IEEE Instrumentation and Measurement
Technology Conference
Anchorage, AK, USA, May 19-21, 2002

Modelling and Characterization of Pipelined ADCs

D.Dallet
Laboratoire IXL, ENSEIRB, University of Bordeaux I,
UMR-5318 351, Cours de la Libération, 33405 Talence Cedex, France
Phone: (33) 5 56 84 26 32, E-mail: dallet@ixl.u-bordeaux.fr

P.Daponte, E.Mancini, S.Rapuano
Università del Sannio, Piazza Roma 82100 Benevento, Italy,
Phone: (39) 0824 305817, E-mail: {daponte,rapuano}@unisannio.it

***Abstract** – The paper deals with the problems of modeling and testing of pipeline ADCs. In particular, such problems are faced through the realization of a modular Virtual Instrument. It has been developed in Java language in order to be remotely manageable through a common Internet browser. The instrument features includes (i) a module able in modeling an ADC through the specialization of a simplified behavioral model, also sketched in the paper; (ii) a module executing the dynamic testing of the device in frequency domain; (iii) a scalable database providing the data sharing among more remote users; and (iv) some interface modules to programmable instrumentation. The paper also presents the results of the first validation phase of the instrument, carried out on two pipeline ADCs.*

***Keywords** – pipelined ADC, modeling, virtual instrumentation, Java*

I. INTRODUCTION

Researchers consider that System-On-Chip will be the revolution of the new century in electronic design like ASIC was at the end of the last one. The problem is to propose some tools in order to simulate the behavior of such a complex structure in the design phase. In particular the ADCs play a fundamental role in interfacing the processing core with the analogue world. Therefore, many efforts are devoted to the ADC modeling. In the present work modeling has been intended as the description of the ADC architecture by different behavioral language like SIMULINK or VHDL-AMS, taking in account parameters linked to the device performances.

The ADC considered in this paper is a pipelined converter constituted of several stages, each of them composed by an ADC, a DAC and a Sample and Hold (S/H), with the aim to convert a sub-range of entire code [1]. The input full scale signal to be converted is processed by the first stage that acts a conversion of the most significant bits and sends the code both to the correction logic block and to a DAC. The DAC acts a conversion from digital to analogue and the resulting signal is subtracted to the input. Then, this error signal is sent to the next stage to be processed in a similar way. Each stage adds also a delay cycle. The logic correction block must recover the correct digital output starting from the partial outputs and delays of the stages and by applying opportune shifts. The component used as a case study within this work is a 10 bit, 20MHz ADC from Analog Devices with three stages, each containing a 4 bit flash ADC. The overall 10 bit word is obtained by means of a digital sum of the output code MSB of each flash ADC with the output code LSB of the ADC in the preceding stage[2].

The attention has been focused on a pipelined architecture because it uses lower resolution ADCs to act fast conversions with high resolution. However, the need of the S/H and the correction logic have the disadvantage of complex device design. For such a reason, a lot of interest arises both from the scientific and production points of view. In particular, the efficient modeling of such a structure is still a problem and there is an increasing demand for testing tools for pipelined ADCs.

Section II gives a SIMULINK behavioral model of the pipelined A/D converter as well as the different elements, flash ADC, Digital to Analog Converter (DAC), Sample and Hold Amplifier (S/H). An Input-Output mathematical relation is also proposed. Section III is dedicated to the methodology used for the error parameter estimation taking in account measurement results at different input amplitude levels. Finally, a couple of virtual instruments (VIs) dealing with both modeling and testing problems within the IEEE 1241 standard specifications [3] is presented. The VIs have been developed in Java, in order to be independent from the working platform and assure a native networking support. Moreover, they are modular, assuring high-grade expandability and flexibility.

II. MODEL OF A PIPELINED ADC

In Fig.1 a block scheme, in SIMULINK environment, for a three stages pipelined ADC is given. Each stage is similar to that shown in Fig.2. In order to implement S/H, ADC and DAC, simple models have been used for each block [1], [4]-[12]. The following equation has been used for the modeling of each ADC stage:

$$y(k) = Q\left(S\left(\sum_{i=0}^{n} a_i x^i(t) + b\frac{dx(t)}{dt}\right)\right)$$

where x(t) is the analog input signal, $Q(\cdot)$ is the ideal quantizer function, $S(\cdot)$ is the saturation function, a_i are the error coefficients, $b\frac{dx(t)}{dt}$ the jitter model.

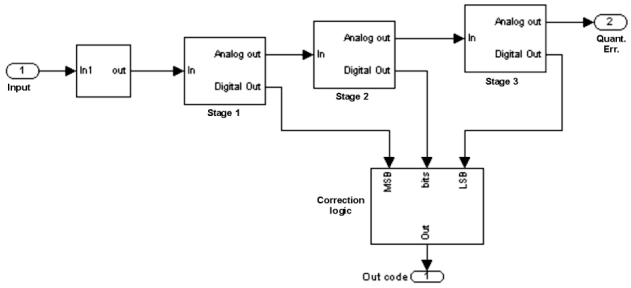

Fig. 1. SIMULINK scheme of the pipelined ADC model.

Each stage of the pipelined ADC uses a DAC as shown in Fig.2. It has been considered that this component could be described by a third order polynomial function. The following relation has been adopted for the modeling of each DAC:

$$y(t) = \sum_{i=0}^{3} c_i x^i(k)$$

where c_i are the error coefficients.

Then, the last component used in each stage, the sample and hold amplifier has been modelled by the following equation:

$$y(t) = f_t x(t) + Z_h\left(d_i x^i(t)\right)$$

where f_t is the feed-through coefficient, $Z_h(\cdot)$ is the hold function, and d_i the error coefficients.

The main target of the modelling phase is to identify the a_i, b, c_i, d_i and f_t parameter values which minimize the error between the actual and model output.

III. PARAMETER ESTIMATION

In order to evaluate the error parameters, a sine wave signal is generated as input to a real ADC (Fig.3). The obtained samples are used to estimate the analogue input sine wave parameters, following the indications given in [3]. From this estimate the amplitude, offset, phase and frequency of the signal can be used to produce an analytical signal x(t) as input to the model. The numerical results of the model are then compared with the ADC's samples. The comparison could be done in frequency

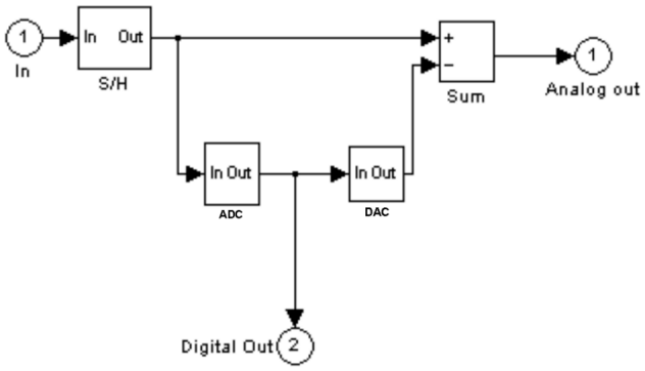

Fig. 2. SIMULINK scheme of each stage of the pipelined ADC model.

domain, in time domain or by means of an histogram. By using a multidimensional minimization algorithm, the model can be improved, reducing the difference between the two data sequences. The fitting of the model outputs to the real ADC ones depends on the chosen minimisation algorithm.

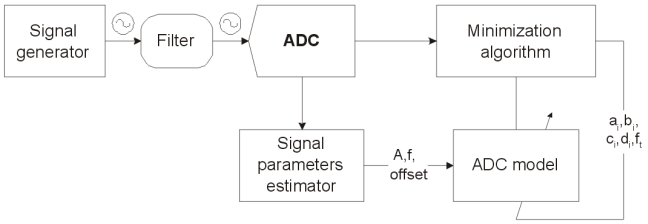

Fig. 3. Method scheme.

Many algorithms have been proposed in literature for such a task [13]-[15]. At this time a modified Aitken algorithm has been used. The modified Aitken algorithm splits the parameter range in sub-ranges, and executes iteratively the following steps:

1. For each parameter it searches the value that minimizes the error between the real and model output.
2. The step 1 is repeated from the first to the last parameter and then from the last to the first one.
3. The iterations stop when a previously fixed improvement is obtained or when the improvement obtained in the last step is under a certain percentage of the former one.

IV. THE IMPLEMENTED VIRTUAL INSTRUMENTS

Two VIs have been realised in order to execute the model calibration and the ADC testing [16]-[18]. They have been realised by using the Java language and the CORBA technology in order to ensure good portability, good expandability and the networking support. As it is shown in Fig.4, an overall system architecture has been designed. It is based on the above mentioned VIs and two external servers that are used to sample and

collect data. In particular, the second one, a document manager, sorts and distributes the measures to the working group participants. The first VI, called AdcX, implements the method described in the previous section. It takes sampled data from a file or a CORBA server and sets the parameters of a Java written model. In order to assure an easy expandability, it is as general as possible. Thanking to its modular structure, the VI leaves the user free to include his own ADC model by providing a Java written module with the required interface to the main program. In order to speed-up the development of new modules, a partially prewritten template as been realised with a set of Java interfaces that only have to be specialised. Fig.5 shows the main graphical user interface (GUI) for AdcX.

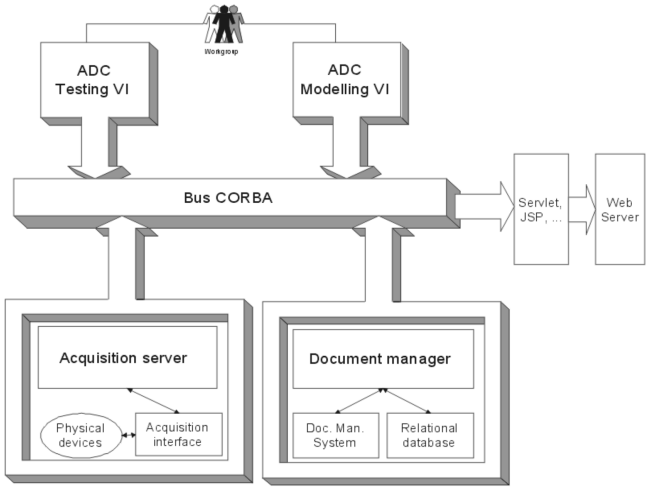

Fig. 4. System architecture.

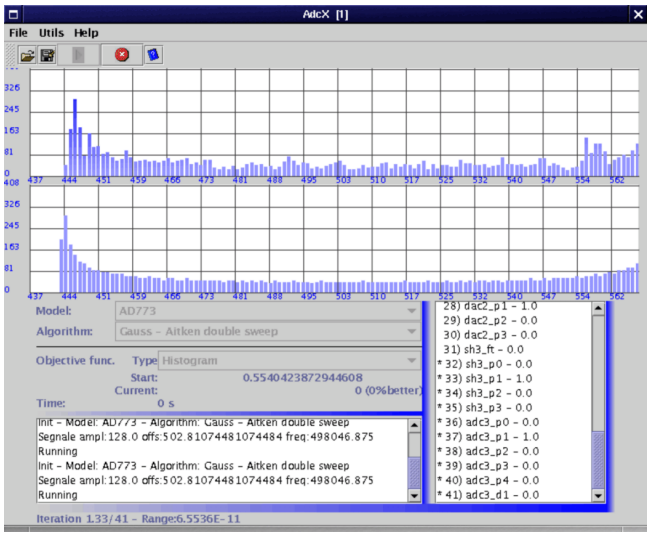

Fig. 5. AdcX GUI.

The second VI, called AdcT (Fig.6), implements the ADC testing. It retrieves sample data both from the ADC under test and from its corresponding model. Then, it carries out some statistical, and spectral analysis. The statistical analysis is based on the histogram test (Fig.7). At this time the spectral analysis is based on the FFT, as shown in Fig.8, but in future a time-frequency analysis tool will be implemented. Moreover, it supports the data coding (e.g. Gray coding) with the possibility to implement new coding algorithms with a minimum software change. Like AdcX, AdcT is designed to have a good modularity and expandability. It acquires data from a local or remote CORBA server. Therefore, in order to test ADCs by means of IEEE 488, or VXI instruments, it would be necessary to develop a connection module. At now, a software module for connecting GPIB instrumentation has been developed and another for VXI instrumentation is on the study.

As stated above, the system architecture has been designed in order to set the modeling and testing procedures independent from the specific data sources, managed by the acquisition server. In such a way, the analysis subsystems such as AdcX and AdcT software modules can work whichever is the effective data source: instrumentation, models optimized by AdcX or data coming from former acquisitions. This separation is obtained by using a common data repository, independent from the particular acquisition and the required analysis. As mentioned above the software module with the task of managing the data is called document manager. The acquired samples are sent to the document manager that stores them in opportune data structures. These contain also some information about the source, such as the sampling frequency or the particular ADC. The implemented database is scalable, in order to provide the maximum flexibility to the specific needs of the user. The data storage is carried out in a hierarchic way, in order to group data from several acquisitions with similar characteristics (e. g. from the same ADC) and retrieve them by using the above quoted characteristics instead of an arbitrary file name. An additional feature of the designed document manager is the possibility of using pre-processing modules such as a Fourier transform one, that starting from time domain data can calculate their amplitude spectrum. This feature is particularly useful in the Web interfaces to the system that enables the user in remotely analyzing the acquired data.

V. EXPERIMENTAL EVALUATION

In order to experimentally evaluate the realized VIs two Analog Devices pipelined ADCs have been used: the AD773 with a 10bits resolution and a three stages pipeline, and the AD872 with 12bits and four stages. The ADCs are stimulated with sinusoidal filtered signals at different amplitudes, in order to involve a new pipeline stage at each step. The input signal has been generated by using an Agilent 8904A multifunction synthesizer, while the output has been retrieved from an Agilent HP16500C Logic Analysis System, equipped with a 16517A/18A and a 16554A Logic Analyzers, connected to a PC through a GPIB interface. For the purposes of this first

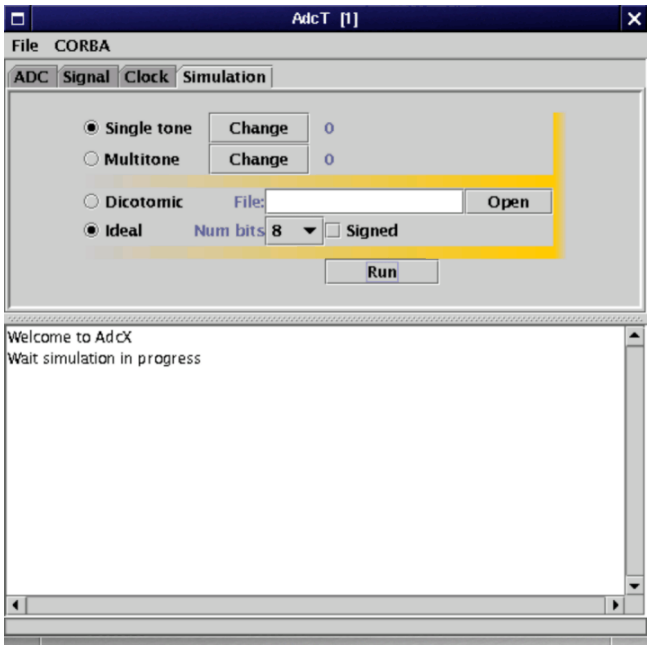

Fig. 6. AdcT GUI.

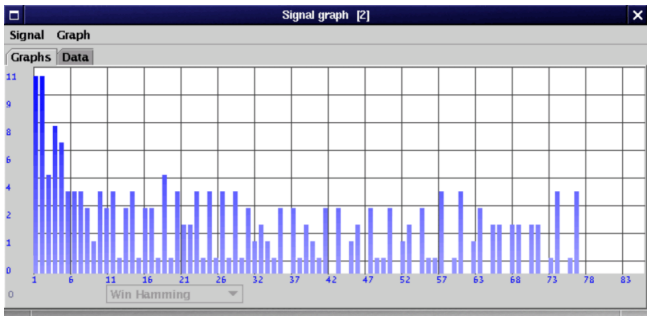

Fig. 7. Test results.

experimentation, only the spectral analysis approach has been used, but an histogram matching approach could be followed as well. A multiple step process has been adopted to set the optimal parameters value by applying the above quoted modeling technique to each stage of the ADC from the first to thelast one.

An improvement quantification indicator QI has been chosen as follows:

$$QI = \frac{f(p_0) - f(p)}{f(p_0)}$$

where p_0 is the starting point (ideal model), p is the current point in the parameter space and $f(\cdot)$ is the objective function, defined as the error between the actual ADC output spectrum and the model one, limiting the evaluation to the higher 50 harmonics (Fig.9). By using the modified Aitken algorithm and the AD773 model the starting models have been improved by 21the first stage setting, by 25% on the second stage setting, and by 64% on the third stage, corresponding to an overall improvement by 20% on the ADC model, from the spectral point of view. Note that the equation (4) gives a QI value of 0% at the minimization start (no match) and a QI = 100% if the model perfectly matches the real component. The first test results highlight the fitting of the proposed tools to the research on the pipelined ADC spectral distortion characteristics. Further experimentation will be carried out by considering the statistical properties optimization of the model. Moreover, as the model adaptation results are strictly dependent on the chosen algorithm, a research aiming to find the best fitting minimization algorithm will be carried out by adding new modules to the AdcX VI.

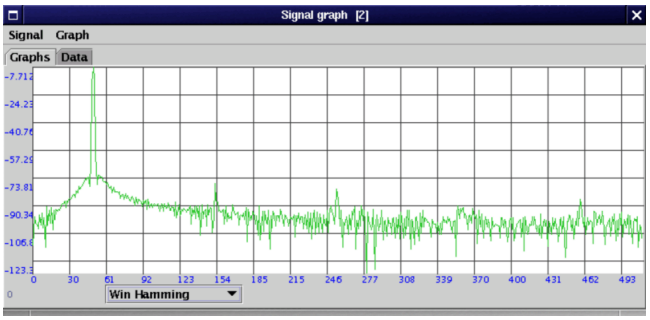

Fig. 8. Test results.

VI. CONCLUSIONS

The paper presented a simplified behavioral model of a pipelined ADC and the modeling tool implemented to fit the model to the data obtained by a real component. Moreover, a tool for the ADC testing has been set up and illustrated. At the present time, both the tools implement a limited number of features, as the AdcX module works only with the Aitken algorithm and both work only in frequency domain, but they have been realized with a modular structure that allows an easy expandability with the addition of more optimization algorithms or testing procedures. Additional modules are currently being developed in that direction. The module implementation in Java language with the use of CORBA architecture allows their remote managing through a common browser without requiring the use of proprietary software or complex setup procedures. This feature enables the collaboration of geographically distributed research groups on the themes of ADC modeling and testing.

References

[1] D.F. Hoeshele, *Analog-to-digital and digital-to-analog conversion techniques*, John Wiley and Sons, 1984.

[2] P.Real, D.H.Robertson, C.W.Mangelsdorf, and T.L.Tewksbury, "A wideband 10-b 20-ms/s pipelined adc using current-mode signals," *IEEE J. Solid-State Circuits*, vol. 26, no. 8, pp. 1103-1109, 1991.

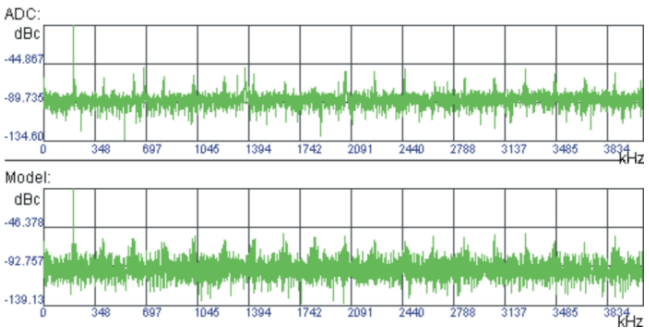

Fig. 9. Spectrum comparison on stage 2 setting.

[3] *IEEE 1241 standard, Terminology and test methods for analog-to-digital converters*, IEEE TC-10, 2001.

[4] P.Arpaia, P.Daponte, and Y.D.Sergeyev, "A-priori statistical parameter design of ADCs by a global optimisation algorithm with local tuning," accepted for publication on IEEE Trans. on Instrum. and Measurement.

[5] P.Arpaia, P.Daponte, and L.Michaeli, "The influence of the architecture on adc error modelling," *IEEE Transactions on Instrumentation and Measurement*, vol. 48, no. 5, pp. 956-966, 1999.

[6] P.Arpaia, P.Daponte, and L.Michaeli, "A dynamic error model for integrating analog-to-digital converters," *Measurement*, vol. 25, no. 4, pp. 255-264, 1999.

[7] P.Arpaia, F.Cennamo, P.Daponte, and H.Schumny, "Modeling and characterization of sigma-delta analog-to-digital converters," in *Proc. of IEEE Instrumentation and Measurement Technical Conference*, St.Paul, Minnesota, USA, 1998, vol. 1, pp. 96-100.

[8] P.Arpaia, F.Cennamo, and P.Daponte, "Calibration and error compensation of analog to digital scan converters," *Computer Standard and Interfaces*, vol. 19, pp. 119-128, 1998.

[9] P.Arpaia, P.Daponte, and L.Michaeli, "Analytical a-priori approach to phase-plane modeling of SAR A/D converters," *IEEE Trans. on Instrum. and Measurement*, vol. 47, no. 4, pp. 849-857, 1998.

[10] P.Arpaia, F.Cennamo, and P.Daponte, "An error correction technique for scan conversion-based transient digitizers," *IEEE Trans. on Instrum. and Measurement*, vol. 48, no. 4, pp. 785-792, 1999.

[11] R.Holcer, L.Michaeli, "DNL ADC by the experimental shaped voltage", IMTC Budapest, May 21-23, 2001, pp. 693-697.

[12] L.Michaeli, J.Saliga, V.Sedlak, "An approach to diagnostic of the A/D converter embedded on ATMEL microcontrollers", Proc. of 4^{th} Workshop on ADC Modeling and Testing, Bordeauz, Sept. 9-10, 1999, pp. 247-252.

[13] W.T.Vetterling, W.H.Press, and S.A.Teukolsky, *Numerical Recipes in C. The Art of Scientific Computing*, Cambridge University Press, 2 edition, 1997.

[14] J.E.Dennis and N.Alexandrov, *Algorithms for bilevel optimisation*, Houston, TX, USA,, 1994.

[15] W.Gomez and J.A.Gomez, *On multi-directional search in optimisation*, 1995.

[16] P.Arpaia, F.Cennamo, and P.Daponte, "Metrological characterisation of analog-to-digital converters - a state of the art," in *Proc. of 3rd IEE Int. Conf. on Advanced A/D and D/A Conversion Techniques and their Applications*. 1999, pp. 134-144, IEE ADDA-99, Glasgow, UK.

[17] D.Dallet, *Contribution la caractrisation des convertisseurs analogiques-numeriques: valuation des mthodes et mise en oeuvre de nouveaux procdures*, Ph.D. thesis, Universite de Bordeaux 1, 1995.

[18] D.Dallet, P.Daponte, "Introduction to Special Issue on ADC Modelling and Testing," *Measurement*, 2002.