Software Based Sigma-Delta Converter with Auto-Calibration Capabilities

J.M Dias Pereira [1,2], ***Vítor Viegas*** [1,3], ***Octavian Postolache*** [1,4], ***and P.S. Girão*** [1,5]
[1] *Instituto de Telecomunicações, Lisboa, Portugal*
[2] *ESTSetúbal, Instituto Politécnico de Setúbal, Setúbal, Portugal*
[3] Centro de Investigação Naval (CINAV), Escola Naval, Base Naval de Lisboa – Alfeite, Almada, Portugal
[4] *ISCTE – Instituto Universitário de Lisboa, Lisboa, Portugal*
[5] *Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal*

Tel.: + 351.265.790000, fax: + 351.265.790000
E-mail: dias.pereira@estsetubal.ips.pt

Summary: This paper presents a software based sigma-delta converter. The proposed A/D converter represents a low-cost and flexible digitalization solution that can be used in a large number of applications. The proposed discrete A/D converter tasks, that are performed by a microcontroller, include: digital filtering, oversampling, noise shaping and decimation. A simple implementation of the digital filtering is easily implemented by counting the number of high and low pulses contained in the feedback bit stream generated by the microcontroller.

The first part of the paper contains an overview of the main characteristics of sigma-delta converters, the second part includes a brief description of the proposed sigma-delta converter, hardware, software and implementation details, and the last part will include simulation and experimental results.

Keywords: discrete A/D conversion, sigma-delta converter, digital filtering, oversampling, noise shaping, decimation, auto-calibration.

1. Introduction

This paper presents a new and simple method that can be used to perform measurements' errors compensation, namely: offset, gain and non-linearity errors compensation. Autocalibration is performed, in a digital way using a software based sigma-delta converter architecture [1] that includes a microcontroller device and a set of reference signals with amplitudes equal to 10%, 50% and 90% of the full-scale range of the A/D converter. The digital codes associated with each of the input reference signals are used to set the correction coeffients that compensate the errors previously referred. Besides the accuracy improvement that is achieved, self-testing capabilities of the A/D converter can also be implemented. This paper is organized as follows: section 2 includes a review of the main principles that are associated with the sigma-delta converter; section 3 presents system description; section 4 presents the software routine that was implemented to compensate offset, gain and first order non linearity errors; section 5 presents the simulink model that was used to perform simulation tests and includes some simulation and experimental results and the last section, section 6, draws conclusions.

2. Sigma-Delta Converters Working Principle: an overview

Sigma-delta A/D converters are specially known by their accuracy, resolution, low conversion bandwidth and digital implementation. The fast development of very large scale integration (VLSI) techniques, verified during last years, enabled an increasing selection of sigma-delta converters in different application domains.

For example, in smart sensing measurement systems [2] besides the advantages previously referred, it is possible to obtain a flexible A/D conversion solution [3], since the conversion rate, resolution and energy consumption can be dynamically adjusted according to the digitalization requirements of each measuring channel. These characteristics are easily implemented in hybrid sigma-delta converter solutions since the analog part of the converter enables a simple interface between conditioning circuits and converter input, and the software part enables a flexible and dynamic adjustment of converters' characteristics, namely, a variable conversion rate and resolution, and the implementation of oversampling, noise shaping, digital filtering and decimation [4], to name a few.

2.1 Oversampling

When the conveter's input signal is sampled at a much higher ratio than its bandwidth (Nyquist frequency), the converter's errors, namely quantization error, is spread over a large frequency range, from DC till one-half the sampling ratio. The oversampling ratio (OSR) of these converters are generally defined by,

$$OSR = \frac{f_{SR}}{f_{NY}} \quad (1)$$

where f_{SR} represents the signal's sampling rate and f_{NY} represents the input signal Nyquist frequency.

Using the relationship that gives the signal-to-noise ratio of an ADC (S/N=6.02·n+1.76 dB, where n represents the converter's number of bits) it is easy to conclude that each time the oversampling rate is increased by a factor of 4 the converter's ENOB (effective number of bits) is increased by one. Usually an external dither component, or more frequently the noise contained in the input signal (self-dithering), is used to smooth the noise amplitude over the oversampling frequency range (0-f_{SR}) and to avoid limit cycle oscillations [5].

2.2 Noise Shaping

This peculiar characteristic of any sigma-delta converter, derived from its feedback working principle, is associated with its capability to remove the conversion noise from signal bandwidth to higher frequencies. It is possible to deduce that the ENOB gain ($ENOB_G$), for a first order sigma-delta converter is given by [4],

$$ENOB_G = \frac{10}{6.02} \cdot \log_{10}\left[\frac{1}{2} \cdot \left(\frac{1}{OSR} - \frac{1}{2\pi}\sin\left(\frac{2\pi}{OSR}\right)\right)^{-1}\right] \quad (2)$$

2.3 Digital Filtering

Digital filtering basically removes the quantization noise from signal's bandwidth. It can be implemented using a finite impulse response filter or more frequently by using a "comb" filter [6] whose implementation does not require multiplications or usage of filter coefficients.

2.4 Decimation

Decimation reduces the bit stream rate to a lower value without loosing information as long as the Nyquist criterion is verified. In our implementation the decimation factor (M) is given by the inverse of the oversampling ratio,

$$M = \frac{1}{OSR} \quad (3)$$

Figure 1 represents, in a graphical way, the joint effect of the four digital signal processing techniques previously referred (oversampling, noise shaping, digital filtering and decimation).

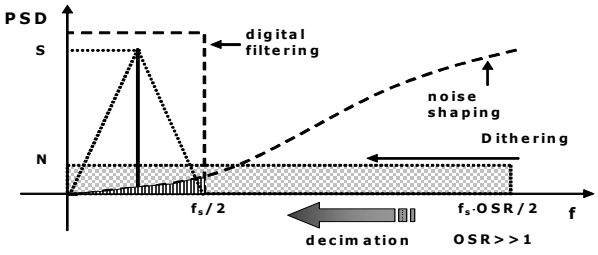

Fig. 1. PSD- power spectral density, S- signal PSD, N- noise PSD, OSR- oversampling factor.

3. System Description

Figure 2 represents the circuit block diagram used for experimental tests. The analog part of the modified sigma-delta A/D converter is implemented by an external integrator (O.A.) and a set of four passive components. For a fixed reference voltage (Vref) the ratio between resistors (R2/R1) can be used to set up the convert's full scale range. Since the integrator contains a single capacitor for charging and discharging phases, the conversion result is not affected by its accuracy. The analog comparator is implemented by the internal comparator of the microcontroller device (PIC10F204) [7] and the digital filtering and decimation is performed by a microcontroller's internal counter that is reset after each digitalization. The integrator circuit can also be initialized with the input voltage (Vin) before each conversion cycle in order to improve A/D conversion rate.

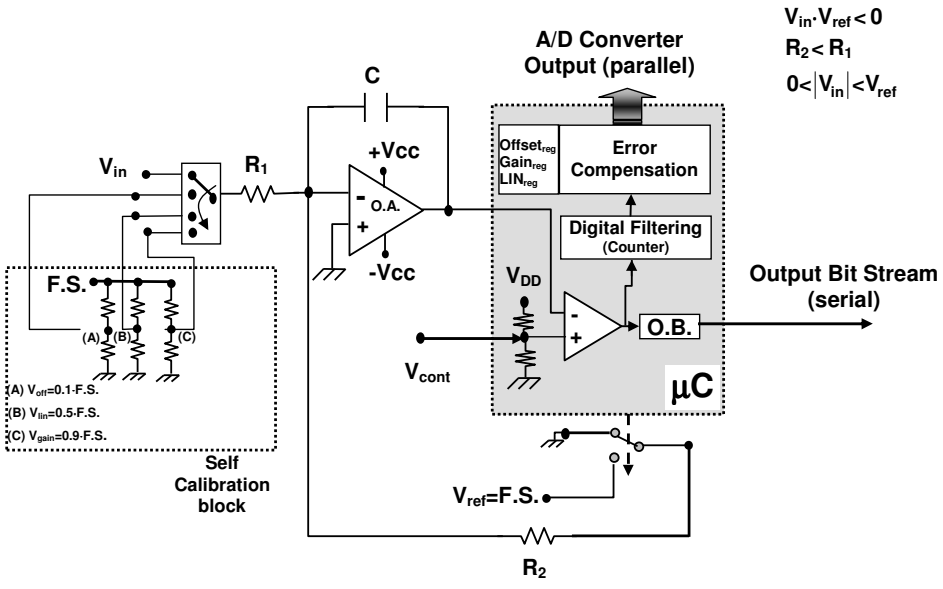

Fig. 2. Electrical circuit used for experimental tests (Vin- input voltage, Vref- reference voltage, µC- microcontroller PIC10F204, O.B.- output buffer, O.A.- operational amplifier).

4. Auto-calibration

A continuous auto-calibration procedure [8-11] can be implemented to compensate offset, gain and first order non-linearity error in the middle point of the A/D conversion range. Since it is assumed that the input voltage range of the voltages to be converted varies between 10% and 90% of the converter F.S., reference voltages of 0.1·F.S. and 0.9·F.S. are used to compensate offset and gain errors, respectively. Considering that N_{10}, N_{50} and N_{90}, are the output codes associated with 0.1·F.S. , 0.5·F.S. and 0.9·F.S. , respectively, the code for a normalized input voltage $x_n=V_{in}$/F.S., is given by:

$$N=N_{10}+K_1(x_n-0.1)+K_2(x_n-0.1)(x_n-0.9) \quad (4)$$

being:

$$\begin{cases} K_1=\frac{N_{90}-N_{10}}{0.8} \\ K_2=\frac{1}{0.16}\left(N_{10}+\frac{N_{90}-N_{10}}{2}-N_{50}\right) \end{cases} \quad (5)$$

From the previous relationships it is possible to obtain the input voltage with offset, gain and first order non-linearity errors' compensation that is associated with an output A/D converter code equal to N.

From (4)-(5), it is also possible to confirm that for negligible values of mid-range non-linearity error K_2 coefficient can be assumed almost null, since:

$$N_{50}\cong\frac{N_{10}+N_{90}}{2} \quad (6)$$

It is also obvious, but important to underline, that the value of the midrange code (N_{50}) can also be used to implement self-testing capabilities of the A/D converter.

As an example figure 3 represents the auto-calibration characteristic that is obtained for a 10 bit A/D converter (Nmax=1023) when N_{10}=128, N_{50}=400 and N_{90}=950.

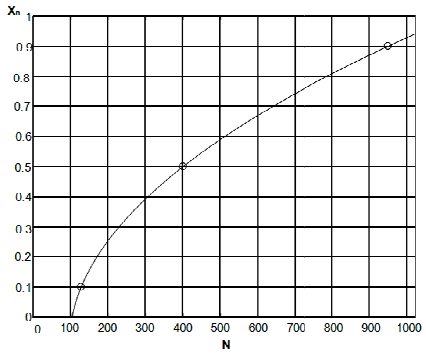

Fig. 3. Auto-calibration characteristic for a 10 bit converter with N_{10}=128, N_{50}=400 and N_{90}=950.

$$n_C=\frac{n-n_{OFF}}{n_{FS}-n_{OFF}}\cdot[1+K(n-n_{FS})]\cdot 2^n$$

$$K=\frac{1}{2}\cdot\frac{n_{FS}-2n_{FS/2}+n_{OFF}}{(n_{FS/2}-n_{OFF})(n_{FS/2}-n_{FS})}$$

5. Simulation and experimental results

Simulation tests where performed using the MATLAB-Simulink [12] program. The simulation model, represented in figure 4, includes the following main blocks: an input signal generator configured as a ramp generator; a stochastic dither generator to simulate self-dithering; a backlash block with zero-crossing detection; an amplifier with gain=100; a discrete time integrator; a relay block configured as a null comparator; a dicrete FIR filter configured as a moving average filter with 32 delay elements and a polynomial evaluation block to compensate offset, gain and first order non-linear errors. This polynomial block is configured as a second order polynomial that implements (4).

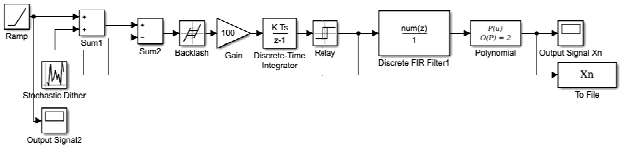

Fig. 4. MATLAB-Simulink block diagram used for simulation purposes.

Figures 5 and 6 represent the experimental results that were obtained with three different values of offset and gain errors, respectively. The set-up of the offset and gain errors were performed by changing the resistor values of the calibration switch in order to obtain the V(A) and V(C) voltages associated with the amplitude of each offset and gain error, respectively.

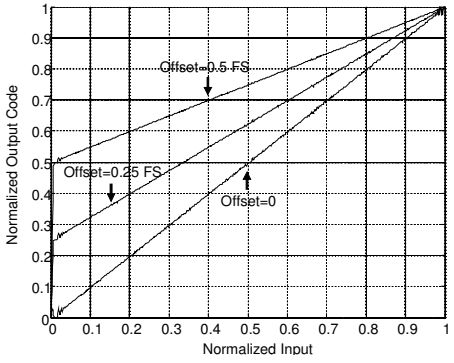

Fig. 5. Experimental results of the normalized transfer curves of the 10 bit A/D converter simulating offset errors with the following amplitudes: 0; 0.25·F.S. and 0.5·F.S..

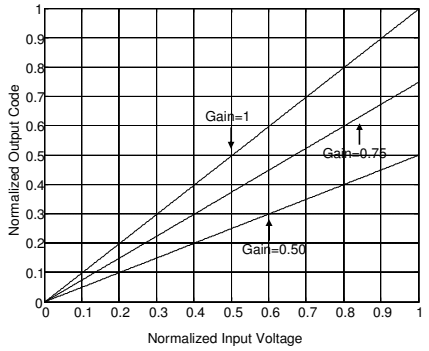

Fig. 6. Experimental results of the normalized transfer curves of the 10 bit A/D converter simulating F.S. gain errors with the following amplitudes: 0; -0.25·F.S. and -0.5·F.S..

The differences that were obtained between the normalized input voltages and the associated normalized output codes, before compensation, confirm the values of the offset and gain errors that were introduced with a deviation that is lower than 0.5 %. These deviations are within the deviation of the 1% accuracy resistors that were used to obtain the reference voltages in the calibration switch. This means that the proposed method compensates successfully the offset and gain errors that were purposely introduced

6. Conclusions

The present paper explained how microcontrollers can be used to implement some digital blocks of a software based sigma-delta A/D converter. The proposed A/D converter represents a low-cost and flexible digitalization solution that can be used in a large number of applications with smart sensing devices. The main disadvantages of the proposed converter are related with its maximum conversion rate, noise sensitivity and limitations caused by using a microcontroller for acquisition and data processing tasks. Several simulation and experimental results validate the theoretical expectation about the working principle of the proposed A/D converter.

Future work will focus on a complete metrological characterization of the A/D converter, whose main characteristics in terms of data conversion rate and resolution depends on microcontroller capabilities, and on the accuracy provided by the analog components, mainly the relative accuracy of the resistors that are used to generate the reference voltages used for auto-calibration purposes. Future implementations based onf Arduino and ESP [13-15], to take advantage of their increased gains in the manipulation of the resolution and frequency of PWM signals, will also be envisaged.

References

[1]. Rudy van de Plassche, "Integrated A/D and D/A Converters", Kluwer Academic Publishers, 1994.

[2]. Gert Van der Horn, Johan L. Huijsing, “Integrated Smart Sensors, Design and Calibration”, Kluwer Academic Publishers, 1998.

[3]. M. Pereira, O. Postolache, P. Girão, A. Cruz Serra, “A Discrete and Cost Efective ADC Solution based on a Pulse-Width Modulation Technique”, Proc. of the 3rd Conference on Telecommunications, Figueira de Foz, Portugal, April 2001, p.p. 153-156.

[4]. James C. Candy, Gabor C. Temes, “Oversampling Delta-Sigma Data Converters – Theory, Design and Simulation”, The Institute of Electrical and Electronics Engineers, Inc., New York – IEEE Press, 1992.

[5]. C.Dunn and M.Sandler, “Efficient Linearisation of Sigma-Delta Modulators using Single-Bit Dither”, Electronic Letters, Vol.31, No.12, pp.941-942, 1995.

[6]. C.Marven and G.Ewes, “A Simple Approach to Digital Signal Processing” John Wiley & Sons, Inc., 1996.

[7]. Microchip Technology, "PIC10F204 Data Sheet" (http://ww1.microchip.com/downloads/en/DeviceDoc/40001239F.pdf).

[8]. Sleiman Bou-Sleiman, Mohammed Ismail, "Built-in-Self-Test and Digital Self-Calibration for RF SoCs", Springer (SpringerBriefs in Electrical and Computer Engineering), 2012.

[9]. Gerard Meijer, Kofi Makinwa, Michiel Pertijs, "Smart Sensor Systems: Emerging Technologies and Applications", Wiley, 2014.

[10]. Oscar S. van de Ven, Johan G. Vogel, Sha Xia; Jo W. Spronck, Stoyan Nihtianov, "Self-aligning and self-calibrating capacitive sensor system for displacement measurement in inaccessible industrial environments", IEEE Transactions on Instrumentation and Measurement, Volume 67, Issue 2, Feb 2018, pp. 350-358.

[11]. J.G. Vogel, Stoyan Nihtianov , "Auto-alignment of a High-Precision Eddy-current Displacement Sensor Using a Thermal Slider Actuator", In Proceedings of the 18th international conference of the EUSPEN, 2018, pp. 171-2.

[12]. MathWorks, "Simulation and Model-Based Design" (https://www.mathworks.com /products/simulink.html).

[13]. B. N. Karthik, L. Durga Parameswari, R. Harshini, A. Akshaya, "Survey on IOT & Arduino Based Patient Health Monitoring System", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Volume 3, Issue 1, 2018, pp. 1414-1417.

[14]. Antonio Carlos Bento , "IoT: NodeMCU 12e X Arduino Uno, Results of an experimental and comparative survey", International Journal of Advance Research in Computer Science and Management Studie, Volume 6, Issue 1, 2018, pp. 47-56.

[15]. Espressif Systems, ESP8266EX Datasheet (https://espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf).