

AN IMPLEMENTATION OF SINE WAVE TEST SETUP
FOR AUTOMATED MEASUREMENT OF HIGH-PERFORMANCE
ANALOG-TO-DIGITAL CONVERTERS

BY

PAVLE MILOSEVIC

B.S., University of Belgrade, 2003

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2007

Urbana, Illinois

To my family, for their love and support

ACKNOWLEDGMENTS

First and foremost, I would like to express my appreciation to my advisor, Dr. José Schutt-Ainé, for his attention, guidance, and insight during my thesis research. I thank him for providing me with the opportunity to perform research in the exciting field of electromagnetics.

Many thanks to my research group colleagues and friends, Yidnek Mekonnen, Dr. Ki-Hyuk Kim, and Dmitry Klokotov, for their swift assistance with numerous technical aspects of this thesis.

I acknowledge the Department of Electrical and Computer Engineering of the University of Illinois at Urbana-Champaign for providing me with financial support during my graduate studies.

Finally, I am forever indebted to my parents and Vesna for being the source of love, energy, understanding, and encouragement at all times.

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	viii
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.2 Motivation	2
1.3 Contents	3
CHAPTER 2 OVERVIEW OF IEEE 1241-2000 STANDARD	4
2.1 Overview	4
2.2 Defining the Selected Dynamic Parameters	6
CHAPTER 3 SINE WAVE TEST SETUP	8
3.1 Overview	8
3.2 Equipment Specification	9
3.3 ADC Evaluation Board	11
3.4 Characterizing the Jitter of a Clock Source	13
3.5 Selecting Coherent Input and Sampling Frequencies	17
CHAPTER 4 MEASUREMENT PROGRAM STRUCTURE	19
4.1 Overview	19
4.2 Front Panel	19
4.3 Setup Phase	21
4.4 Equipment Initialization	23
4.5 Acquiring Data	24
4.6 Communicating With the Logic Analyzer	26
4.7 Filtering the Clock Close-in Phase Noise	29
4.8 Calculating Measured Parameters	31
CHAPTER 5 MEASUREMENT RESULTS	34
5.1 Overview	34
5.2 Testing of Analog Devices AD9841	34
5.3 Testing of Analog Devices AD9245	35

CHAPTER 6 CONCLUSIONS	38
REFERENCES	39

LIST OF TABLES

3.1	Maximum allowable clock jitter of ADCs under test	16
5.1	Comparison of expected and measured data for AD9481 ADC	35
5.2	Comparison of expected and measured data for AD9245 ADC	37

LIST OF FIGURES

2.1	(a) Staircase transfer function and (b) nominal quantization error of a N-bit ADC	5
2.2	Defining spurious free dynamic range (SFDR)	6
3.1	Block diagram of the implemented sine wave test setup	9
3.2	Photograph of the entire test system	10
3.3	AD9245 evaluation board	12
3.4	Maximum allowable rms jitter versus fullscale analog input frequency for various ENOB	13
3.5	Phase noise curve with close-in and broadband regions	14
3.6	Integrating phase noise to compute jitter	15
3.7	Measured broadband jitter of E4433B	16
3.8	Effect of noncoherent sampling on DC, fundamental, and harmonics	17
4.1	Measurement program flowchart	20
4.2	Front panel of the measurement program	20
4.3	Block diagram: Setting test parameters	22
4.4	Block diagram: Initializing measuring equipment	24
4.5	Block diagram: Acquiring data from the ADC	25
4.6	Block diagram: Communicating with the 1692AD logic analyzer	27
4.7	Smearing of fundamental signal due to close-in phase noise: (a) spectrum before filtering, (b) notch filter reveals spurs, and (c) spectrum after filtering and restamping fundamental	30
4.8	Block diagram: Compensating for clock generator's close-in phase noise	30
4.9	Block diagram: Calculating ADC dynamic parameters	32
5.1	Spectrum of sampled signal for AD9841 ADC	35
5.2	Spectrum of sampled signal for AD9245 ADC	36
5.3	Expected signal spectrum for AD9245 ADC	37

CHAPTER 1

INTRODUCTION

1.1 Background

Analog/digital converters (ADC) represent one of the most important building blocks of most modern mixed-signal devices, due to the advances of DSP technology. They present a key component in a typical DSP chain because signal degradation introduced by an ADC placed on the border of a digital domain cannot be recovered by subsequent processing. Leading high-performance ADCs that are currently commercially available can achieve 500 Msamples/s at 12-bit resolution [1] and 16-bit, 130 Msamples/s [2]; however, for future wireless communication, radar and high-speed test equipment systems, both of those metrics will have to be significantly improved.

Along with the design, testing and measurement of state-of-the-art ADCs also presents a number of significant challenges, both conceptually and implementation-wise. Modern high-speed and high-resolution systems call for redefining the metrics used to characterize their performance. There is an increased trend of migrating test methods from time to frequency domain, due to the latter being a native environment for spectral analysis of individual signal components and also easily implementable on modern computer systems using fast DSP algorithms.

Previously, the ADC market was suffering from a lack of standard terminology in describing ADC behavior, with many vendors providing their own definitions of key metrics and specifications, thereby making the process of selecting a particular ADC for a given

application difficult. With the emergence of IEEE 1241 Standard in 2001 [3], a common platform was established for comparing the performance of various ADCs, both commercial and novel designs.

1.2 Motivation

The importance of an accurate measurement and characterization system in the flow of designing novel mixed-signal devices cannot be overestimated. In order to meet the ever-increasing market demand for better performance, new deep-submicron processes are developed and clock frequencies are increased, while power supply voltages are being lowered. All these factors significantly contribute to noise and can compromise signal integrity if not properly accounted for. At the same time, simulation models used in design stages need to grow in complexity in order to accurately predict system behavior.

On the other hand, what ultimately matters for an ADC to be used as a part of a signal-processing system are not the simulation-obtained values, but whether its real-life performance conforms with often stringent standards needed for a particular application. Therefore, being able to obtain reliable and accurate measurements of ADC performance is an indispensable stage of every novel design.

The system described in this thesis was developed as part of a larger effort to design a high-performance ADC from the signal integrity standpoint. The direct motivation for designing a custom system was the need to evaluate the performance of the newly designed device, but also to create a vendor-independent solution that could be used to verify the validity of the PCB design techniques proposed to improve signal integrity of data converters.

Existing manufacturer-provided testing systems are created to verify the performance of that specific manufacturer's devices, and cannot be modified to characterize custom-designed ADCs. Furthermore, they cannot provide the automation of the test process

because they do not have the means of controlling the measurement instruments available in a given laboratory setup.

1.3 Contents

The material presented in this thesis is organized in four main parts. First, the IEEE 1241-2000 Standard for testing ADCs is introduced, and its key concepts and benefits are explained. Important metrics of ADC performance are identified and defined.

The sine wave test setup for testing ADCs is presented next. The overall concept of this testing scheme is explained. Hardware chosen to implement the scheme is discussed. Several hardware issues and limitations are identified, along with the methods of characterizing and overcoming them.

Next, the LabVIEW program written for measurement automation, instrument control, data acquisition, and postprocessing is discussed in detail. Both custom-written code and modification of vendor-provided code are documented. Each of the steps in program flow is illustrated in the form of a block diagram. Where appropriate, enhancements to existing algorithms and potential improvements are suggested.

Finally, the proposed hardware and software setup is used to measure the dynamic parameters of two high-performance commercial ADCs, in order to test the validity and operation of the setup against manufacturer-specified performance metrics used as reference. Measurement results are presented and sources of discrepancies are discussed.

CHAPTER 2

OVERVIEW OF IEEE 1241-2000 STANDARD

2.1 Overview

IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters (IEEE Std 1241-2000) identifies ADC error sources and provides test methods with which to perform the required error measurements. The emergence of this standard provides the users with a robust and consistent set of terms, definitions, and test methods, in order to facilitate specification, characterization, and evaluation of ADCs produced by different manufacturers [4].

The standard takes into consideration the ADCs whose output is discretized in both value (quantized) and time (sampled). The model used divides the full-scale input range into uniform intervals - code bins with nominal width Q , as shown in Figure 2.1(a). The lowest code bin is numbered 0, the next is 1, and so on up to the highest code bin, numbered $2^N - 1$ (for an ADC with resolution, or digitized number of bits, equal to N). In case the ADC under test employs a different type of output coding scheme (e.g., 2's complement, Gray, BCD, etc.), consecutive mapping of its output codes to unsigned binary can be used to adapt the tests described in the standard. The number of code transitions in the discrete input-output transfer functions is also $2^N - 1$, if no missing codes exist at the output. Either the midtread (where the first transition is $Q/2$ above the minimum input value V_{min}) or midriser convention (where it occurs Q above V_{min}) can be used, with no

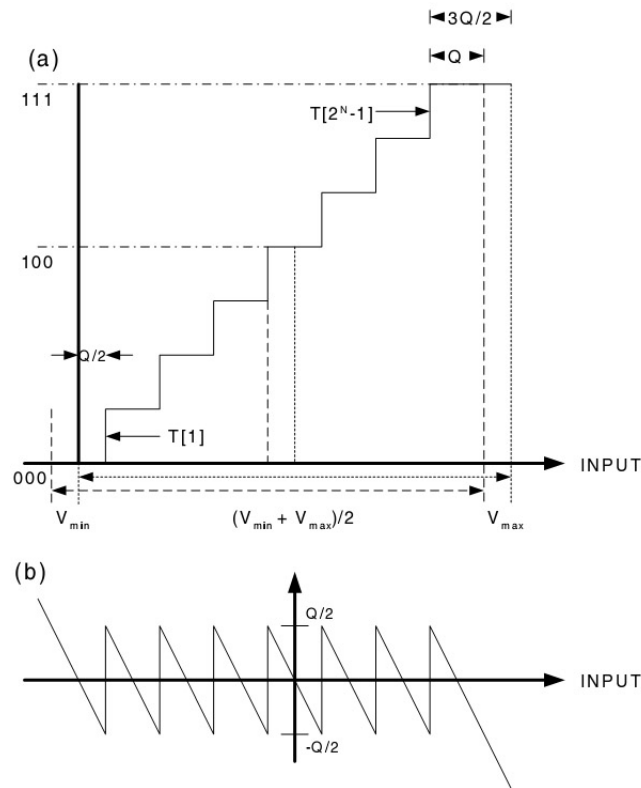


Figure 2.1 (a) Staircase transfer function and (b) nominal quantization error of a N-bit ADC

effects on the result, the only difference being a displacement along the input axis by an amount of $Q/2$.

The Standard defines a large number (more than 30) of ADC performance parameters that can be used to characterize a given ADC, not all of which are relevant for the intended application of a particular converter. For example, high-speed ADCs often have poor static performance metrics (e.g. integral and differential nonlinearity), but if they are used to sample fast changing signals, the quality of the sampled signal will depend much more on the dynamic performance of the ADC used. The user would therefore usually want to test just a subset of all possible parameters. To help make this choice, the Standard provides a table of critical ADC parameters for a given typical application.

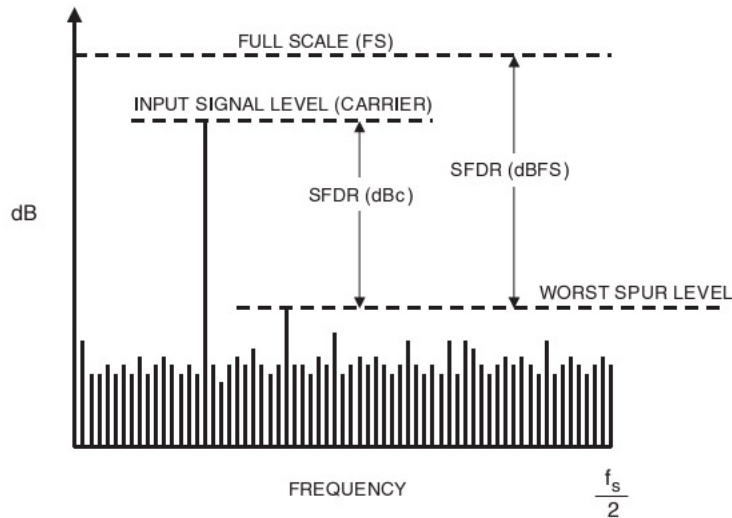


Figure 2.2 Defining spurious free dynamic range (SFDR)

2.2 Defining the Selected Dynamic Parameters

Defined below are the most important dynamic parameters, whose measurement will be the focus of the system described later. Unless otherwise specified, the input to data converter is defined as a pure sinewave input with a known frequency f_i and amplitude A_i . All the measured parameters are functions of both f_i and A_i , as well as sampling clock frequency f_s . It is also worth noting that the value of parameters can be expressed relative to the full-scale input (in dBfs) or relative to the carrier amplitude (dBc).

- *Spurious-free dynamic range* (SFDR) is the ratio of the amplitude of the ADC's output averaged spectral component at the input frequency, f_i , to the amplitude of the largest harmonic or spurious spectral component observed over the full Nyquist band (DC - $f_s/2$). An illustration of SFDR is given in Figure 2.2.
- *Signal-to-noise and distortion ratio* (SINAD) is the ratio of the root-mean-square (rms) amplitude of the output signal to the rms amplitude of the output noise (where noise includes random errors, nonlinear distortion, and sampling jitter errors).

- *Signal to nonharmonic ratio* (SNR) is defined for a pure sine wave input of specified amplitude and frequency, the ratio of the root-mean-square amplitude of the output signal to the rms amplitude of the output noise which is not harmonic distortion.
- *Total harmonic distortion* (THD) is the root-sum-of-squares (rss) of all the harmonic distortion components including their aliases in the spectral output of the ADC. THD can also be expressed as a decibel ratio with respect to the root-mean-square amplitude of the output component at the input frequency.
- *Effective number of bits* (ENOB) is defined as (2.1):

$$ENOB = N - \log_2\left(\frac{rms_noise}{IRQE}\right) \quad (2.1)$$

where N is the ADC resolution (number of output bits) and $IRQE$ is the ideal rms quantization error. $IRQE$ can be found as the standard deviation of the ideal ADC's assumed uniform-distribution error ("sawtooth error," as shown in Figure 2.1(b) to obtain the well-known result, $IRQE = Q^2/12$, thus giving (2.2)

$$ENOB = \log_2\left(\frac{V_{max} - V_{min}}{rms_noise \times \sqrt{12}}\right) \quad (2.2)$$

ENOB represents another measure of the rms noise used to compare actual ADC performance to an ideal ADC - an ADC with nominal output resolution of $N = 16$ bits, but with $ENOB = 14.0$, produces the same level of rms noise as an ideal 14-bit ADC.

It is also worth noting that, starting from the definitions, a number of useful relations between the parameters can be established [5], [6], some of which will be discussed and used later in this thesis. Those relationships can be used to simplify the test process, where only some of the parameters are calculated from the measured data, and others are derived from them. Alternatively, they can be used as a method of verification in case all the discussed parameters are measured, since it is expected that for a correctly performed measurement the relations should still hold.

CHAPTER 3

SINE WAVE TEST SETUP

3.1 Overview

Back-to-back testing techniques that were dominant in the early days of ADC testing can no longer be used because of the need for the DAC used to have an accuracy significantly greater than the ADC under test. For high-performance ADCs, such a condition (at least 2 bits more accurate) would mean that an 18-bit DAC would be needed to test a 16-bit ADC, which is not widely available with the required fast settling times and low level of distortion [5]. Therefore, the actual evaluation of the ADC should be performed using DSP techniques in frequency domain.

Several different test setups are proposed in IEEE Standard 1241, depending on the intended application of the ADC and the set of desired parameters to be measured. Frequency domain testing is the most prevalent technique used in the ADC community (although the sine-fit test, also described in the standard, provides an alternative way of measuring some parameters). It has the advantage of input signals being relatively easy to precisely generate in a standard laboratory setting, as opposed to the step signal setup (obtaining precision pulse and step signals, as well as phase-linear filters, can prove difficult) and arbitrary signal setup (generating a linear ramp becomes impractical for testing ADCs of more than 12-bit resolution).

The block diagram of the implemented sine wave test setup is shown in Figure 3.1. The basic principle of operation conforms to the theory discussed in previous chapters. ADC

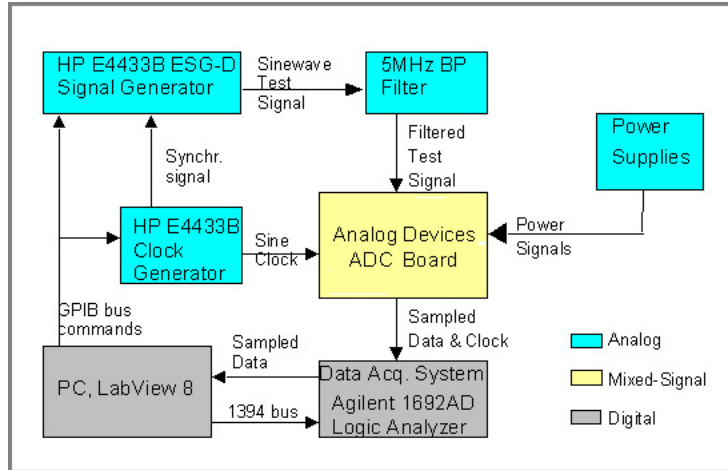


Figure 3.1 Block diagram of the implemented sine wave test setup

under test is provided with a pure sinewave input, whose frequency is carefully chosen in regards to the clock frequency of sampling. A sufficient number of samples needed for FFT is retrieved (possibly using subsampling if the ADC under test is faster than the sample retrieval equipment can handle). Acquired data is sent to the workstation, where it is postprocessed and measurement parameters are calculated. The program also performs test automation, controlling the analog and clock generators and data retrieval equipment.

3.2 Equipment Specification

The assembled sine wave test setup is shown in Figure 3.2. It consists of two phase-locked synthesizers, a bandpass filter, two DC power supplies, a logic analyzer, and a workstation on which the LabVIEW program is running. The connections to signal synthesizers are via GPIB interface, while logic analyzer is accessed via a 1394 (Firewire) interface.

For generating clock and analog input sine wave signals, two Agilent E4433B signal generators were used. The analog input generator should be phase locked to the clock generator in order to establish a known relationship between the signal frequency and

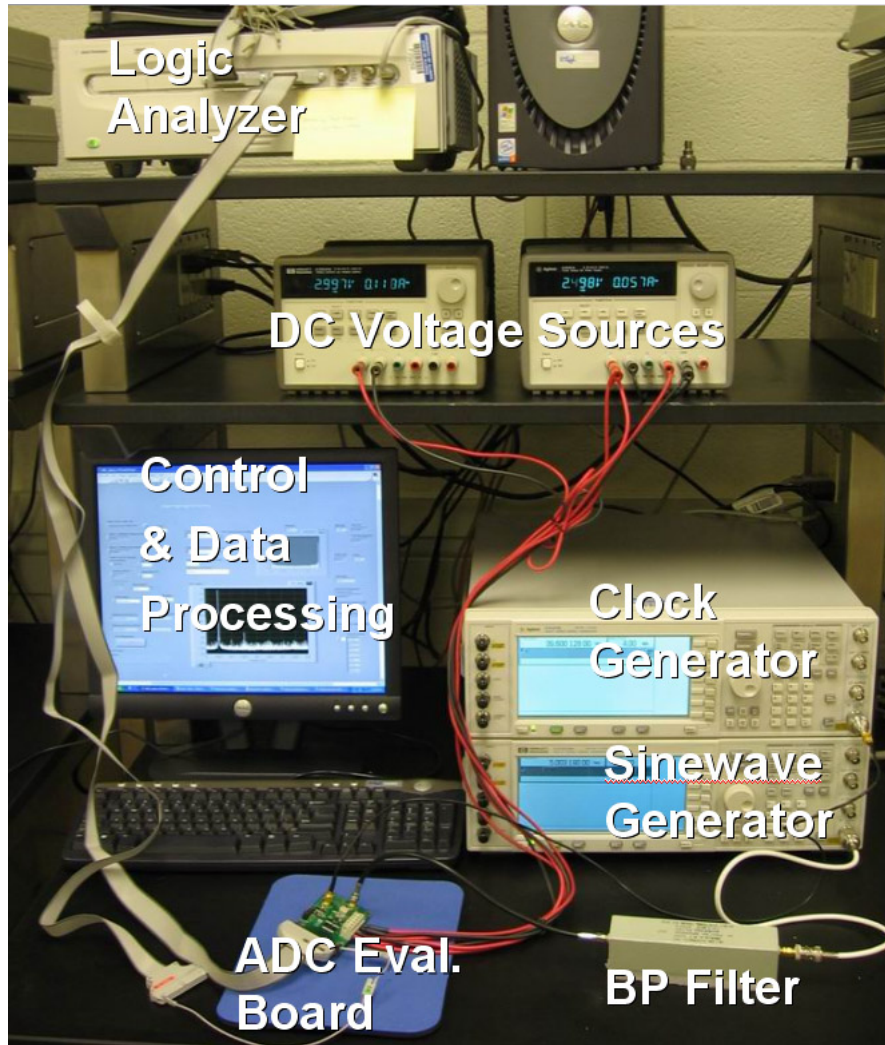


Figure 3.2 Photograph of the entire test system

sampling frequency. In order to measure dynamic performance of the ADC, its analog input should be driven by a large sine wave signal (one whose peak-to-peak amplitude covers the entire full scale, but does not cause clipping). Selection of the clock and analog input frequencies will be discussed separately.

In general, in order not to affect the measurements, harmonic distortion of analog input sinewave needs to be at least 10 dB better than the SFDR of ADC under test [7]. Since the distortion specification of the E4433B was unacceptably high (declared at -30 dBc), a band pass filter with at least 65 dB of stopband attenuation had to be used to suppress

the harmonics. The filter used was Allen Avionics BPS05P00S, designed for testing up to 16-bit ADCs, with center frequency of 5 MHz.

For providing stable DC voltages used for power and reference, two multioutput low-noise DC power supplies Agilent E3631A were used. Voltage fluctuation did not present an issue, since the ADCs under test provide internal voltage stabilization, and their allowable range of input voltages exceeded E3631A declared uncertainty. However, special attention should be paid to grounding, in order to prevent multiple current loops from voltage supply to ground through the ADC. Recommendations on possible grounding strategies are discussed in [5], [8].

For collecting the sampled data, an Agilent 1692AD logic analyzer was used. Since the data was to be processed in a separate LabVIEW program, the logic analyzer was used only for data retrieval, effectively having the role of a high-speed FIFO buffer. In state (clocked) mode of operation, 1692AD's maximum clocking speed is 200 Msps, with data depth of up to 1M samples. The specifications are adequate for dynamic tests; in order to test static parameters, a greater number of samples would have to be acquired using the "stitching" technique of several data sets [9]. If a high-speed ADC operating at more than 200 MHz was to be tested, it would be necessary to utilize the subsampling technique [5].

3.3 ADC Evaluation Board

A critical part of the ADC test setup is the PCB board on which the converter is located, used to bridge the gap between ADC and test equipment. For accurate and repeatable measurements, it is important not to damage the signal integrity of high-quality input signals, and also to precisely capture the output for analysis, yet the operating environment of the converter inherently suffers from effects such as simultaneous switching noise, power/ground plane edge radiation, and board trace parasitics. This prohibits the use of an off-the-shelf PCB, but rather calls for the design of a dedicated evaluation board

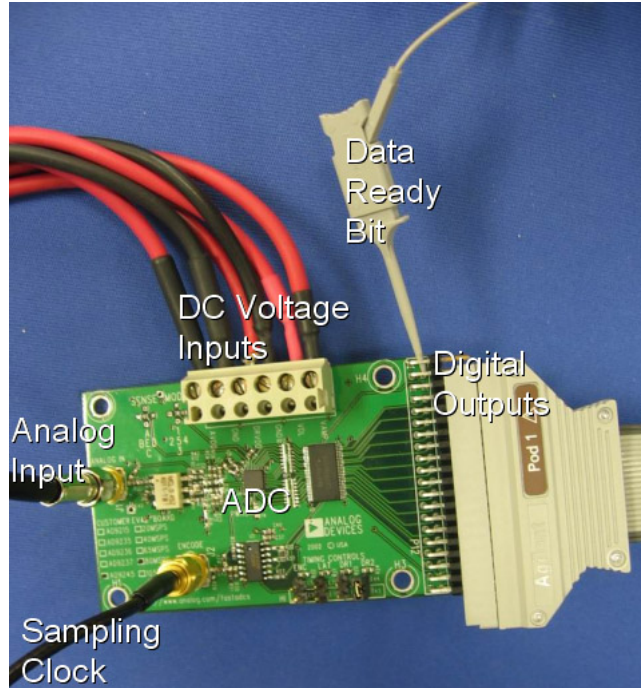


Figure 3.3 AD9245 evaluation board

for each ADC model, designed to take into account all the intricacies of the particular converter under test.

For commercial ADCs, usually the chip manufacturer provides the evaluation PCB whose layout is designed to optimize the performance of the data converter. Such was the case with the two Analog Devices ADCs tested with the proposed system, one of which is shown in Figure 3.3. In case of testing a custom-built ADC, an appropriate evaluation board would have to be designed and produced. Excellent starting points for designing a custom PCB are the manufacturer-provided electrical schematic, parts list, and board layout of its own evaluation boards. Each layer of the multilayer board is shown, and usually the CAD layout files will be provided to the user as well if needed. Many system level problems can be avoided by examining the working board layout and using it as a guide. A good review and checklist is given in [8], while a comprehensive reference for evaluation board design is [5].

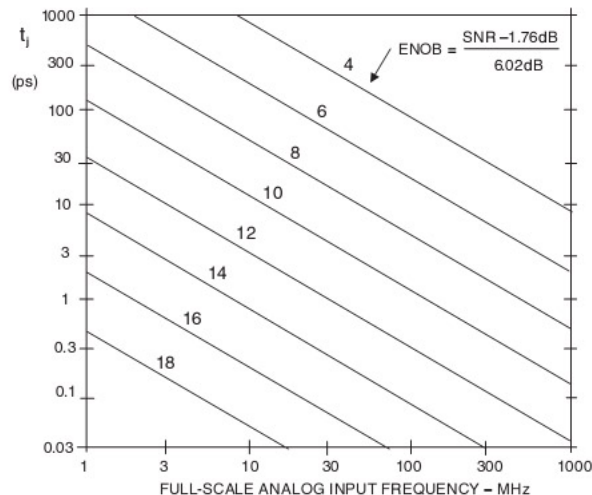


Figure 3.4 Maximum allowable rms jitter versus fullscale analog input frequency for various ENOB

3.4 Characterizing the Jitter of a Clock Source

It can be shown [10] that the theoretical SNR for an ideal A/D converter, where the only error source is assumed to be jitter, is given by (3.1)

$$SNR = 20 \log_{10} \frac{1}{2\pi f_{in} t_j} \quad (3.1)$$

where f_{in} is the full-scale analog input frequency, and t_j is the total jitter (the jitter of the sampling clock and the ADC internal aperture jitter, combined on a root-sum-square basis, since they are not correlated). In many cases, the dominant contributor to SNR degradation is the sampling clock jitter, which can be several times larger than the ADC aperture jitter. Therefore, when testing A/D converters it is of utmost importance that the sampling clock jitter is low enough so that SNR and other related parameters can be measured correctly. Figure 3.4 shows the maximum allowable RMS jitter versus fullscale analog input frequency for various ENOB (and thus SNR) values.

For general-purpose signal generators, the jitter value is usually quoted for the standard phase noise bandwidth of 12 kHz to 20 MHz, or at some other frequency range appropriate

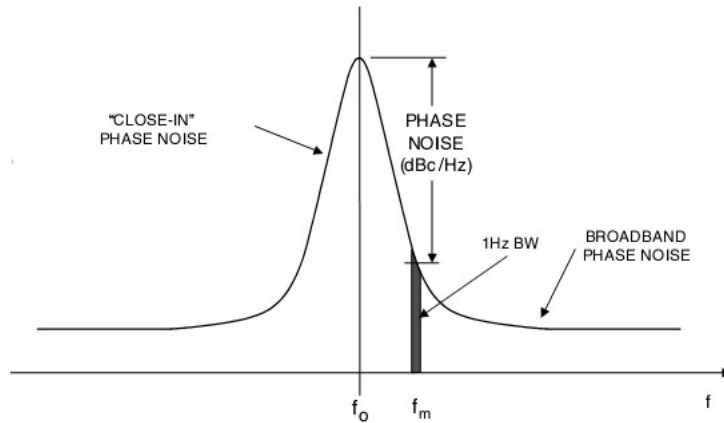


Figure 3.5 Phase noise curve with close-in and broadband regions

for a specific application, and is also given only at several frequencies of choice. However, for accurate characterization of clocking source it is necessary to get a more detailed insight into its phase noise performance. The upper frequency range for integration should be twice the sampling frequency, to approximate the bandwidth of sampling clock input. The lower frequency should be chosen so that only broadband phase noise is taken into account for SNR calculation; “close-in” phase noise has the effect of “smearing” the fundamental signal into a number of frequency bins, thus reducing the overall spectral resolution, as shown in Figure 3.5. Usually the lower frequency is chosen to be 10 kHz, depending on the application [5].

One possible approach for calculating jitter from phase noise is to integrate phase noise directly on the measuring instrument (e.g., a spectrum analyzer with corresponding personality installed). The other approach is to obtain phase noise values at discrete frequency offsets (e.g., at every decade), approximate the phase noise curve by a number of individual line segments, as shown in Figure 3.6, and then use a program to perform the integration by segments and calculate the RMS jitter.

Another interesting approach would be to compute jitter indirectly, by examining the degradation in SNR of an ADC as the analog input frequency increases, and then

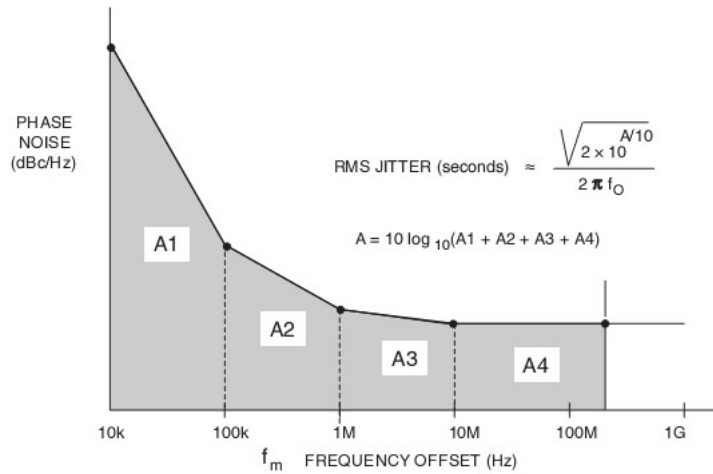


Figure 3.6 Integrating phase noise to compute jitter

subtracting the contribution of ADC aperture jitter from the total value obtained [10]. However, this method requires use of several high-quality analog input filters, which are not always available at the desired frequencies, and is also ineffective in measuring close-in jitter (although usually measuring broadband phase noise is of more interest).

E4440A Spectrum Analyzer with Option 226 (Phase Noise Measurement Personality) was used for characterizing the phase noise and jitter of the Agilent E4433B ESG-D clock sinewave generator. The procedure for measuring phase noise and integrating jitter is detailed in [11]. Broadband jitter was measured at a range of frequencies 10 MHz - 1 GHz. The settings used were as follows: E4433B Output RF Signal Power = 0 dBm, PN2 mode; E4440A set to averaging using 5 averages, smoothing (Trace 2) selected; phase offset measured from 10 kHz to $2 \cdot f_s$, with the high end of broadband phase noise curve assumed to be constant with respect to the value at 100 MHz; integration performed using Raltron jitter calculator [12]. Measured jitter is shown in Figure 3.7.

Table 3.1 summarizes the maximum allowable jitter of the two ADCs tested. Comparing the calculated results with the corresponding values measured at 80 MHz and 250 MHz, it can be seen that this signal generator can be used to clock a high-speed AD9481,

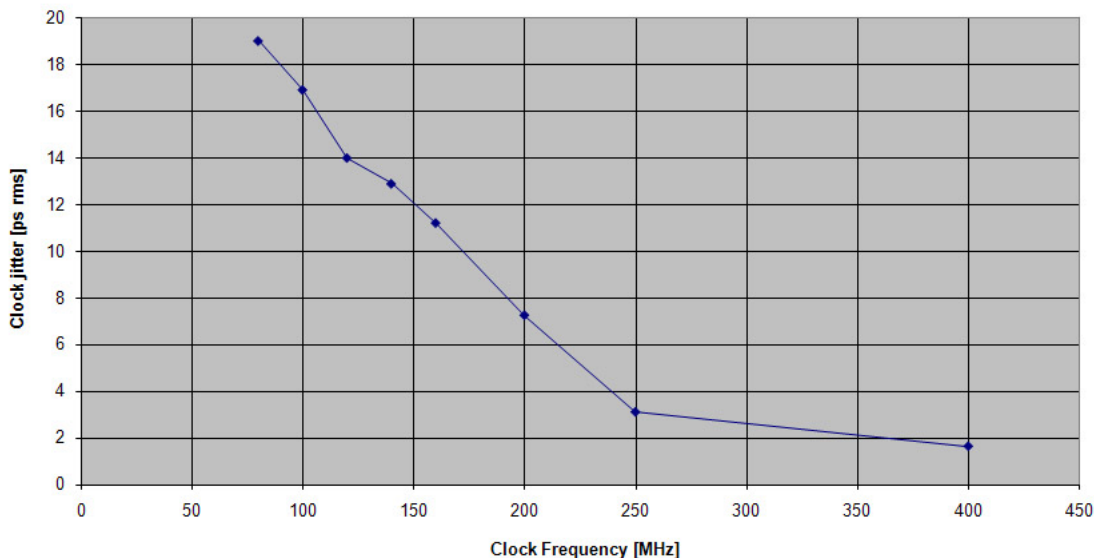


Figure 3.7 Measured broadband jitter of E4433B

Table 3.1 Maximum allowable clock jitter of ADCs under test

<i>[all values in ps]</i>	Ap. jitter	Max allowable jitter		Max allowable clock jitter	
Analog input freq.	N/A	$f_{in} = 5$ MHz	$f_{in} = f_s/2$	$f_{in} = 5$ MHz	$f_{in} = f_s/2$
AD9245 (80Msps)	0.3	7.13	0.92	7.12	0.87
AD9481 (250Msps)	0.25	159.53	7.16	159.52	7.15

due to its low 8-bit resolution (and therefore SNR), at both low f_{in} (e.g., 5 MHz) and higher frequencies closer to Nyquist. On the other hand, for clocking 14-bit AD9245, even at low f_{in} of 5 MHz, the measured jitter value is increased more than two times compared to the maximum allowable clock jitter; therefore, a drastic decrease in SNR would be expected.

However, as will be shown later, the measured SNR exhibited only a few dBs reduction, in contrast to 8 dB predicted by the theoretical formula. At the same time, effects of the close-in phase noise were very pronounced, causing the need for digital filtering to remove the obstructive spurt. Therefore, for this particular application the lower frequency of integration should be set to a value higher than 10 kHz to more accurately take into account only broadband noise.

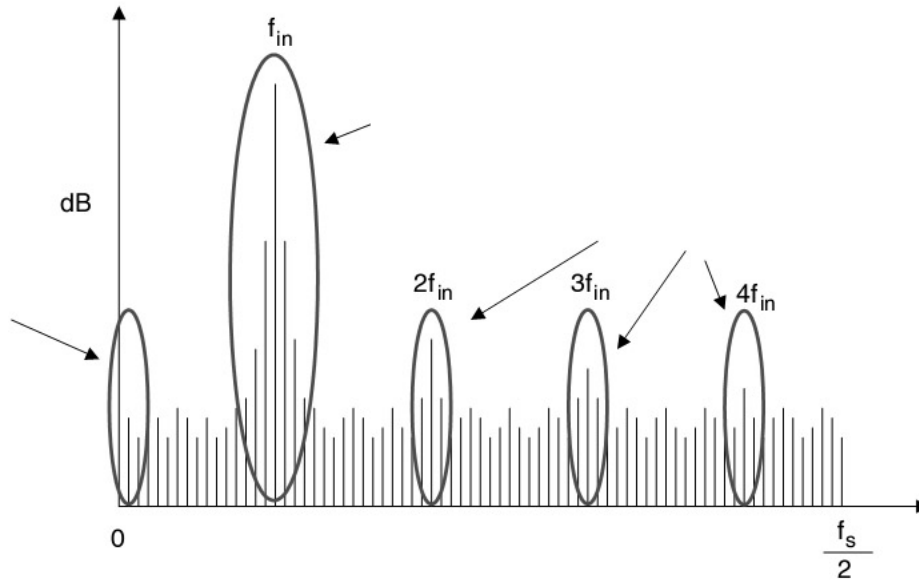


Figure 3.8 Effect of noncoherent sampling on DC, fundamental, and harmonics

3.5 Selecting Coherent Input and Sampling Frequencies

Coherent sampling is the sampling of a periodic signal with an integer number of its cycles fit into a sampling window. If this condition is not satisfied, frequency leakage will occur, with signals being smeared to several adjacent frequency bins, as shown in Figure 3.8. This effect introduces a significant inaccuracy to all the measured parameters. The errors can be reduced using windowing, but in controlled laboratory settings it is recommended to utilize coherent sampling if possible, since it yields more accurate and repeatable test results.

Standard IEEE 1241 proposes a formula (3.2) for calculating optimum clock and analog input frequencies for coherent sampling:

$$f_s \times J = f_{in} \times M \tag{3.2}$$

where J should be an integer representing the number of cycles of the input waveform in the data record of size M . However, using the proposed method with a fixed clock frequency results in an analog input frequency which in general can have as many as 10 significant digits, which can potentially present a problem for the analog input signal generator. An improved approach was proposed in [6] by offsetting the clock frequency slightly from the desired but enabling frequency selection with an arbitrary number of significant digits. The offset introduced is in the order of several kHz, which typically does not result in any noticeable discrepancy in comparing dynamic ADC measurements with manufacturer's data, since the parameters tend to vary slowly with frequency.

CHAPTER 4

MEASUREMENT PROGRAM STRUCTURE

4.1 Overview

Based on the sine wave test setup described in the previous chapter, a program was written in National Instruments LabVIEW v.8, with the purpose of controlling the measuring equipment, performing digital data retrieval, and postprocessing it to compute the dynamic parameters of ADC under test. The flowchart of the program is shown in Figure 4.1. As discussed previously, the general flow of the program follows that of the IEEE 1241 sine wave setup, with each block described in detail in the following sections.

4.2 Front Panel

As is the case with every LabVIEW program, there are two distinct parts that need to be designed: Front Panel, the interface towards the user, and Block Diagram, which defines the actual behavior of the program. Front Panel is shown in Figure 4.2, and can be divided in five parts:

1. Panel with user-specified input parameters, such as the descriptive name of ADC under test, its full-scale voltage range, resolution, the number of samples to be taken, and also the desired sampling rate and analog input frequency.
2. Panel used to specify equipment parameters, such as the GPIB addresses of clock and analog input generators, symbolic address of the logic analyzer, and power of

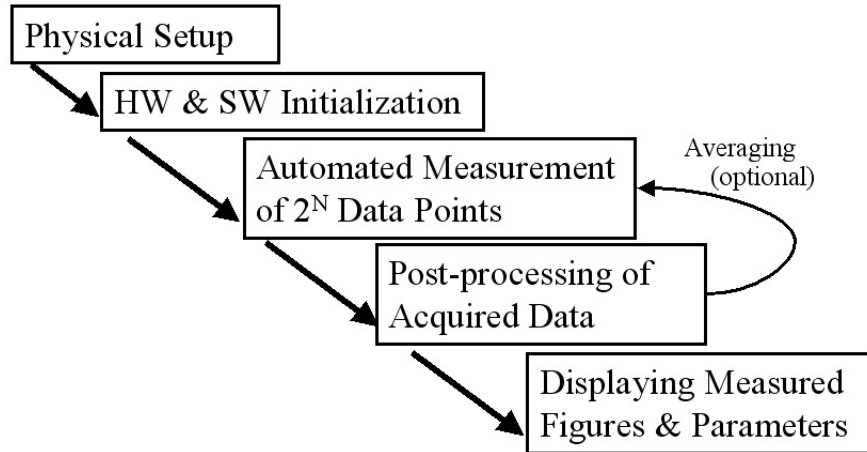


Figure 4.1 Measurement program flowchart

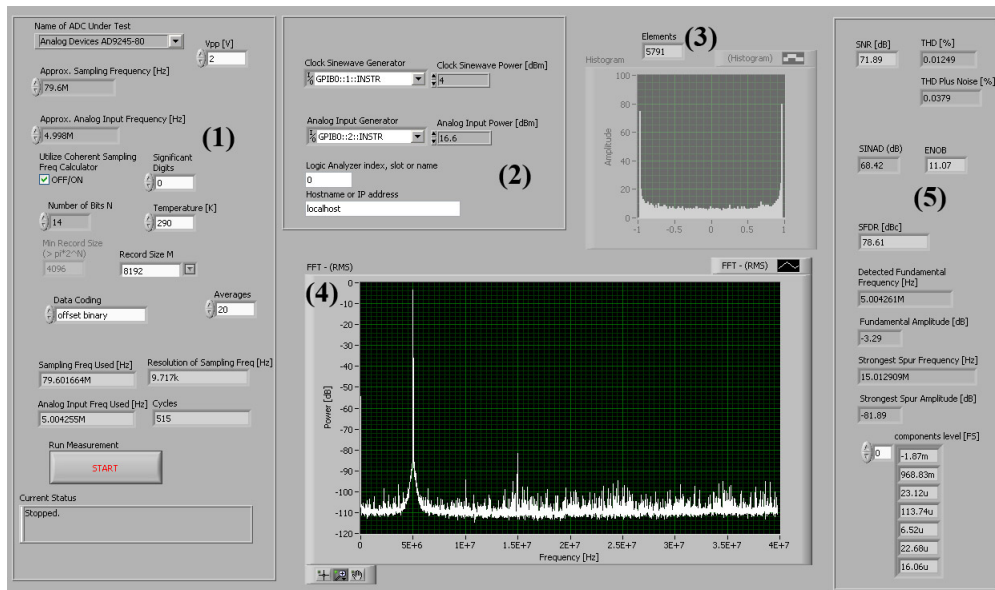


Figure 4.2 Front panel of the measurement program

generated sinewaves. As described before, input powers are chosen in accordance with ADC specifications to produce a clear clock input and a large input signal with peak-to-peak value slightly lower than the full-scale input.

3. A histogram displaying the distribution of samples across the full-scale output, as well as the number of different output values. This is primarily used to calibrate the analog input power, since it is reduced by bandpass filter, but also as an indicator of

correct system setup - in a system which is set up properly, the well-known “bathtub” shape is expected, with almost all the output values represented. This data can also be used for histogram measurement of static ADC parameters, provided that a large enough number of samples is acquired.

4. A spectrum plot of acquired data, providing a visual indication of the noise floor, fundamental, harmonics, and spurious signals. If the number of averages is set to a value greater than 1, this plot will represent the averaged spectrum, which provides a more clear representation of spurs.
5. Measured parameters of the ADC under test, including the detected fundamental and strongest spur frequency, amplitudes of DC, fundamental and first five harmonics (relative to full-scale), as well as the dynamic parameters previously discussed.

4.3 Setup Phase

Front Panel objects serve as a bridge which enables user-program interaction - the values of some objects can be set by the user before the start of measurement, and other objects can accept values from the program. Therefore, each Front Panel object has the corresponding object on the Block Diagram. Figure 4.3 illustrates both types of Block Diagram objects used in the setup phase; its functional blocks are discussed as follows (note that, in general, inputs to LabVIEW blocks shown are entering them from left or top, and results flow out of bottom and right):

1. User-specified ADC resolution N is used to calculate minimum record size needed to produce at least one representative sample of every frequency bin, which for this setup amounts to $\pi \times 2^N$ [6]. In order to have a number which represents a power of two for later FFT processing, this number is increased to $4 \times 2^N = 2^{N+2}$. A drop-down list with possible power-of-two values is generated, and if the user instead keys

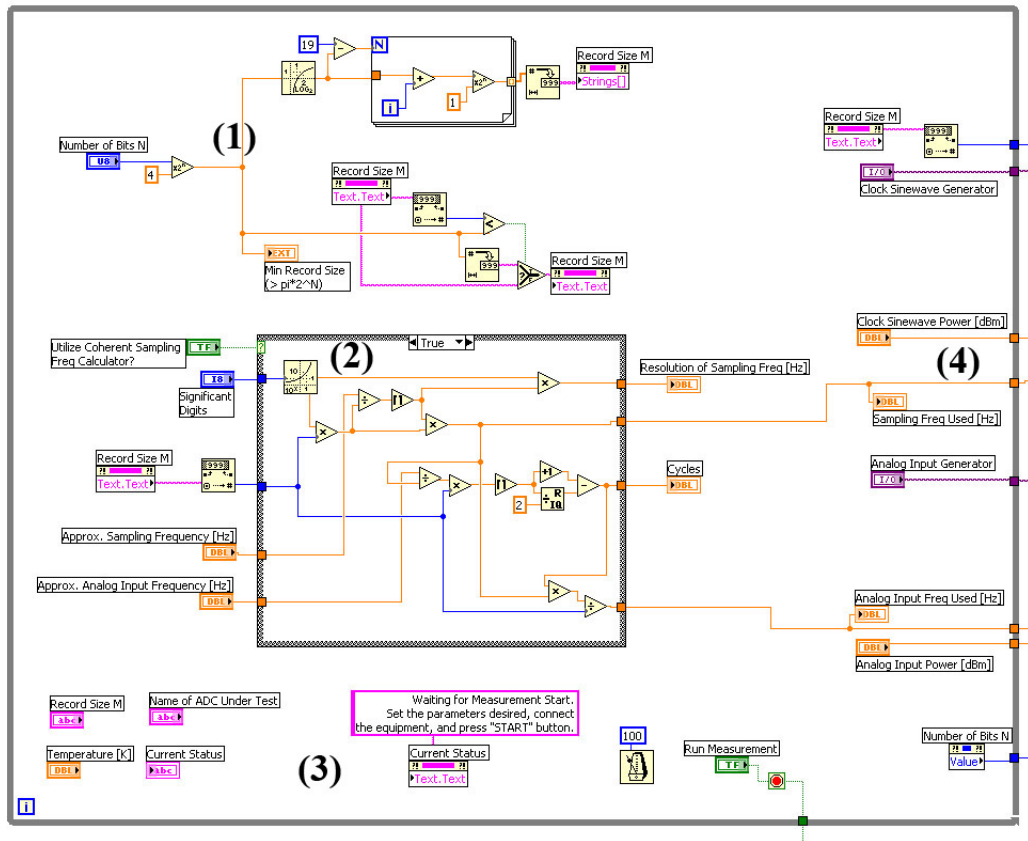


Figure 4.3 Block diagram: Setting test parameters

in a number lower than the minimum acceptable, the program rounds it up to the precalculated minimum record size.

- As with number of samples, there are certain limitations to the input and clock frequencies as well. Block (2) implements a coherent sampling frequency calculator, as described in [6]. Alternatively, the user can choose not to use coherent sampling, in which case the entire block (2) is replaced by shorted connections, meaning that the rest of the program will use user-set frequencies for clock and analog input instead of the optimally calculated ones. This could be used for observing the effect of various windows in postprocessing.

3. The program runs in a loop, waiting for the user to set the desired parameters, and updating the values every 100 ms (the delay was added in order not to occupy the whole processor time, since by default loops in LabVIEW execute as fast as they can). This procedure repeats until the *START* button is pressed.
4. When *START* is pressed, the calculated values of optimum sampling and analog input frequencies are passed to the next block of the program, along with other important values that will be used in subsequent blocks.

4.4 Equipment Initialization

After the setup phase, the program performs the initialization of measuring equipment, as shown in Figure 4.4.

1. Analog input sine wave generator is first reset, and then its frequency and power are modified from the default values to correspond to the ones received from the previous program block. LabVIEW's *Build Text* function is used to create the command string, which is then sent to the signal generator using VISA control objects.
2. The same pattern from (1) is applied to initialize and set up the clock sine wave generator. After a time delay, set to a pessimistic value of 5 s to give the instruments time to preset and reconfigure, the control is passed to the following block, along with several values that will be used later.

The contents of both *Build Text* boxes are as follows:

```
FREQ %fin% Hz;POW:AMPL %powin% dBm
OUTP:STAT ON
```

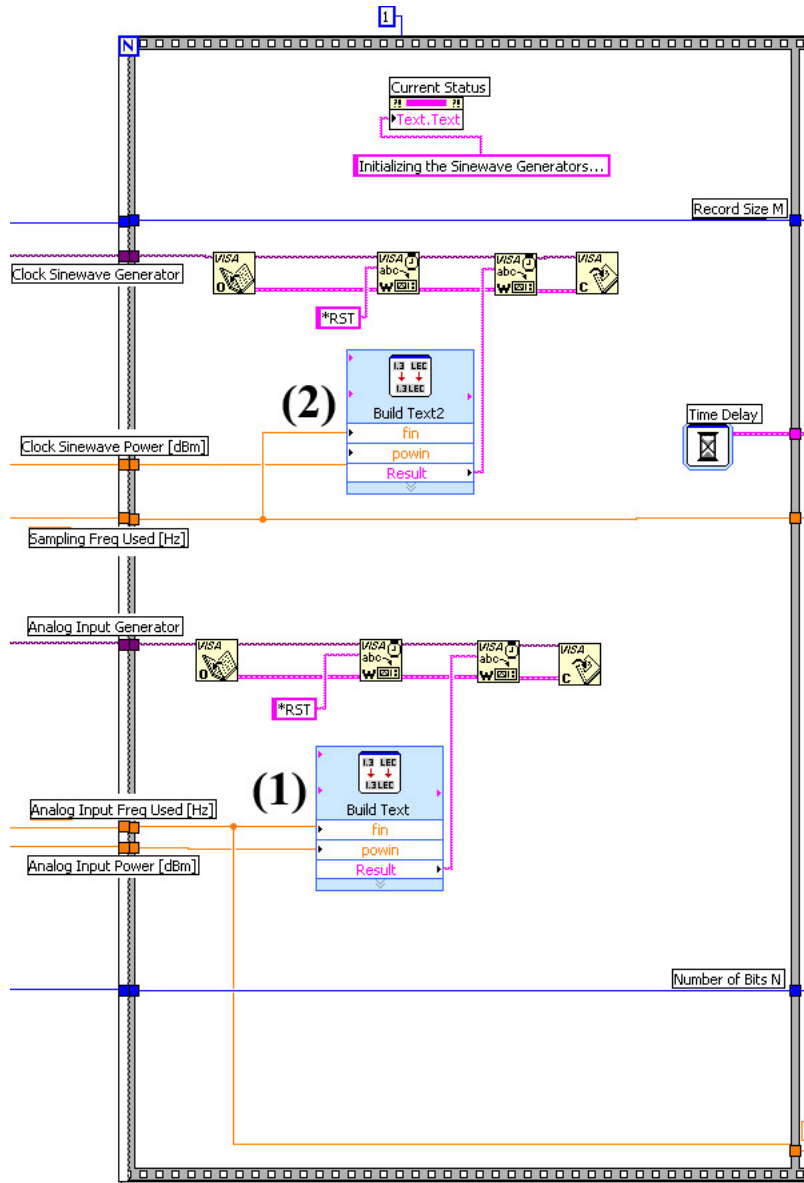


Figure 4.4 Block diagram: Initializing measuring equipment

4.5 Acquiring Data

Figure 4.5 shows the block diagram of data acquisition code. This and all subsequent blocks will be repeated as many times as the user-specified *Averages* parameter dictates. In general, using 10 to 20 averages provides a clear spectrum with pronounced spurs. How-

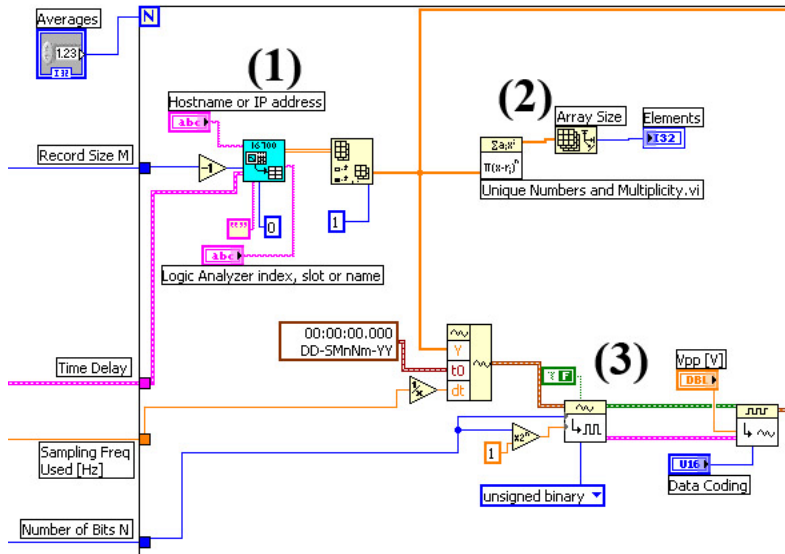


Figure 4.5 Block diagram: Acquiring data from the ADC

ever, for purposes of comparing the spectrum with vendor-specified data obtained without averaging, setting *Averages* to 1 effectively treats each measurement as an individual test.

1. In order to communicate with the logic analyzer, a modified version of sub-VI provided by Agilent [13] is used (the modifications of which will be discussed later). This sub-VI is executed after the delay object from the previous block enables it using the *Error In* signal, the standard way of implementing flow control in LabVIEW. Parameters passed to the sub-VI include the equipment address, number of samples we want to obtain, as well as some dummy control signals. The desired number of samples is then extracted from the data sub-VI returns because it is in the form of a 2-D array containing pairs of time and sample values. The time column is discarded because of insufficient time resolution guaranteed by the logic analyzer (greater of 4 ns or $\pm 0.1\%$); instead, time between samples will be calculated as $1/f_{CLK}$. The column indexed as 1 contains the sample values, in raw 16-bit unsigned format, and is passed on to the subsequent blocks for postprocessing.

It is also worth noting that in case of dual-channel output ADCs this would be the

point when the acquired data would need to be de-interleaved to produce proper results. For example, in the AD9481 which was tested, two 8-bit channels are retrieved as high and low octets of 16-bit samples. A portion of LabVIEW code (omitted here for clarity) which processes the entire array of samples and keeps alternately the high and the low octet of each should be inserted.

2. The number of unique samples is determined and displayed for diagnostic purposes. If the system was properly set up, this value should be close to 2^N , which would correspond to at least one sample of each output data code taken.
3. The raw sample values are decoded to their representative values, taking into account the output format of ADC under test, as well as the actual number of output bits N . The samples are normalized to correspond to the full scale of input values, and converted into the waveform datatype because most of LabVIEW's waveform analysis functions require input data to be of this type. The waveform is passed on to the next block.

4.6 Communicating With the Logic Analyzer

Since the physical connection of the 1692AD logic analyzer to the measurement station is via FireWire interface, native VISA objects for communicating on GPIB interface could not be used. Instead, a sub-VI provided by Agilent was modified to fit into the setup. The VI used is a part of Agilent Technologies IntuiLink 16700 Software, and uses the Agilent IntuiLink ActiveX/COM Automation Server to communicate with 1692AD.

Figure 4.6 shows the modifications made to the sub-VI.

1. In the phase of logic analyzer initialization, a custom-assembled XML string is sent to 1692AD in order to define characteristics of the digital signal to be retrieved from

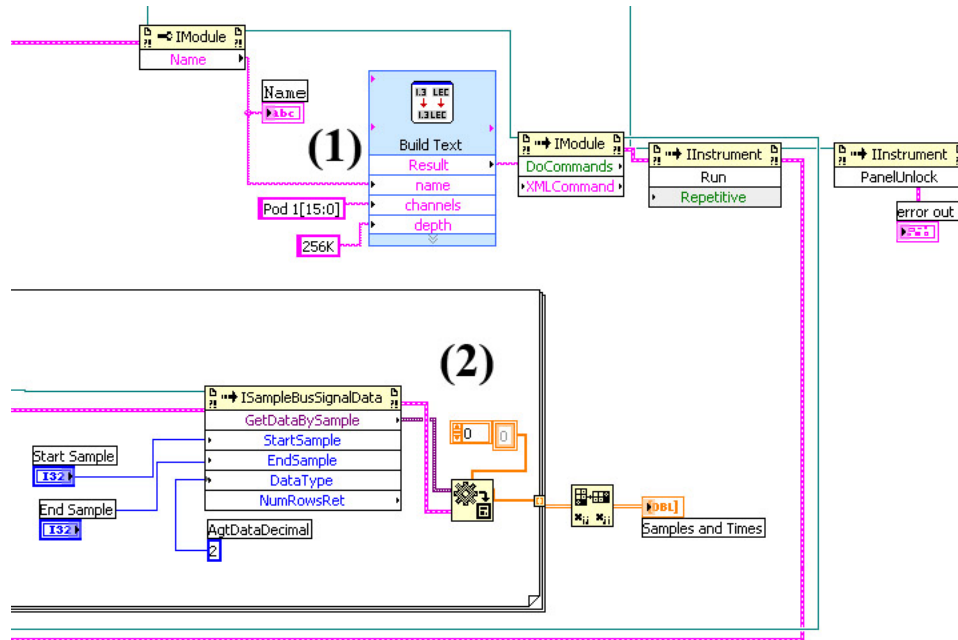


Figure 4.6 Block diagram: Communicating with the 1692AD logic analyzer

ADC. This is performed using the *Build Text* box to create the string, which is then passed to the logic analyzer by calling *XMLCommand IModule* object. Inputs to the *Build Text* box are *%name%* (referencing the analyzer in case there are more than one attached to the station), *%channels%* (defining the exact bits we want to retrieve and their ordering - in order to test ADCs of various resolution, we will always capture all 16 bits, and the main program will extract only those corresponding to real sampled data), and *%depth%* (set to a compromise value of 256 K, so that data retrieval will not take too long, and yet allow for some flexibility in the actual choice of a number of samples).

2. The normal flow of the sub-VI is then resumed, with 256 K samples retrieved and only first M actually kept (by setting $StartSample = 0$, $EndSample = M-1$ in the sub-VI call). The output data is then passed back to the main VI in the form of a 2-D array containing sample values in 16-bit unsigned decimal format (to be decoded later) and approximate sample times.

Shown here is the XML string used for communication:

```
<Module Name='%name%'>
  <BusSignalSetup>
    <BusSignals>
      <Clear />
      <BusSignal Name='My Bus 1' Polarity='Positive' DefaultBase='Decimal' Comment=''>
        <Channels>%channels%</Channels>
      </BusSignal>
    </BusSignals>
  </BusSignalSetup>
  <SamplingSetup>
    <Sampling ChannelMode='Full' Acquisition='State'
      AcquisitionDepth='%depth%' MaxSpeed='200' TriggerPosition='50' />
    <StateClockSpec Mode='Master'>
      <Clear/>
      <Master>
        <ClockGroup>
          <Edges>
            <Edge PodIndex='1' Value='Rising' />
            <Edge PodIndex='2' Value='Rising' />
          </Edges>
        </ClockGroup>
      </Master>
    </StateClockSpec>
  </SamplingSetup>
</Module>
```

Pods 1 and 2 of the logic analyzer are configured to clock the data (thus enabling the acquisition of a dual-channel ADCs output words), maximum acquisition speed is set

to 200 MHz (this being the limit of the 1692AD), and conservative triggering and data waveforms are assumed.

4.7 Filtering the Clock Close-in Phase Noise

As discussed in previous chapters, if the ADC is clocked using a variable signal generator, even a high-quality laboratory one, it is difficult to achieve the level of performance needed for adequate clocking of high-performance ADC. Broadband phase noise has the effect of decreasing SNR, and close-in phase noise reduces the overall spectral resolution by smearing the fundamental into a number of frequency bins. This smearing effect observed in the assembled system poses a problem for LabVIEW's built-in signal analysis functions - for example, measuring SFDR of the system using raw data shown in Figure 4.7(a) yields inaccurate results, because the strongest spur is observed to be in one of the neighboring bins, as shown on the zoom-in with fundamental bin filtered out, Figure 4.7(b). The proposed solution is to extract the fundamental, filter out the neighboring spurs, and then reconstruct the original signal by inserting the fundamental back, as will be shown in Figure 4.8.

1. The spectrum of the actual signal obtained from the logic analyzer is displayed first, in order for the user to be able to zoom in and examine the effect of close-in phase noise.
2. The waveform is then processed in the manner previously described, with the top branch carrying the extracted fundamental, and the bottom branch processing the residual signal with a very sharp bandstop filter centered around the frequency of the fundamental. Before filtering, the residual signal is displayed (outside of the Front Panel) for examination.

In this setup, since the analog input signal was predetermined to be 5 MHz due to

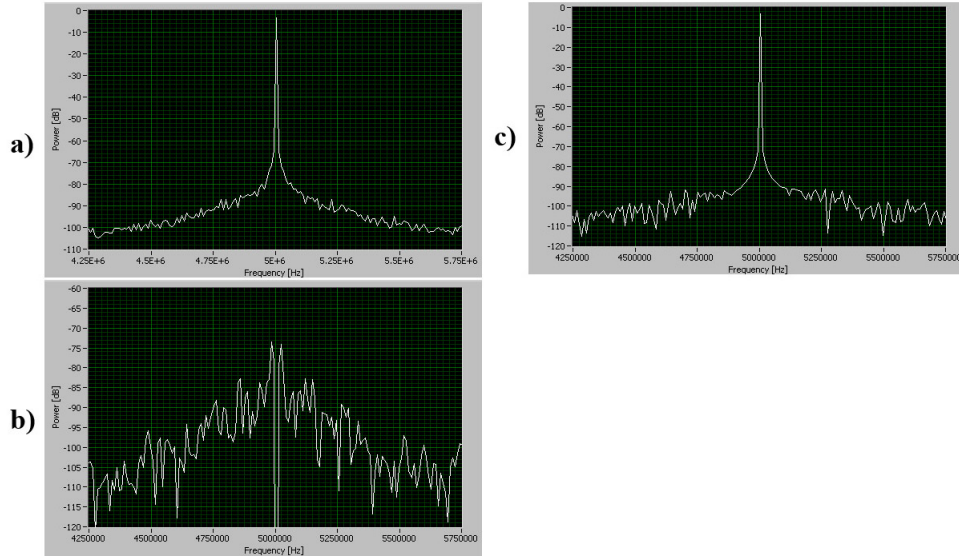


Figure 4.7 Smearing of fundamental signal due to close-in phase noise: (a) spectrum before filtering, (b) notch filter reveals spurs, and (c) spectrum after filtering and restamping fundamental

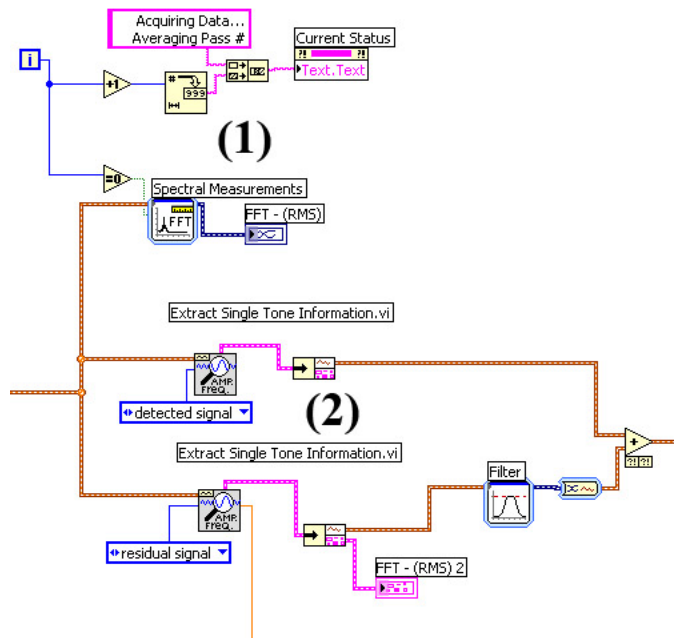


Figure 4.8 Block diagram: Compensating for clock generator's close-in phase noise

analog filter availability, the same frequency was hard-coded in the digital filter. An IIR Bessel filter of order 3 was used, with the bandstop width of 250 kHz on each side of the fundamental. Finally, the two waveforms are added together to reconstruct the measured signal, which is then passed on to the following stage. The reconstructed waveform is shown in Figure 4.7(c). The fundamental amplitude is also passed on (note that both *Extract Single Tone Information* VIs produce the same information about fundamental, even though their output waveforms are different).

4.8 Calculating Measured Parameters

The process of calculating desired dynamic parameters of the ADC under test is shown in Figure 4.9. Note that the measured data and other information on test setup are routed using the top and bottom lines to the subsequent block, which can then perform additional manipulation (e.g., saving the data to file, comparing it with previous measurements or vendor-specified reference values, etc). The VIs used are either included in standard LabVIEW 8 installation or are part of the Sound and Vibration Toolkit, also available from National Instruments.

1. *SINAD Analyzer* VI directly calculates SINAD and THD+noise, and provides info on the fundamental frequency detected (to easily verify whether the expected spur was recognized as the fundamental). SINAD is also used to calculate ENOB [5] using the well-known relationship (4.1)

$$ENOB = \frac{SINAD - 1.76dB}{6.02dB} \quad (4.1)$$

2. *Harmonic Analyzer* extracts the level of DC, fundamental, and its harmonics. It also directly calculates THD, which is then used with SINAD to determine the value of SNR [14] using the relationship (4.2)

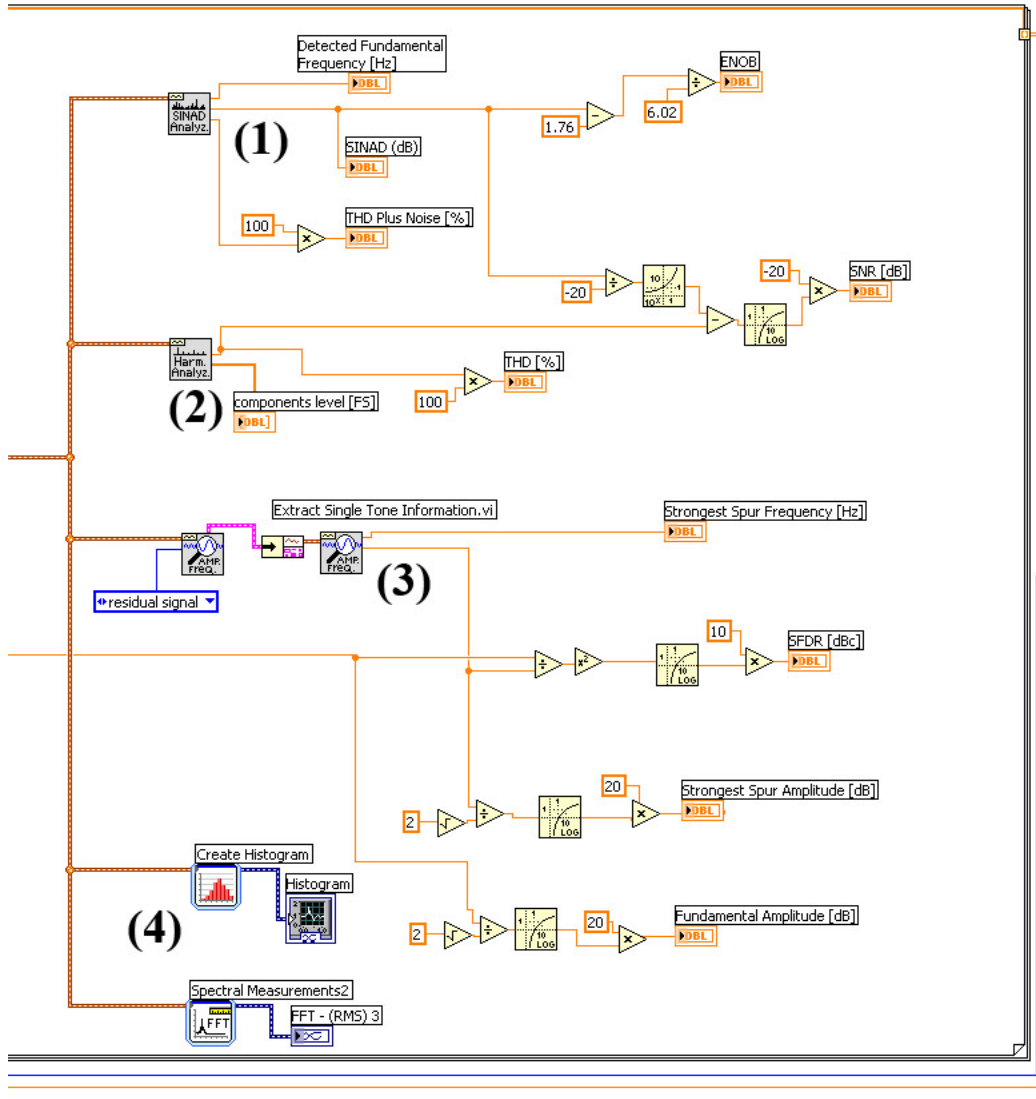


Figure 4.9 Block diagram: Calculating ADC dynamic parameters

$$SNR = -10 \log_{10}(10^{-SINAD/10} - 10^{-THD/10}) \quad (4.2)$$

3. Since the close-in phase noise filtering block was optional, we again extract the residual signal in order to find the strongest spur frequency and amplitude. Now SFDR can be calculated as the difference in amplitudes of the fundamental and the strongest spur.

4. At this point, another histogram can be created as a starting point for static parameter measurements, but also as a visual and numerical indicator of how the close-in phase noise filtering block influenced the acquired samples.

CHAPTER 5

MEASUREMENT RESULTS

5.1 Overview

The designed setup and methodology were tested by measuring two commercial A/D converters placed on vendor-designed evaluation boards. Due to the different format of some parameters (dB or dBc versus percentile values), an online conversion/calculation tool [15] was used to convert them into the format specified by IEEE Std 1241.

5.2 Testing of Analog Devices AD9841

The first ADC under test was the Analog Devices AD9841, a dual 8-bit data converter operating from a single 3.3 V power supply, which was clocked at 75 MHz and supplied with an input sinewave of 5 MHz (both values adjusted using coherent sampling algorithm), with the amplitude of 6 dBm (with full-scale being $A_{fs} = 1 \text{ V}_{pp}$, and also taking into account the bandpass filter attenuation of around 3 dB). $M = 8192$ samples were taken, and the spectrum of sampled signal is shown in Figure 5.1.

The measured results are shown in Table 5.1 and are contrasted to the expected data from the ADC data sheet. Note that the data sheet did not provide numerical values for the given input conditions, so they had to be extrapolated taking into account the smooth and slow-changing variation of parameters over the entire Nyquist frequency band. A

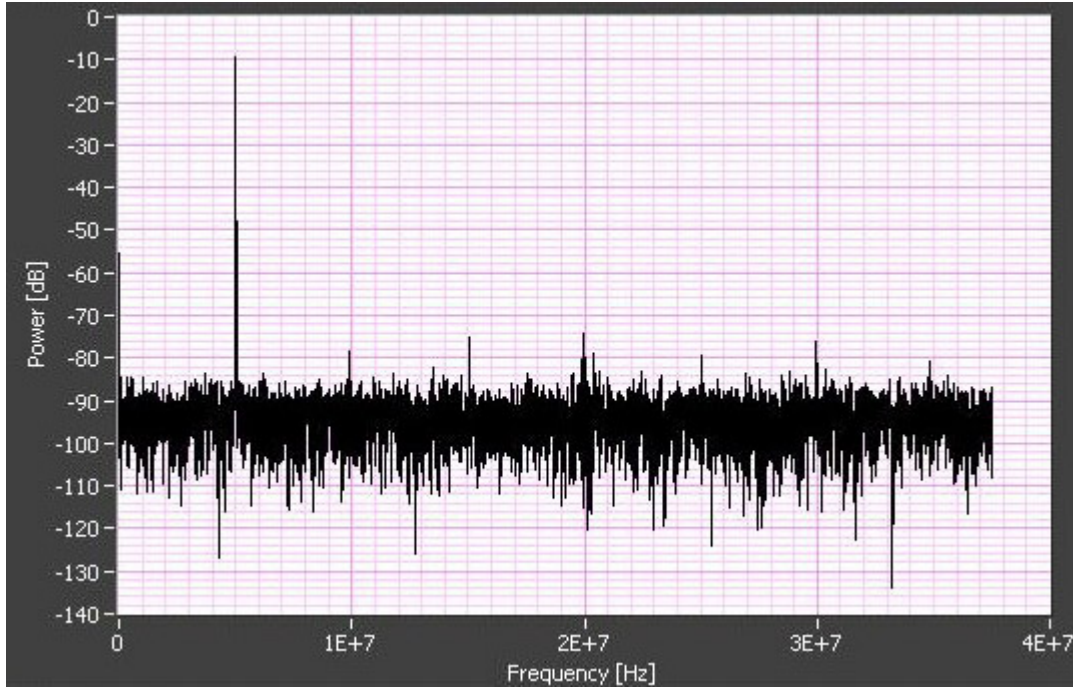


Figure 5.1 Spectrum of sampled signal for AD9841 ADC

good agreement between two sets of data can be seen, confirming the validity of the measurement setup.

Table 5.1 Comparison of expected and measured data for AD9481 ADC

AD9481	Datasheet	Measured
SNR [dB]	48.2	47.99
SFDR [dBc]	64.8	64.72
SINAD [dBc]	47.1	46.56
THD [dBc]	53.6	52.08
ENOB [bits]	7.53	7.44

5.3 Testing of Analog Devices AD9245

The second ADC under test was the Analog Devices AD9245, a single-channel 14-bit 3 V data converter. The physical layout of the ADC placed on its evaluation board is shown in Figure 3.3.

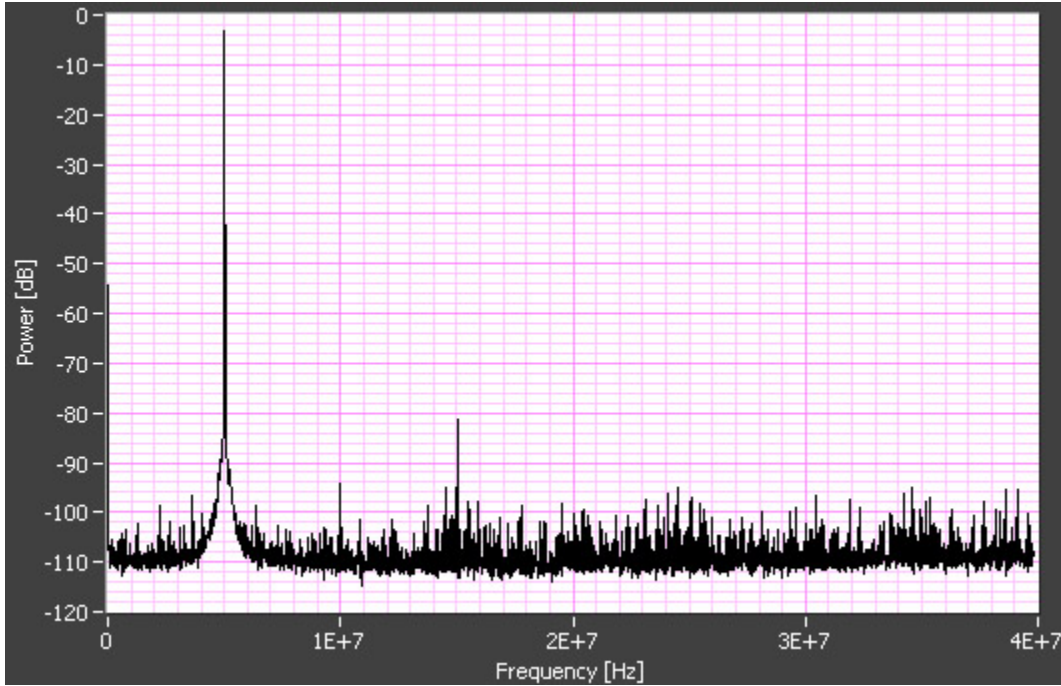


Figure 5.2 Spectrum of sampled signal for AD9245 ADC

The AD9245 was clocked at its maximum frequency of 80 MHz and supplied with an input sinewave of 5 MHz (both values adjusted using coherent sampling algorithm), with the amplitude of 16.6 dBm (with full-scale being $A_{fs} = 2 V_{pp}$, and also taking into account the bandpass filter attenuation of around 3 dB). $M = 8192$ samples were taken, and the spectrum of sampled signal is shown in Figure 5.2. For purposes of making spurs and harmonics more visually pronounced, averaging over 20 measurements was used. The vendor-specified spectrum is shown in Figure 5.3; note that this figure is obtained using ADC spectrum simulation tool, since the data sheet did not provide the spectrum plot for $f_{in}=5$ MHz, and is provided for illustration.

The measured results are summarized in Table 5.2 and are contrasted to the vendor-specified data. The measured SNR is reduced from the expected value due to clock generator's broadband phase noise; however, SFDR (and therefore SINAD) and THD have significantly lower values than expected. The reason for this disagreement most probably arises from a pronounced third harmonic of the fundamental, due to the analog

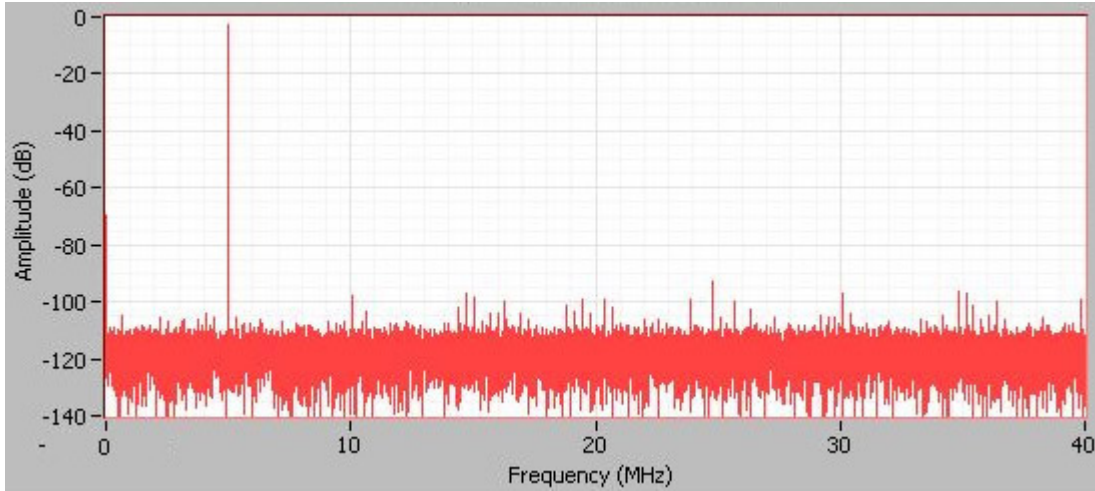


Figure 5.3 Expected signal spectrum for AD9245 ADC

input signal of 2 V_{pp}, which introduces a high level of distortion of the signal generated by the E4433. An additional band-pass or low-pass filter would be needed in the analog input signal chain.

Table 5.2 Comparison of expected and measured data for AD9245 ADC

AD9245	Datasheet	Measured
SNR [dB]	73.1	71.89
SFDR [dBc]	90.5	78.61
SINAD [dBc]	73	68.32
THD [dBc]	89.4	70.83
ENOB [bits]	11.8	11.07

CHAPTER 6

CONCLUSIONS

In summary, a functional analog/digital converter dynamic test setup was designed and implemented. Algorithmic foundations for the measurements were discussed. The details of the equipment needed for the setup were presented, and the accompanying automation and measurement program was explored in detail. Finally, measurement results of two commercial ADCs were presented in order to verify the validity of the setup and the program itself.

Due to the complex nature of ADCs themselves, a large number of performance metrics and tests to measure them can be designed, and no single program could be written to cover all possible combinations. However, thanks to the inherently modular structure of LabVIEW programming language, modifying and expanding existing code to add new functionality proves simple and straightforward. The demonstrated system also utilizes two different communication interfaces (GPIB and Firewire) and can easily be expanded to accommodate any other. Therefore, the program presented here, currently customized towards measuring high-performance ADCs for wireless systems, can also serve as a solid foundation on which to build various other automated test setups, depending on the needs of a specific measurement application.

REFERENCES

- [1] Texas Instruments, “ADS5463, 12-bit, 500 MSPS analog-to-digital converter with buffered input data sheet,” 2006, <http://www.ti.com>.
- [2] Linear Technology, “LTC2208 - 16-Bit, 130 Msps ADC data sheet,” 2005, <http://www.linear.com>.
- [3] IEEE-Std-1241-2000, Standard for terminology and test methods for analog-to-digital converters, June 13, 2001.
- [4] S. J. Tilden et al., “Overview of IEEE-STD-1241, standard for terminology and test methods for analog-to-digital converters,” in *Proc. of Instr. Meas. Techn. Conf.*, 1999, pp. 1498-1503.
- [5] W. Kester, *The Data Conversion Handbook*. Burlington, MA: Newnes, 2005.
- [6] Dallas-Maxim, Appl. Note 728, 2001, <http://www.maxim-ic.com>.
- [7] Analog Devices, *Practical Analog Design Techniques*. Norwood, MA: Analog Devices, Inc., 1995, pp. 4.12-15.
- [8] A. Boni et al., “Issues in the design of a test set-up for high-speed A/D converters,” *Measurement*, vol. 28, no. 2, pp. 105-114, Sept. 2000.
- [9] Dallas-Maxim, Appl. Note 3557, 2005, <http://www.maxim-ic.com>.
- [10] B. Brannon, Analog Devices, Appl. Note AN-501, 2006, <http://www.analog.com>.
- [11] Agilent Technologies, *Agilent PSA Series Spectrum Analyzers Phase Noise Measurement Personality, Technical Overview with Self-Guided Demonstration, Option 226*, 2005, <http://www.agilent.com>.
- [12] Raltron Electronics Corporation, “Convert SSB phase noise to jitter,” 2007, <http://www.raltron.com>.
- [13] Agilent Technologies IntuiLink 16700 Software, 2003, <http://www.agilent.com>.
- [14] W. Kester, Analog Devices, Appl. Note MT-003, 2005, <http://www.analog.com>.

- [15] Analog Devices, "Interactive design tools - SNR / THD / SINAD calculator," 2007, <http://www.analog.com>.