

User-friendly Matlab tool for easy ADC testing

Tamás Virosztek, István Kollár

*Budapest University of Technology and Economics, Department of Measurement and Information Systems
Budapest, Hungary, H-1521, email: Virosztek.Tamas@mit.bme.hu, kollar@mit.bme.hu*

Abstract-This paper describes a freely available, open source, user-friendly data evaluation program developed for ADC testing with a sine wave. This tool performs multiple kinds of mathematical methods (nonlinear least squares and maximum likelihood) to extract the information from the recorded data. It has been created in the form of a Matlab toolbox for users who have some knowledge in ADC testing, and would like to perform the mathematical computations in an efficient way – especially those that require complex numerical methods. The software is a significantly extended version of the earlier similar-purpose toolbox. As it is equipped with a graphical user interface, its usage does not require any programming knowledge, nevertheless one can even get familiar with the framework of the software, as the program files contain the source code that runs using the Matlab interpreter. A further important goal of this software is to create the possibility of publishing reproducible ADC test evaluation results. Statements of related papers can be verified easily: if the raw data are shared, the computations can be repeatedly executed using this software tool.

Keywords-ADC testing, maximum likelihood, least squares, histogram test, standard IEEE-1241

I. Introduction

The aim of this software is to handle and evaluate measurement results for ADC testing with sine wave excitation. The idea is that the user

- creates an experiment descriptor from the data samples and the circumstances of the measurement,
- imports this descriptor into the graphical user interface,
- performs the selected analysis method simply by pushing buttons.

In the description below, first the available analysis methods are described. Then, important aspects of the GUI are discussed, and the structure of the descriptor is described with a suggested way to create it.

II. Possibilities to process measured data

Data recorded for ADC testing can be processed in several different ways. This toolbox offers the following options to evaluate the measurement.

A. Three-/four-parameter sine wave fit in least squares sense as described in standard IEEE 1241-2010 [2]

This method fits a sine wave to the recorded data with respect to three or four parameters: amplitude, phase, DC component, and in the case of a four-parameter fit, the frequency. Technically the sine wave is decomposed to a cosine and a sine wave with zero initial phases: the coefficients of the cosine and the sine wave provide the same information as amplitude and initial phase. The cost function is the sum of squares of residuals calculated using the samples of the measured sine wave and the samples of the fitted sine wave:

$$CF_{LS} = \sum_{n=0}^{N-1} (y(n) - A \cdot \cos(2\pi f t_n) - B \cdot \sin(2\pi f t_n) - C)^2 \quad (1)$$

In this formula $y(n)$ denotes the output of the ADC in the time moment t_n , A is the coefficient of the cosine component, B is the coefficient of the sine component, C is the DC component of the signal, and f is the frequency of the sine wave. This way the three-parameter fit is linear in all parameters: it can be performed using a simple matrix operation. The four-parameter sine wave fit is somewhat more complicated: the frequency is also a parameter to be estimated, thus iterative methods are required. The algorithm described in Annex B of [2] is implemented in the toolbox: this is the default numerical method to perform a four-parameter LS fit. The frequency estimator is corrected in each step: the other parameters are estimated solving a linear least squares

¹ This work has been supported by grant TÁMOP - 4.2.2.B-10/1--2010-0009

problem using the corrected frequency estimator. The initial frequency estimator is very important concerning the stability of the algorithm: inaccurate initial frequency can totally mislead the optimization. Therefore, f_0 is calculated using the interpolated FFT of the measurement record.

The toolbox also provides another method to fit the sine wave in least squares (LS) sense using the general MATLAB function *lsqnonlin* (*Optimization Toolbox*) [7]. Computation time is larger in this case, but this robust algorithm can be useful when the above problem-specific fast method would be carried away from the optimum.

Before performing the LS fit, the recording can be edited: a range in time domain can be selected to discard irrelevant parts of the record (e. g. when some disturbances are observed in the measurement data). Also the effect of overdrive can be eliminated: samples containing the ADC codes near the edge of the full scale can be ignored. The default setting is to discard the samples containing the highest and lowest codes of the ADC under test.

The residuals (the difference of the recorded data and the fitted sine wave) are used to calculate the quantities that describe the device under test. RMS value of the residuals can be compared to the quantization noise of an ideal quantizer: by this way the effective number of bits (ENOB) can be calculated. The power of the base harmonic of signal can be compared to the power of noise and distortion components: this way signal to noise and distortion ratio (SINAD) can be calculated.

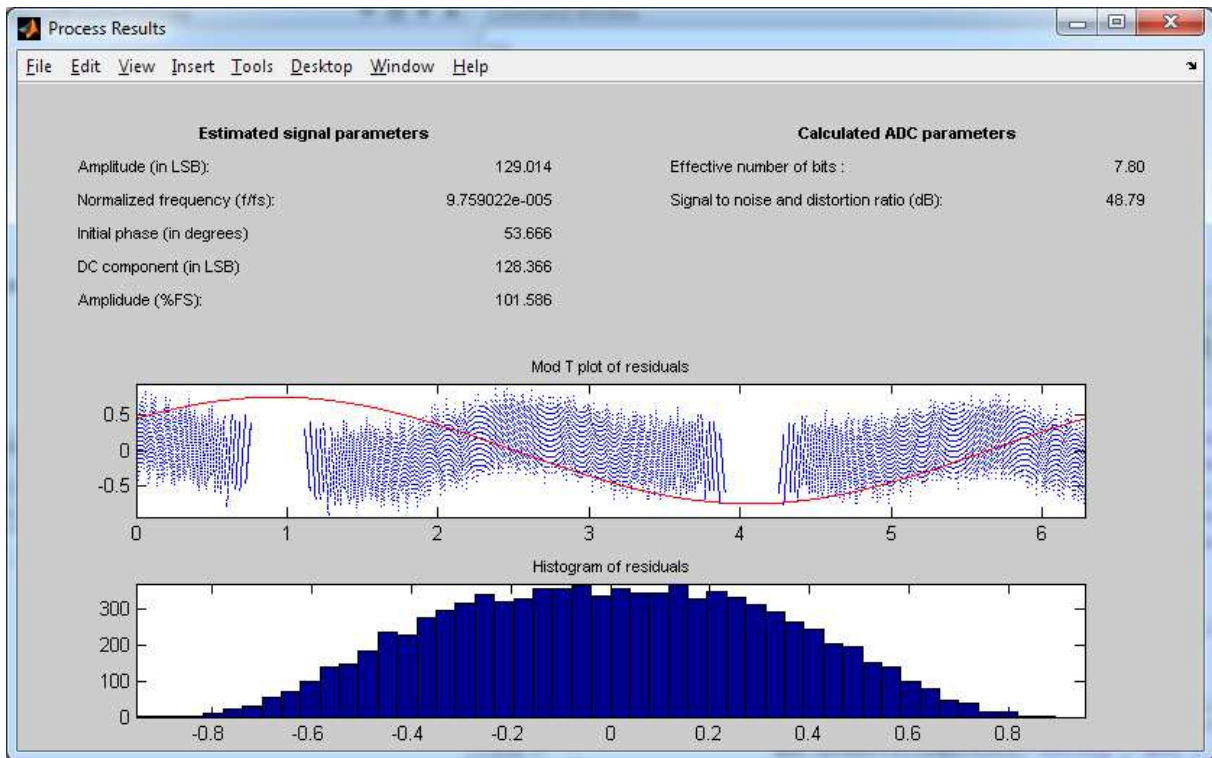


Fig. 1 Resulting window of the four-parameter LS fit

B. Maximum likelihood estimation of signal and ADC parameters

The aim of maximum likelihood (ML) estimation is to extract the maximum possible information from the recorded data. This method finds the ML estimators of the signal parameters maximizing the likelihood function for ADC testing:

$$L(\mathbf{p}) = \prod_{n=0}^{N-1} P[y(n) = Y(n, \mathbf{p})] \quad (2)$$

where $y(n)$ denotes the n^{th} sample of the measurement record, and $Y(n, \mathbf{p})$ is a discrete random variable. For each n , $Y(n, \mathbf{p})$ can be an integer between 0 and $2^b - 1$, depending on the signal parameters (amplitude, phase, frequency, DC component, deviation of additional noise), and the transition levels of the ADC under test. Thus the parameter vector is: $\mathbf{p}^T = [A, B, C, f, \sigma, T(1), T(2), \dots, T(2^b - 1)]$. To solve the estimation problem, the likelihood

function shall be maximized with respect to the parameters in \mathbf{p} . To ease computations, it is useful to define and minimize the negative log-likelihood cost function (CF_{ML}):

$$CF_{ML}(\mathbf{p}) = -\ln(L(\mathbf{p})) = -\sum_{n=0}^{N-1} \ln(P[y(n) = Y(n, \mathbf{p})]) \quad (3)$$

This cost function can be minimized using general optimizing methods. The theoretical fundamentals of ML parameter estimation for ADC testing are described e.g. in [4].

The algorithm implemented in the toolbox is an approximate solution of the ML problem. In the likelihood function (3) the transition levels are parameters to be estimated. The excitation signal is described by five parameters: the four parameters from LS fit and the standard deviation of the additive noise on the analog excitation signal. Thus the dimension of the parameter space is $2^b + 4$.

Optimization of the likelihood function with respect to the transition levels is problematic for two reasons. On the one hand the large number of parameters increase the computing time to intolerable values. On the other hand, the sensitivity of the likelihood function to a single transition level is low: the estimators of the transition levels cannot be optimized effectively. If the likelihood function is smooth enough with respect to a transition level, the numerical method can terminate with inaccurate results, depending on the other parameters of optimization.

An approximate solution is to extract the transition level estimators from the histogram test and to maximize the likelihood function with respect to the remaining five signal parameters. This can be performed using an iterative algorithm. Initial signal parameter estimators are taken from LS four-parameter fit and the initial noise deviation estimator is calculated using the noiseless LS fitted signal and the recorded data. The numerical method implemented for iterative minimization is the Levenberg-Maquardt method. Through the GUI, the progress of iteration can be observed: the actual values of the most important quantities appear on the screen at each step. The iteration can be paused, resumed and stopped: the termination criteria can be adjusted, or the algorithm can be stopped if the criteria were chosen improperly (see Fig. 2).

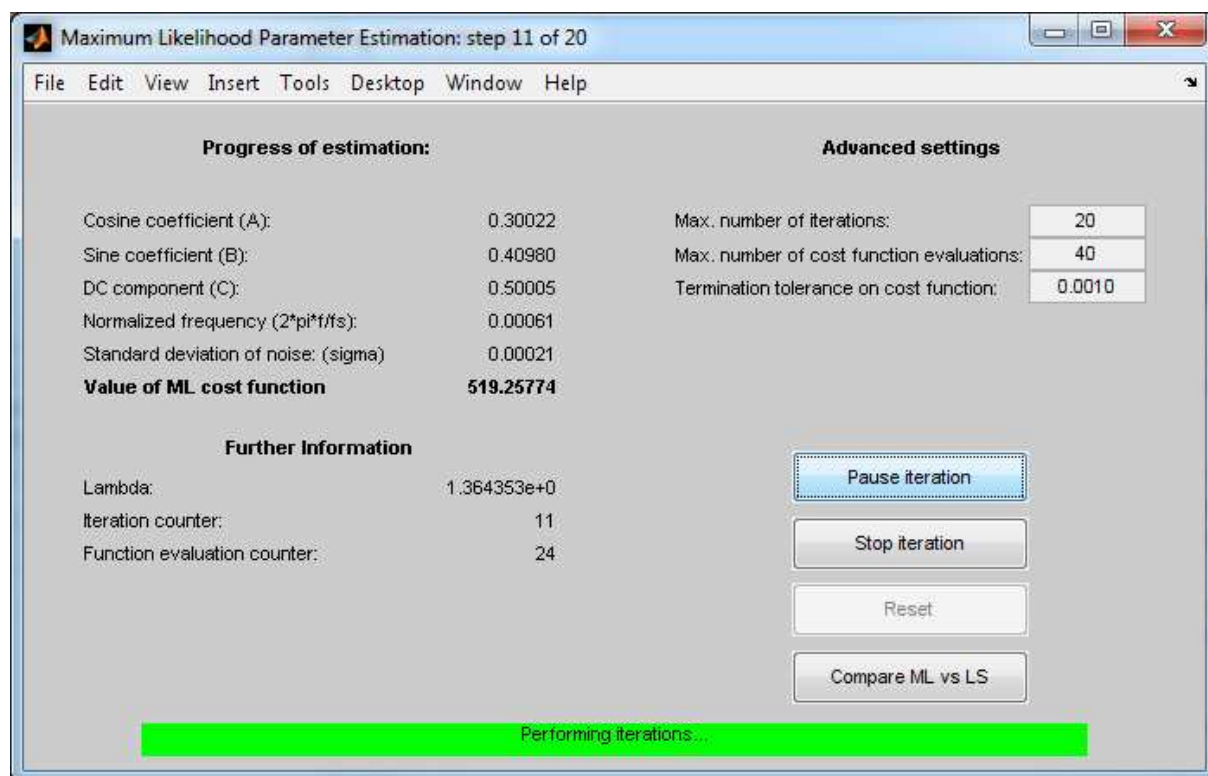


Fig. 2 Minimization of the ML cost function: performing iterations

Maybe the most important new feature of the toolbox is the comparison of ML and LS estimation: the signal and ADC parameters appear beside each other in the comparison window: results for the same data sets using different methods can be compared directly. In this window the residuals can be observed in the phase domain

(in the so-called mod T plot) and in the amplitude domain (in the histogram of residuals). Fig. 3 shows an example for this comparison.

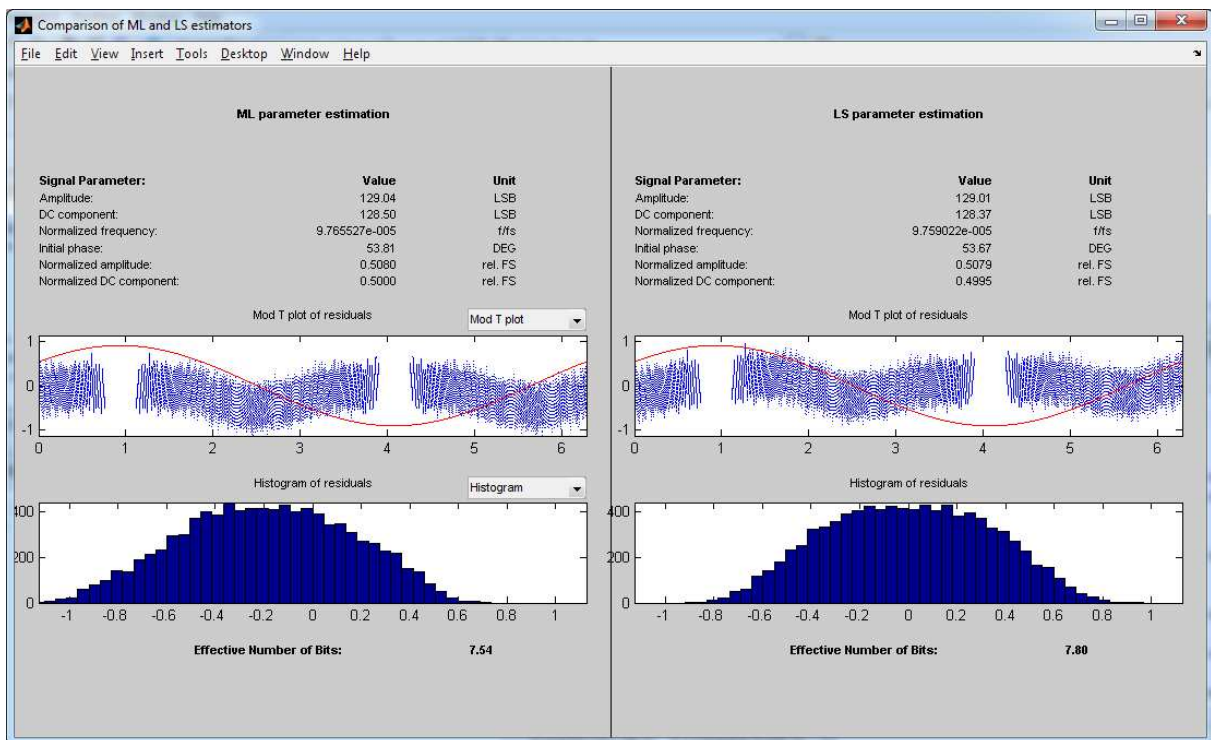


Fig. 3 Comparison of ML and LS parameter estimation

The result in Fig. 3 is quite typical: for nonlinear ADC characteristics: the effective number of bits is smaller when calculated using the maximum likelihood method than with LS. Thus LS returns a more attractive number ... but this is not correct. ML which better fits the sine wave returns the proper ENOB (here smaller by 0.26 bit).

C. Estimation of the static transfer characteristic using histogram test

The transition levels of the ADC can be directly estimated creating a histogram from the measured data. The recorded histogram can be compared to the probability density function (PDF) of the sine wave. The nonlinearities can be observed through the irregularities of the histogram: wider code bins are overrepresented, narrow code bins are underrepresented in the histogram. The fundamentals of histogram testing are very precisely described in [3]. The most important practical problem is the required number of samples: to keep the variance of INL and DNL estimators low, several samples are required in each code bin. As the PDF of the sine wave is high near the extrema of the signal, and low otherwise, the codes near the DC component of the excitation signal appear rarely. To achieve a few samples also in these code bins, the number of samples in a record must be high ([3] also describes the required number of samples in a record to achieve a desired accuracy of INL and DNL estimators).

This fact raises a requirement concerning the stability of the excitation signal source. To histogram test a high resolution (e.g. 20-bit) ADC, several million samples must be acquired, even at low accuracy (e. g. 10%) and low confidence level (e. g. 90%). Considering the sampling rate of the high resolution ADCs, recording of several million samples can take minutes: the analog signal source must keep signal parameters (especially frequency) stable for minutes.

Fig. 4 shows the result of a histogram test: a 14-bit ADC was simulated with the static transfer characteristic of a real ADC. The simulated measurement record contains 1 million samples. In case of shorter measurement records, the estimated INL and DNL values would be imprecise.

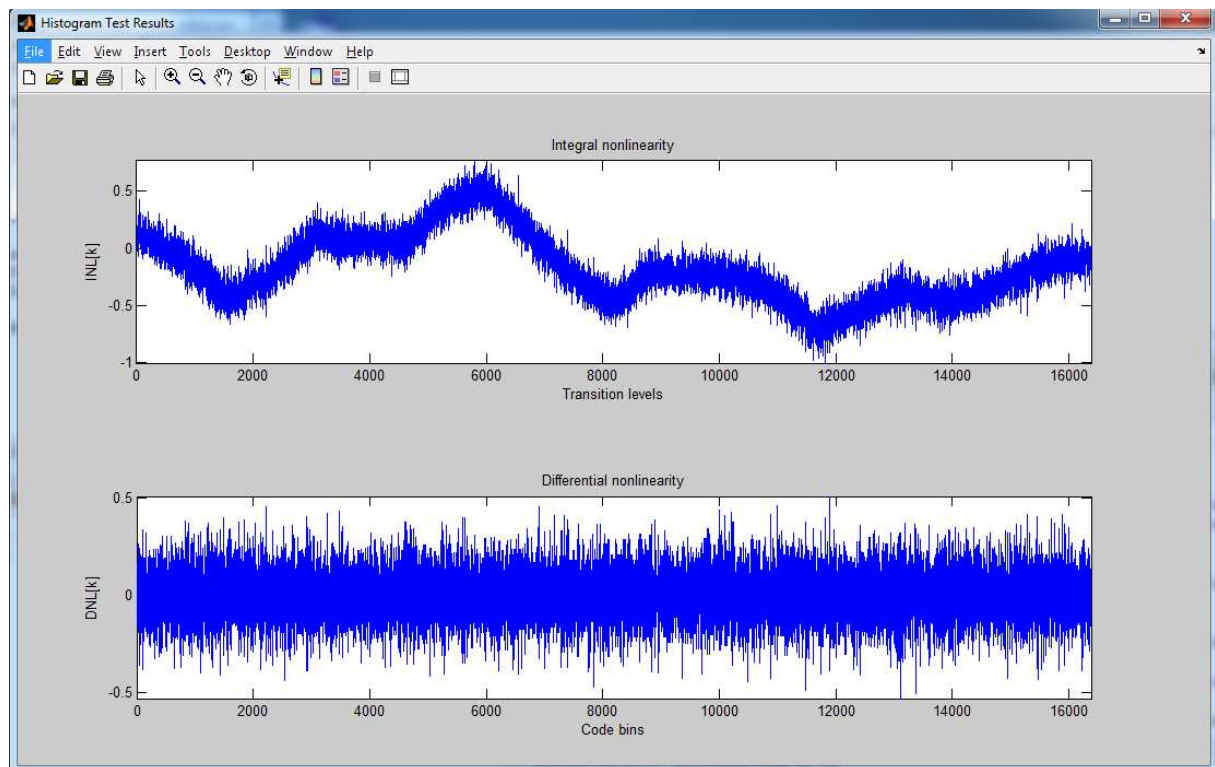


Fig. 4 Result of histogram testing

As explained above, the maximum likelihood method for sine wave fitting uses the transition levels extracted from the available samples, and the number of these samples is usually low for proper histogram testing. However, these inaccurate transition levels represent the full information that can be extracted from the data for sine fit, and acceptably describe the nonlinear characteristics. The best thing one can do is to use these more or less inaccurate estimators of the transition levels in the further maximum likelihood estimation of the sine wave.

D. DFT/FFT test

Frequency domain analysis can also be performed using this toolbox: processing the FFT of the measurement record, important ADC parameters such as harmonic distortion, and spurious-free dynamic range (SFDR) can be observed. One can choose from three window functions in the time domain: Hann, flat top, and Blackman windows are available (and, certainly, FFT can be performed also without windowing: rectangular window). In the GUI the amplitudes of the 2. up to 5. harmonics appear (relative to the amplitude of the base harmonic). Also the spectrum is displayed from DC to the Nyquist cutoff frequency. The SFDR, the amplitude of the highest harmonic or spurious spectral component relative to the base harmonic is shown in the first line of results. Fig. 5 shows a result window of the FFT test. This example is a simulated measurement to illustrate spectral properties of a sine wave quantized with a slightly nonlinear ($INL < 1$ LSB) 8 bit ADC. The static transfer characteristic was designed to generate odd harmonics: it is visible in the figure.

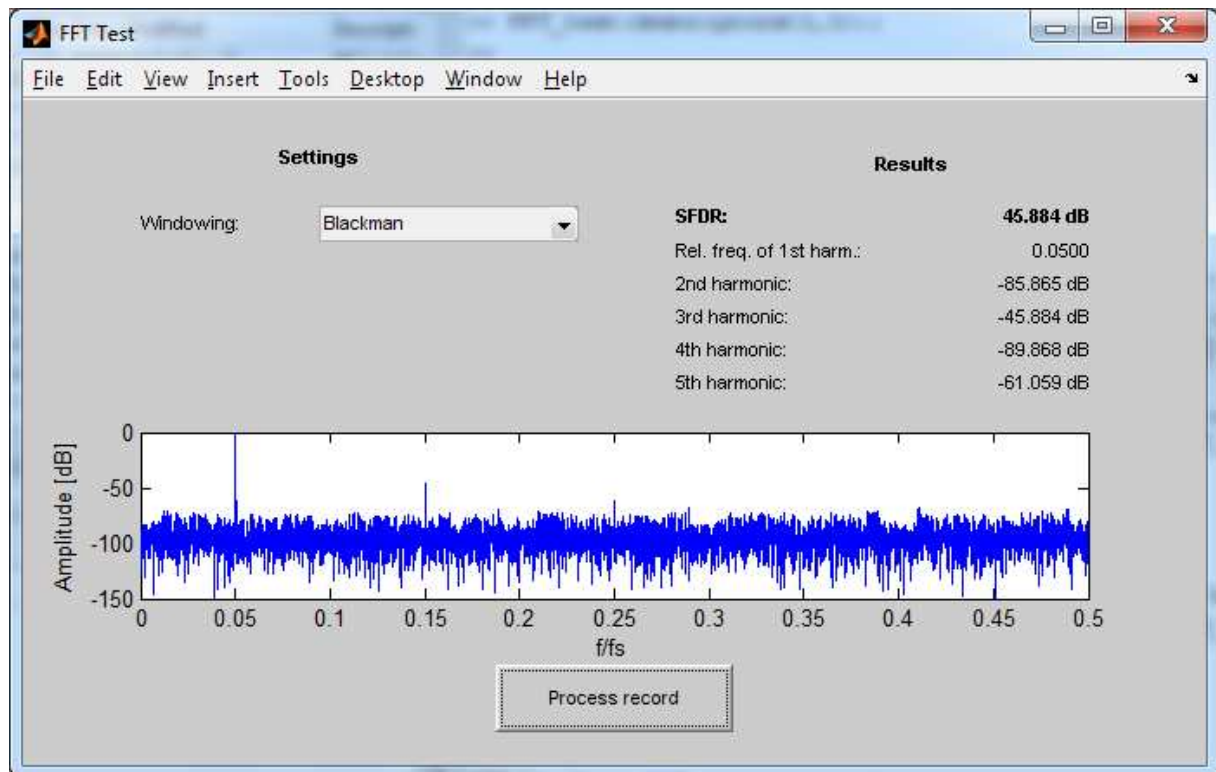


Fig. 5 The result of FFT test

III. Graphical User Interface

The GUI offers these kinds of functionalities in the main window: data handling options are collected into the frame at the right side, and data processing possibilities are available in the frame at the bottom. As multiple measurements can be loaded in the same time, the actual one shall be selected using a selector menu at the top of the window. In the main frame the most important information about the selected measurement appear. To get familiar with the GUI it is recommended to load the example descriptors of the toolbox first. Then it is worth to create and evaluate our own measurement descriptors using the built-in tool.

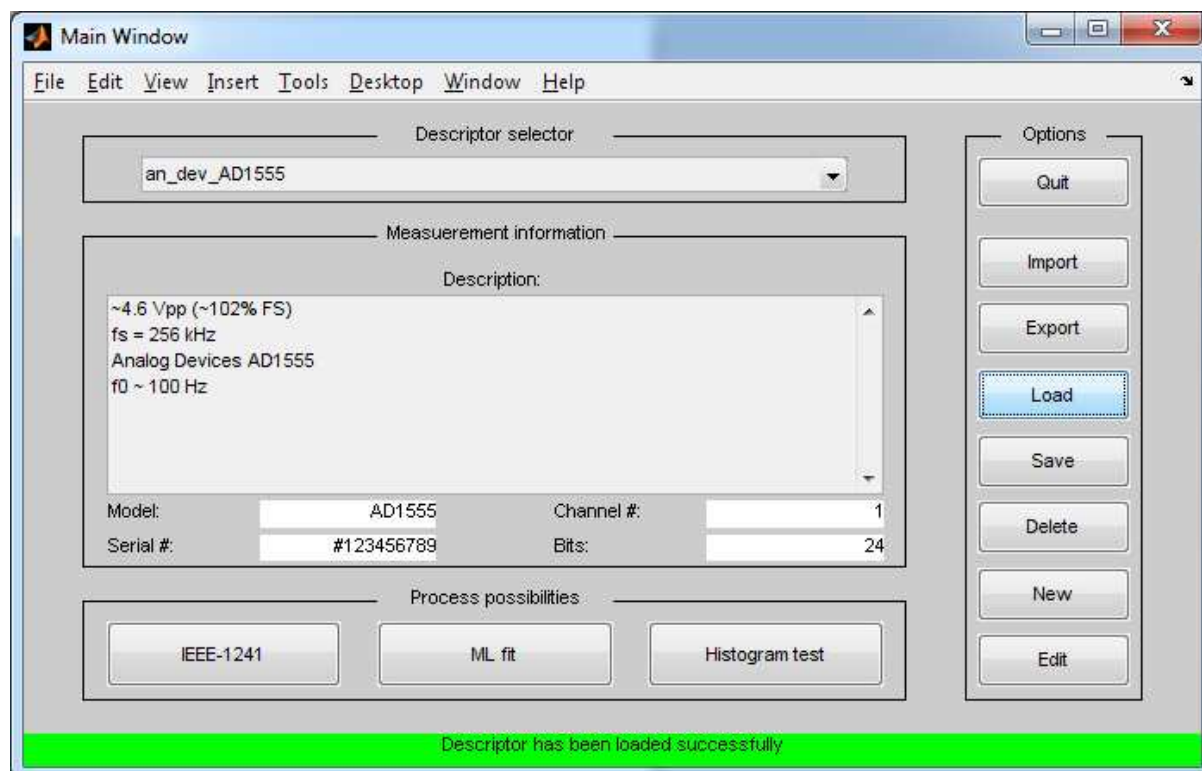


Fig. 6 Main window of the graphical user interface

IV. Data handling

Measurement results are contained in *descriptors*. The descriptor is a brief and informative report of the measurement: all related data are included in one structure. It contains information about the device under test, the circumstances of the measurement, and naturally contains the recorded data. The descriptors have mandatory parts, like the model and the serial number of the DUT, the number of bits, and the measurement record. Other information, like the number of the channel tested (in case of multi-channel devices) can be given optionally. In the descriptor further information can also be given: in case of simulated measurement, the nominal values of the signal and ADC parameters are to be stored, and it is possible to save test settings and estimation results for the measurement. Any other circumstances of the measurement can be described in text comments. The schema definition (see below) itemizes the mandatory and optional parts of the descriptor.

A. XML file handling

To store and exchange measurement information, descriptors can be stored in files. The main requirement for these files were to ensure platform-independence: the information shall be displayed in a widely used format. This way the descriptors can be exchanged between ADC test softwares written for different platforms, e. g. a LabView toolbox, a standalone application, etc. The XML format fulfils these requirements: using the XML Schema Definition (XSD) language, the format of the data file can be described properly: any software can create compatible XML files, that can be exchanged later. The schema definition file is available at the project site [5].

B. Manipulation of descriptors

Measurement descriptors can be created and modified with this toolbox: a new descriptor can be assembled from the recorded data and the additional information about the measurement, and also a new simulated measurement descriptor can be created. In simulation parameters of the DUT and the excitation signal can be set: the five parameters of sine wave (including the deviation of the additive noise) and the static transfer characteristic of the ADC. The INL vector can be imported from the workspace (we can simulate with a transfer characteristic previously measured or extracted from datasheet), or can be assembled using the GUI.

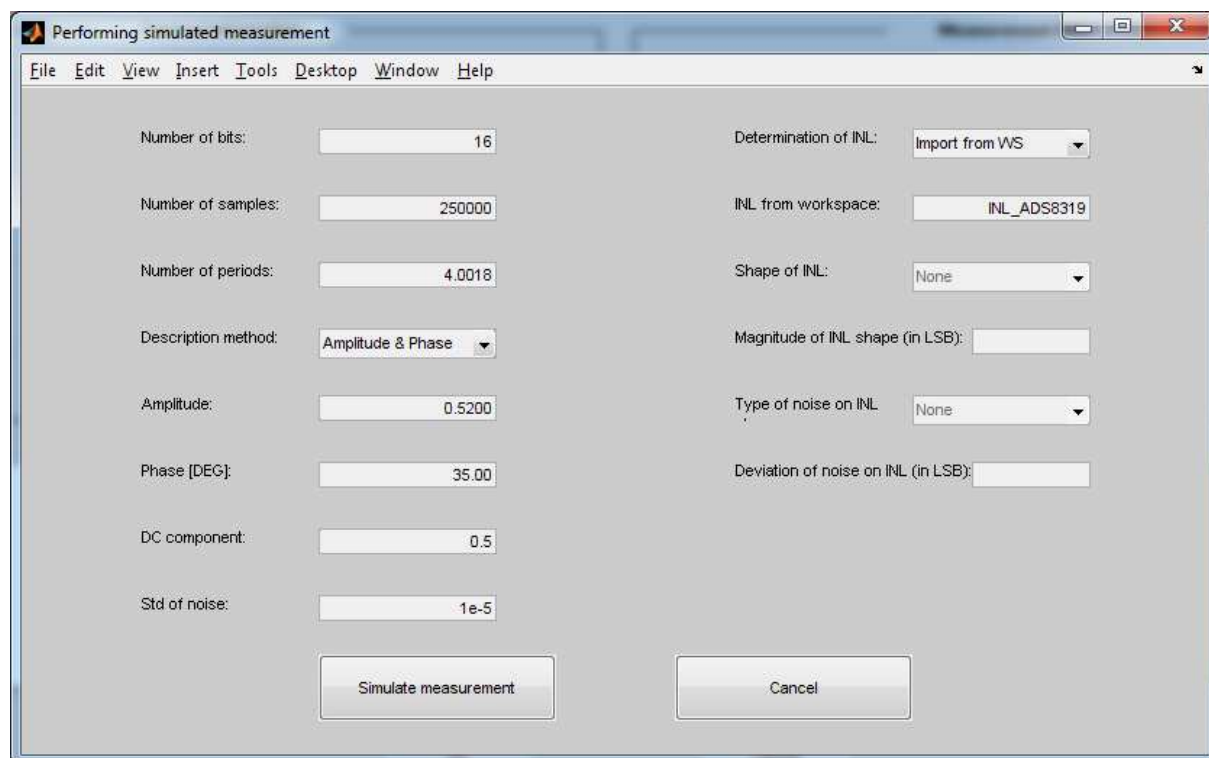


Fig. 7 Creating simulated measurement descriptor in the GUI

V. Conclusions

The toolbox ADCTest offers a possibility to create, handle, manipulate, and evaluate measurement results for ADC testing with sine waves. It also makes possible to publish reproducible evaluation results for research or ADC evaluation purposes. These opportunities could be utilized using this software widely. From the project site the MATLAB toolbox, the XML schema definition, and the user's manual in PDF format are available [5]. On longer run, a LabView equivalent of the main functionalities is also planned to be published on the web [6]. Thus, standard methods for wider comparison will be published.

References

- [1] Kollár I, Márkus J, Standard Environment for the Sine Wave Test of ADCs. *MEASUREMENT* 31:(4) pp. 261-269. (2002) doi: 10.1016/S0263-2241(01)00041-0
- [2] IEEE Standard 1241-2010, "IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters", URL: <http://ieeexplore.ieee.org/xpl/standards.jsp>
- [3] Jerome Blair, "Histogram Measurement of ADC Nonlinearities Using Sine Waves", *IEEE Transactions on Instrumentation and Measurement*, vol. 43., no 3., pp. 373-383, June 1994. doi: 10.1109/19.293454
- [4] Ján Šaliga, István Kollár, Linus Michaeli, Ján Buša, Jozef Lipták, Tamás Virosztek, "A Comparison of Least Squares and Maximum Likelihood Methods Based on Sine Fitting in ADC Testing", submitted to *Measurement*, Vol. 46, 2013.
- [5] ADCTest project site: <http://www.mit.bme.hu/projects/adctest/>
- [6] Linus Michaeli, Ján Šaliga, Jozef Liptak, personal communications.
- [7] MATLAB Optimization Toolbox: <http://www.mathworks.com/products/optimization>