IMPROVING ANALOG-TO-DIGITAL CONVERTER'S RESOLUTION USING THE OVERSAMPLING TECHNIQUE

Flaviu Ilie BOB, Nicolae Cristian PAMPU, Liviu Teodor CHIRA

Faculty of Electronics, Telecommunications and Information Technology, Technical University of Cluj-Napoca
Ilie.BOB@bel.utcluj.ro, Nicolae.PAMPU@ael.utcluj.ro, Liviu.CHIRA@bel.utcluj.ro

ABSTRACT

This paper is going to expose a method that gives us the possibility to use a low-resolution Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC) in high-resolution measurements. We increase the resolution of a 10-bits ADC to 16-bits by superposing a pseudo noise over the signal to be measured. This method also rejects the noise of electrical networks for the 50 Hz and 60 Hz frequencies. It can be only used for measuring very low frequency or continuous signals, but the costs are lower compared to the price of the same high-resolution converter.

Keywords: *oversampling, analog-to-digital converter.*

1. INTRODUCTION

SAR converters are used in PC plug-in data-acquisition boards or PC external data acquisition systems. They are by far the most popular ADCs in today measurement products [1]. One reason for their popularity is their outstanding linearity which comes from the fact that they usually exploit a 1-bit quantizer. Even with a trilevel quantizer, linearity performance up to 20 bits has been reported [2].

The oversampling is a method to improve the resolution of a converter, by software methods, with a little help from outside. It is useful when we have a microcontroller with SAR ADC and we want to measure output signals from a sensor with good resolution.

The principle of oversampling is to take a great number of conversions, and calculate a mean value. It is useless to take lots of equal measurements and to calculate the mean value of them, because you will obtain the same result: the value. The whole ideea is based on the presence of a white noise without a continuous component.

2. METHODS OVERVIEW

2.1 Previous work

A lot of documentation is available for this algorithm in the literature, but it describes the functionality for Sigma Delta Converters. An oversampling method for SAR ADCs is presented in [3] where we can find the most important aspects of this method. A method with noise is presented in [4], where the authors present the importance of the noise to obtain good and accurate results.

2.2 SAR ADC

The SAR ADC is a type of analog-to-digital converter that transforms a continuous waveform into a discrete digital representation via a binary search through all possible quantization levels before finally converging upon a digital output for each conversion (Figure 1).

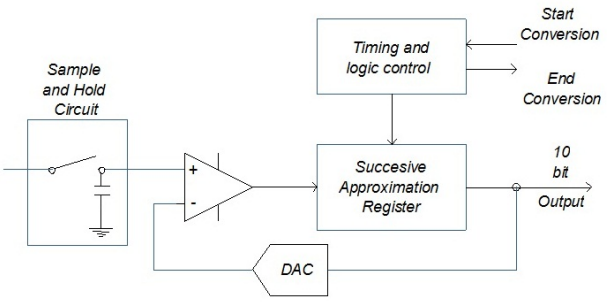

Figure 1*: Internal diagram of the SAR ADC.*

The successive approximation analog to digital converter circuit typically consists of four main sub circuits:

- A sample and hold circuit to acquire the input voltage (V_{in}).
- An analog voltage comparator that compares V_{in} with the output of the internal digital-to-analog converter (DAC) and outputs the result of the comparison to the successive approximation register (SAR).
- A SAR sub-circuit designed to supply an approximate digital code of V_{in} to the internal DAC.
- An internal reference DAC that supplies the comparator with an analog voltage equivalent to the digital code output of the SAR for comparison with V_{in}.

2.3 Oversampling principle

The main microcontroller producers sustain a theory based on the number of oversampling bits that you want to obtain. A formula is calculated that involves taking a number of measurements equal to a power of 4, and dividing the whole sum to a power of 2. The power is equal to the number of oversampling bits that you want to obtain [3, 4].

$$Result = \frac{\sum_{i=1}^{4^N} ADC_i}{2^N}. \quad (1)$$

Where:

- **ADC** - One conversion of maximum 2^x bits;
- **Result** - Final oversampled result of maximum 2^{x+N} bits
  - **x** - Converter's implicit number of bits
  - **N** - Number of oversampling bits

2.4 Issues

One issue is for the devices that are powered from the main power line. This creates a noise that has the same frequency as the power line (50/60 Hz). This may not be noticeable in the normal measurements, but a measurement with oversampling will be seriously affected by this noise. This noise is most likely to be eliminated by taking measurements in a time window that is multiple of the power line noise period.

The most suitable time window is equal to a multiple of 100 milliseconds. This eliminates both the 50 and the 60 Hz noise. It is also necessary to evenly distribute the measurements in this time window, so the distance in time between all measurements is equal.

The circuit that implies this measurement method must be very carefully designed, because the oversampling requires a lot of accuracy and it is very sensitive.

Another issue is the sample rate which is dramatically decreased once the oversampling is used, due to the 100ms time for 1 measurement. Thus it is most likely used to measure DC signals.

2.5 Noise

In [3] and [4] is described the necessity of the noise. To make this method work properly, the signal component of interest should not vary during a conversion. However another criterion for a successful enhancement of the resolution is that the input signal has to vary when sampled. This may look like a contradiction, but in this case variation means just a few least significant bits (LSBs). The variation should be seen as the noise-component of the signal.

When oversampling a signal, there should be noise present to satisfy this demand of small variations in the signal. The quantization error of the ADC is at least 0.5 LSB. Therefore, the noise amplitude has to exceed 0.5 LSB to toggle the LSB. Noise amplitude of 1 or 2 LSB is even better because this will ensure that several samples do not end up getting the same value.

Figure 2 shows the problem of measuring a signal with a voltage value that is between two quantization steps. Averaging four samples would not help, since the same low value would be the result. It may only help to attenuate signal fluctuation. Figure 3 shows that by adding some artificial noise to the input signal, the LSB of the conversion result will toggle.

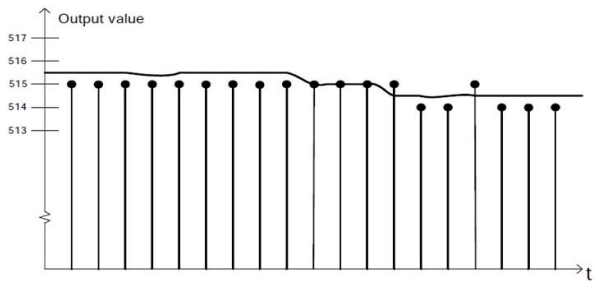

Figure 2*: Initial conditions of measuring [4].*

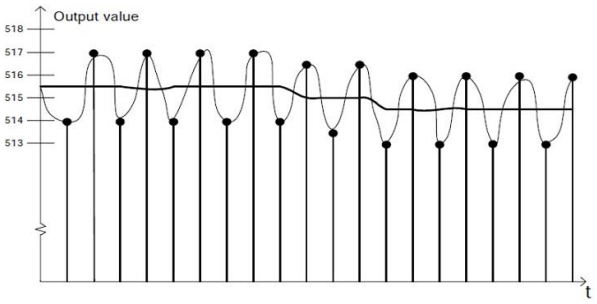

Figure 3*: Measuring with noise [4].*

3. IMPLEMENTATION

3.1 Hardware Implementation

The main hardware configuration for the oversampling method that was tested is shown in Figure 4. It is composed of a noise generator block, a microcontroller with 10-bits SAR ADC, a stable reference V_{ref} and a signal witch is measured as V_{in}. A rectangular clock signal is applied on the noise generator circuit, which produces a noise that is applied over the reference voltage V_{ref}.

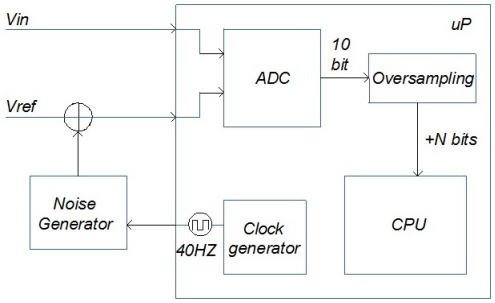

Figure 4*: Block diagram for hardware implementation.*

The noise generator must generate a signal that achieves a zero value before the next half-alternation occurs. The time-constant must be calculated correctly, depending on the number of alternations. We used eight alternations for the clock signal on a 100 ms period.

3.2 Software

The microcontroller's software controls the converter and the adjacent circuits and processes the read information. It also generates the noise signal. This can be done continuously, but it must contain an even integer number of noise alternations in the time-window in which the measurements are taken. There are available two alternatives: either we create the noise generator signal continuously through a PWM generation module of the microcontroller, and synchronize the measurement time-window of 100 ms with the noise generator signal; either we create the noise generator signal by software methods, by setting and resetting a port. Doing this we do not have the problem of synchronizing the measurements time-window. Being necessary for the measurements to be equally distributed in the time-window, there is an aspect that must be had in sight: the fact that when a port is toggled, there is done another operation between two con-

versions, and it could increase the time between them. Thus this operation must be simulated even when it is not done, by using a no operation (NOP) instruction. In the 100 ms time-window the unneeded interrupts are disabled.

The time-window can be obtained by doing a fixed number of conversions, or by the means of a timer interrupt. The converter must work using the noised reference, and it has to work as fast as possible, keeping the controller in ”idle” or ”sleep” while the conversions are done, so that there will be no noise generated by the processor core. The minimum number of required conversions for the oversampling must be respected. The acquisition time needed by the converter must be respected too, and it is recommended that before the 100 ms time-window to be done some pre-conversions (dummy conversions) without noise, so that the internal acquisition level of the converter can reach a value as close as possible to the value to be measured.

3.3 Noise

For this experiment we used a noise signal that has an even number of alternations, so that the continuous component is zero. This is important because we tried not to affect the continuous component of the signal to be measured. The time diagram of the noise is presented in Figure 5.

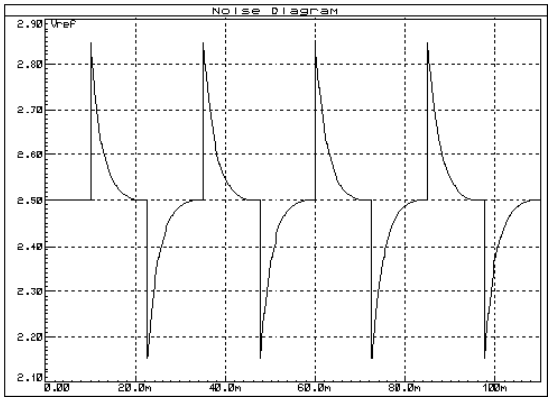

Figure 5*: Noise time diagram over 110ms.*

The noise signal was obtained using a simple RC circuit and a signal command from the controller. The maximum peaks of the noise are 350 mV. This value was obtained via empirical methods. The equations are presented bellow:

$$v_C(t) = \begin{cases} E \cdot e^{-\frac{t}{RC}} & , t < \frac{T}{2} \\ -E \cdot e^{-\frac{t}{RC}} & , t \geq \frac{T}{2} \end{cases} \quad (2)$$

$$v_R(t) = -v_C(t) \quad (3)$$

In (2) and (3) the variables are:

- E - source voltage;
- R , C - resistor, capacitance values;
- t - time variable;
- T - time period of noise;
- v_R , v_C - voltage over the resistor, capacitance;

The noise beeing driven by the controller, it is synchronized with the conversions - the sampling clock - by software methods.

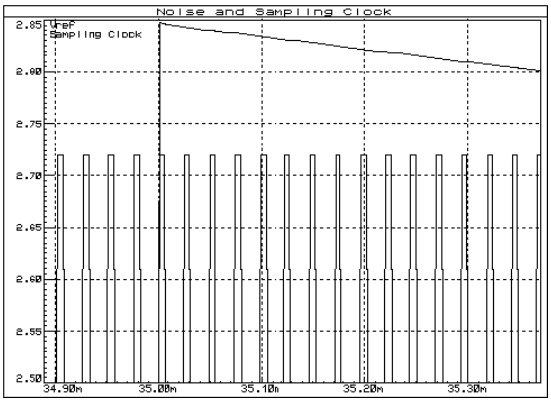

Figure 6*: Zoom on the sampling clock over 500us.*

4. RESULTS AND CONCLUSIONS

For the experiments we used a Microchip PIC16F1936 microcontroller with an integrated 10-bits ADC. The reference voltage is supplied from a dedicated voltage reference circuit LM385-25 of 2.5 volts, passed through an IC27L2 operational amplifier, for adding the noise signal.

The errors of the devices that were used are also to be taken into consideration. The RIGOL DG1012 signal generator used for generating the input voltage has a drift in time of approximately 30 micro volts. The multimeter MASTECH MY-65 used as a reference for measurements has a resolution of 10 micro volts and an accuracy of 0.05% on the 200 mill volts scale that was used. This equals with an accuracy of 200 micro volts at 200 mill volts.

The linearity of the converter will be corrected by means of a 2 point software calibration. For data acquisition we receive data from the Recommended Standard 232 (RS-232) port of the microcontroller.

The stability of the converter's results for different input voltages is presented in Figures 7 and 8. It is visible that all errors are smaller than 1 LSB, which is 38.14 microvolt (4).

$$1LSB = \frac{2500mV}{2^{16}} = 38.14697uV. \quad (4)$$

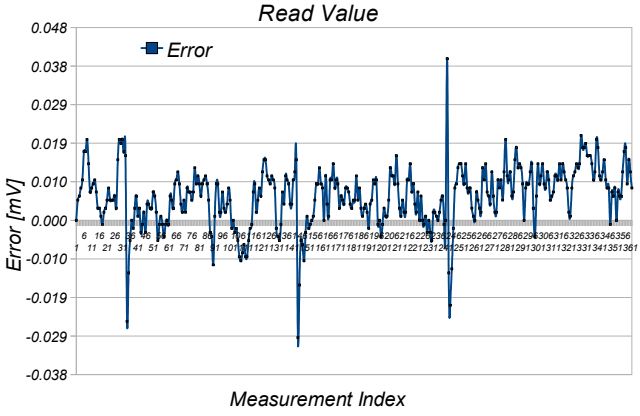

Figure 7*: Error measurements for 100.00mV.*

In Figure 8 is shown the stability of the measurements

for 123.98 mV. For this test there were acquired almost 2000 consecutive measurements. There are to be seen some spikes, caused by the exterior noise sources such as commutations on the power line.

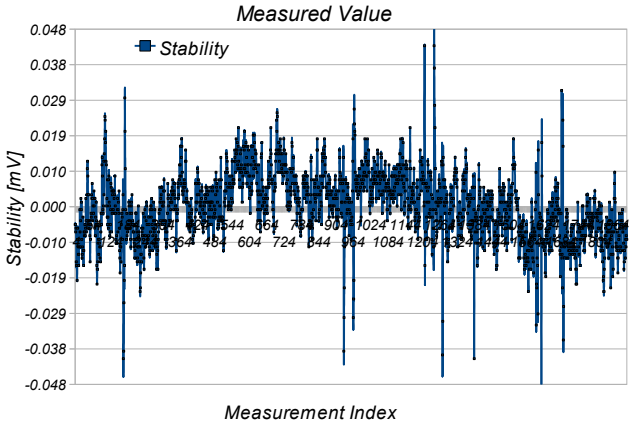

Figure 8*: Error measurements for 123.98mV on 1913 consecutive measurements.*

In Figure 9 is shown the linearity of the measurements for 2.44 millivolts (1 of 10 standard bits of the ADC). For this test we modified the input voltage with in steps of 38.14 uV (1 LSB of 16 oversampled bits of the ADC) and we read the measurement values from the RS-232 port.

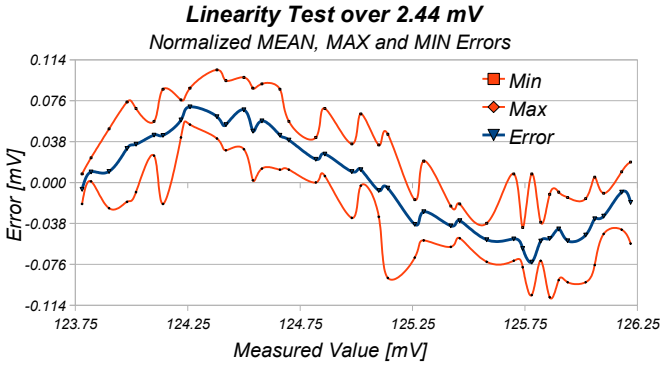

Figure 9*: Linearity test over 2.44 mV.*

In the next two figures is highlighted the difference between the noise and no noise measurements. Figure 10 represents the same result that is shown in Figure 9, but at a different scale, so that the difference can be seen.

Figure 11 is the result of the same tests done without the pseudo-noise. It is the result of an incorrect usage of the oversampling technique: without noise. As it can be seen, the results are disastrous. This makes it clear that our technique substantially improves the converter's performance.

The proposed measure method can be used only for very low frequency signals or continuous signals, like temperature sensors. The bandwidth is 1 Hz and the electrical network noise is eliminated.

To conclude, as visible in Figures 7 and 8, we get 16 bits ± 1 LSB resolution stability in every point of range and the accuracy is maintained within 2 LSB.

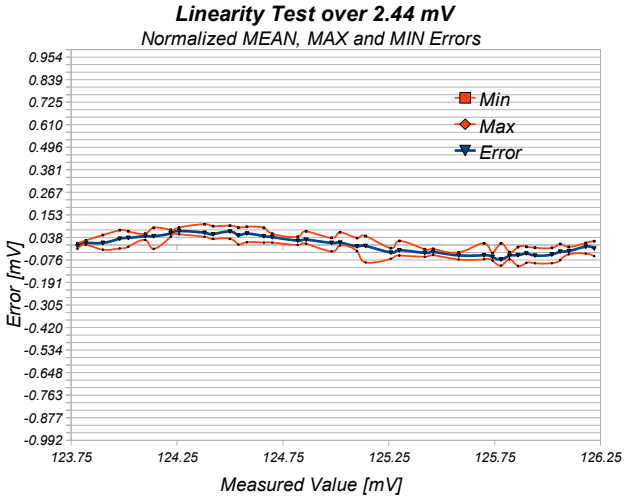

Figure 10*: Linearity test with noise.*

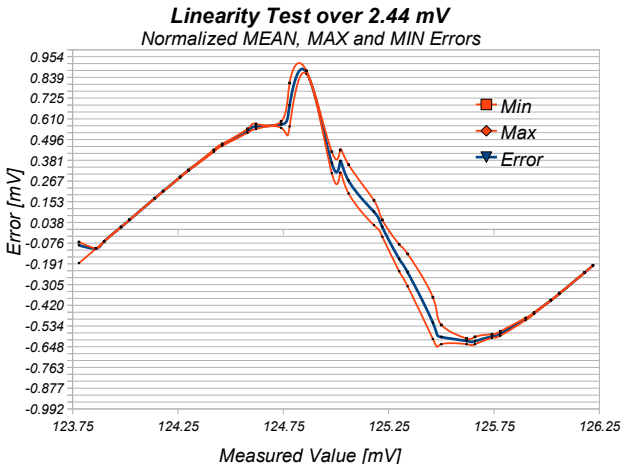

Figure 11*: Linearity test without noise.*

REFERENCES

[1] J.C. Candy and G.C. Temes, *Oversampling Methods for A/D and D/A Conversion*, pp. 1-29, IEEE Press, 1992.

[2] D.A. Kerth, *A 120dB linear switched-capacitor Delta-Sigma modulator*, pp. 196-197 in ISSCC, Feb. 1994,.

[3] Jayanath Murthy Madapura, *Achieveing Higher ADC Resolution using Oversampling*, Microchip Technology Inc., 2008.

[4] *AVR121: Enhancing ADC resolution by oversampling*, ATMEL 8-bit AVR Microcontrollers, Application Note.

Acknowledgment

This paper was supported by the project "Improvement of the doctoral studies quality in engineering science for development of the knowledge based society-QDOC" contract no. POSDRU/107/1.5/S/78534, project co-funded by the European Social Fund through the Sectorial Operational Program Human Resources 2007-2013.