M$ for offset mismatches and at $\pm\Omega_0 + k\Omega_s/M$ for timing and gain mismatches. Although we use a 10bit resolution and 1024 samples for this simulation, we see a noise floor, which is remarkably above 89dBc[1] of an ideal 10bit ADC [5]. This is due to the timing jitter, which increases the noise floor.

Another important error occurring in time-interleaved ADCs is the bandwidth mismatch. In order to simulate this error, we assume low-pass characteristics for the input filters and an ideal cut-off frequency (-3dB), which is five times higher than the sampling frequency. Each input filter has a cut-off frequency that deviates from the ideal one. These deviations are assumed to be Gaussian distributed with a standard deviation of 0.1. The transfer functions of the described filters are shown in Fig. 6. In Fig. 7 we see the output spectrum of a time-interleaved ADC with these input filters. We notice additional spectral lines similar to the case with static gain and timing mismatches.

Finally, we consider static nonlinearities. In [7] the authors show that the INL and the DNL of the time-interleaved ADC are smaller than the INL and the DNL of the worst channel ADC. In the output spectrum of a time-interleaved ADC, Fig. 8, additional spectral lines arise due to the INL mismatch between the channels. Nevertheless, the whole noise and distortion energy decreases in a time-interleaved environment as simulations show. In Fig. 9 we see the SINAD of a time-interleaved ADC ($M = 4$) and its worst channel ADC. For each entry we have simulated 50 realizations of a time-interleaved ADC, where the channel ADCs have an INL drawn from a zero mean uniform INL

[1] dBc = decibels below carrier

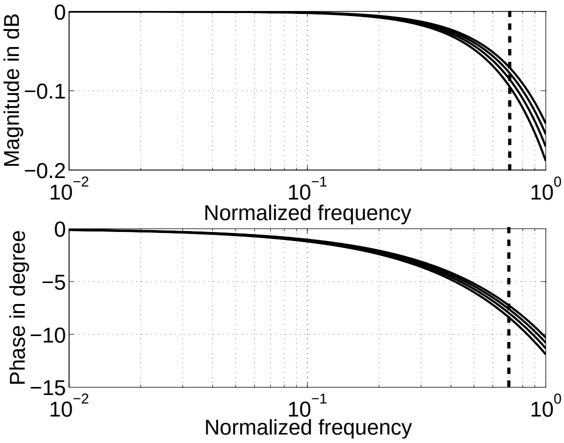

Fig. 6. Input filters used for demonstrating the bandwidth mismatch effect. The dashed line indicates half the sampling frequency.

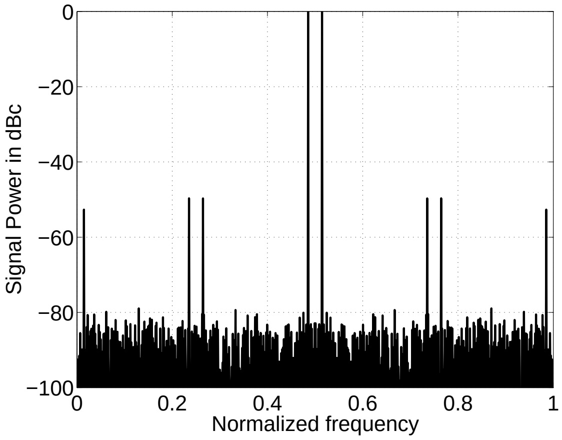

Fig. 7. The effect of bandwidth mismatches.

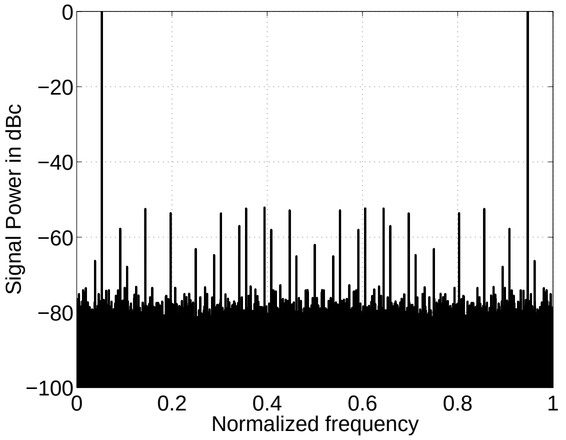

Fig. 8. Time-interleaved ADC where each channel ADC has a different INL.

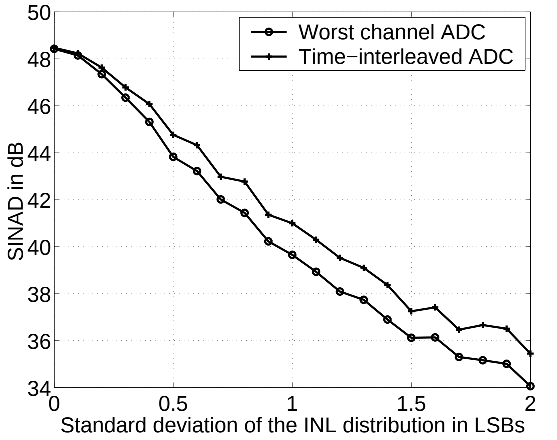

Fig. 9. Comparison of the SINAD of a time-interleaved ADC and its worst channel ADC.

distribution with a given standard deviation. We see that on the average the SINAD of the time-interleaved ADC is better than the SINAD of the worst channel ADC.

IV. Conclusion

We have introduced a MATLAB environment for simulating the behavior of time-interleaved ADCs. We show how to generate continuous-time input signals and filter them with continuous-time filters. We find a solution for the problem of sampling continuous-time signals in a non-uniform manner with high precision. Configurable parameters allow us to degrade the system performance to desired levels and characteristics. We are able to simulate the main signal processing errors in a time-interleaved ADC and can verify the results found in literature. Furthermore, we demonstrate the effect of bandwidth and INL mismatch effects. We show that a time-interleaved structure does not only decrease the INL but also improves the SINAD.

Acknowledgment

Support of our research by Infineon Technologies Development Centers Austria GmbH is gratefully acknowledged.

References

[1] William C. Black , Jr. and David A. Hodges, “Time-interleaved converter arrays,” *IEEE Journal of Solid State Circuits*, vol. 15, no. 6, pp. 1024–1029, December 1980.

[2] Christian Vogel, “Comprehensive error analysis of combined channel mismatch effects in time-interleaved ADCs,” in *Proceedings of the 20th IEEE Instrumentation and Measurement Technology Conference, 2003. IMTC 2003.*, May 2003, vol. 1, pp. 733–738.

[3] Yih-Chyun Jenq, “Digital spectra of nonuniformly sampled signals: Theories and applications-measuring clock/aperture jitter of an A/D system,” *IEEE Transactions on Instrumentation and Measurement*, vol. 39, no. 6, pp. 969–971, December 1990.

[4] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, 1993.

[5] “IEEE standard for terminology and test methods for analog-to-digital converters,” IEEE Std 1241-2000, June 2001.

[6] M. E. Tipping, “Sparse bayesian learning and the relevance vector machine,” *Journal of Machine Learning Research*, vol. 1, pp. 211–244, June 2001.

[7] Jose Basilio Simões, Jorge Landeck, and Carlos Manuel B. A. Correia, “Nonlinearity of a data-acquisition system with interleaving/multiplexing,” *IEEE Transactions on Instrumentation and Measurement*, vol. 46, no. 6, pp. 1274–1279, December 1997.