

POLITECNICO DI TORINO

Master's degree in Electronic Engineering

Thesis

**ADC noise measurement  
on FPGA board**



**Supervisor**

Giovanni Antonio COSTANZO

**Candidate**

Luca FONTICELLI

ACADEMIC YEAR 2018-2019

## **Acknowledgement**

I'm extremely grateful to Professor Giovanni Costanzo for the continuous support and guidance during this project development.

I'd like also to thank Dr. Cardenas Olaya and Dr. Calosso for their help and support through sources research, and all their colleagues at INRiM for their precious advice and sympathy.

At the end, I'd like to deeply thank my family and my friends who have always supported for all these long and challenging years.

## Abstract

Improvements in electronic have allowed front-end part in communication systems to be faster and more precise: the focus on this part of a device is justified as the communication with the external world has to keep the rhythm with the high speed calculations of every CPU. Every application requires then a reliable and low noise-conditioned devices able to reach performances we use everyday. New techniques are developing also in time - frequency metrology field, where the low noise rate is required to achieve high measurement accuracy.

The use of new SoC commercial boards, composed by a programmable logic and processing system, is a matter of interest in this field in order to develop new measurement flows: this kind of applications are reliable in terms of flexibility, where on-the-fly test and re-programming hardware are required, and quite fast in processing data thanks to last years' CPU presence. Implementing board-style techniques could allow researchers to avoid complex instruments tune and long measurement chain in favour of a deep versatility. However, this kind of adaptability may be limited by main blocks inside every commercial board: ADC and DAC for front-end, clock generation (i.e. PLL), distribution and interconnections for back-end.

In this context, my thesis work is focused on analysing and selecting the main noise figures of merit of the Analog to Digital converter, principal block of the front-end, selecting testing techniques and finding a way to implement them using this board potential. Two hardware projects have been designed and implemented using VHDL language and loaded on FPGA part of the System on a Chip. Data have been generated with Agilent instruments, directly processed and converted in the Programmable Logic part and packetized and sent by the Processing System to the main computer: here data have been processed and analysed using MATLAB and many figures and values have been computed and here presented and commented, to understand which could be the frequency performances limiting factor and whether these boundaries could be corrected or not. The figures of merit here presented can be used to analyse other front end part, proposing a complete overview of commercial boards' limiting factor. At the end some improvements are speculated and presented for further developments: these techniques could also be modified to make the board analyse them directly, and other modifications could be designed to increase test flow speed. This project is a part of a bigger and more complex Red Pitaya characterisation carried on at INRiM by Dr. Cardenas and Dr. Calosso, to develop a deep knowledge of this architecture: this can lead to an implementation of this kind of boards in high-specialised techniques, as fibre dissemination and reception of signals to synchronise different research laboratories.

# Contents

List of figures	iv
List of tables	vi
<b>1 Introduction</b>	<b>1</b>
1.1 The importance of the front-end in communication systems . . .	1
1.2 Thesis outline . . . . .	2
<b>2 Theoretical background</b>	<b>5</b>
2.1 The concept of noise in electronics . . . . .	5
2.2 Analog to digital conversion . . . . .	8
2.3 ADC figures of merit . . . . .	11
2.3.1 Static parameters . . . . .	11
2.3.2 Dynamic parameters . . . . .	13
<b>3 Testing an ADC</b>	<b>17</b>
3.1 STEMLab Red Pitaya . . . . .	17
3.1.1 Linear Technology 2145 . . . . .	19
3.2 Measurement techniques . . . . .	21
3.2.1 Histogram test for linearity . . . . .	21
3.2.2 Histogram test for analog noise . . . . .	23
3.2.3 Histogram test for jitter evaluation . . . . .	23
3.2.4 THD evaluation . . . . .	25
3.2.5 SINAD evaluation in frequency domain . . . . .	25
3.3 Hardware implementation . . . . .	26
3.3.1 ADC data converter . . . . .	26
3.3.2 Timing and ST diagrams . . . . .	28
3.3.3 C code development . . . . .	29
3.3.4 On board clock analyser . . . . .	29

## CONTENTS

---

<b>4</b>	<b>Results</b>	<b>35</b>
4.1	DNL and INL evaluation . . . . .	35
4.1.1	Board defect . . . . .	35
4.2	Bathtub distributions . . . . .	38
4.3	DNL results . . . . .	42
4.4	INL results . . . . .	46
4.5	Maximum DNL - INL . . . . .	50
4.6	THD evaluation . . . . .	51
4.7	Analog noise evaluation . . . . .	54
4.8	Jitter evaluation . . . . .	55
4.9	Final SINAD and ENOB evaluations . . . . .	60
<b>5</b>	<b>Conclusions and future developments</b>	<b>64</b>

# List of figures

1.1	Input front-end schematic . . . . .	1
2.1	Example of noise amplitude and related probability density. . .	5
2.2	Resulting noise bandwidth. . . . .	6
2.3	Effect of noise on a sine wave, evaluated with SNR. . . . .	7
2.4	Example of sampled and quantized signal to digital domain. . .	8
2.5	Maximum quantization error referred to every symbol. . . . .	9
2.6	Sampling in an analog to digital system. . . . .	10
2.7	Aliasing effect arising if the input signal frequency $F_m$ does not respect Nyquist criterion. . . . .	11
2.8	Offset and gain error in ADC transfer function. . . . .	12
2.9	DNL and INL effect on ADC transfer function. . . . .	13
2.10	Input - referred noise effect on ADC transfer function. . . . .	13
2.11	Representation of THD in respect to the fundamental. . . . .	14
2.12	Sample jitter representation. . . . .	15
3.1	Zynq platform overview. . . . .	18
3.2	Red Pitaya hardware overview. . . . .	19
3.3	LT2145 core block diagram. . . . .	21
3.4	Example of noise referred output bins. . . . .	23
3.5	Example of 0 V crossing sampling to jitter evaluation. . . . .	24
3.6	SINAD evaluation in frequency domain. . . . .	26
3.7	Scheme of the first project implemented. . . . .	31
3.8	Reset state transition. . . . .	32
3.9	Write full transition. . . . .	32
3.10	Read empty transition. . . . .	32
3.11	State transition diagram. . . . .	33
3.12	Scheme of the second project implemented. . . . .	34
4.1	First evaluation of histogram test. . . . .	36
4.2	First evaluation of DNL. . . . .	37
4.3	Histogram test, 1 MHz . . . . .	38

## LIST OF FIGURES

---

4.4	Histogram test, 5 MHz . . . . .	39
4.5	Histogram test, 10 MHz . . . . .	40
4.6	Histogram test, 20 MHz . . . . .	41
4.7	DNL evaluation, 1 MHz . . . . .	42
4.8	DNL evaluation, 5 MHz . . . . .	43
4.9	DNL evaluation, 10 MHz . . . . .	44
4.10	DNL evaluation, 20 MHz . . . . .	45
4.11	INL evaluation, 1 MHz . . . . .	46
4.12	INL evaluation, 5 MHz . . . . .	47
4.13	INL evaluation, 10 MHz . . . . .	48
4.14	INL evaluation, 20 MHz . . . . .	49
4.15	THD evaluation, 1 MHz . . . . .	51
4.16	THD evaluation, 5 MHz . . . . .	52
4.17	THD evaluation, 10 MHz . . . . .	52
4.18	THD evaluation, 20 MHz . . . . .	53
4.19	THD values over frequency. . . . .	53
4.20	Analog noise evaluation, input grounded. . . . .	54
4.21	Overall jitter evaluation, 1 MHz . . . . .	55
4.22	Overall jitter evaluation, 5 MHz . . . . .	56
4.23	Overall jitter evaluation, 10 MHz . . . . .	57
4.24	Overall jitter evaluation, 20 MHz . . . . .	58
4.25	Overall SINAD evaluation. . . . .	60
4.26	ENOB computed from the SINAD values. . . . .	61
4.27	SINAD evaluation in time and frequency domain. . . . .	62

# List of tables

3.1	Figures of merit from Linear Technology datasheet. . . . .	20
4.1	Maximum DNL values in respect to the frequency. . . . .	50
4.2	Maximum INL values in respect to the frequency. . . . .	50
4.3	Results of the first jitter analysis through the first hardware equipment. . . . .	59
4.4	Results of the second jitter analysis through the second hardware and the frequency meter. . . . .	59
4.5	Results of SINAD evaluation with ENOB values. . . . .	63
4.6	Results of the second SINAD analysis in time and frequency domain. . . . .	63

# Chapter 1

## Introduction

### 1.1 The importance of the front-end in communication systems

In recent years the demand for high speed communication systems has grown, reaching high performances that we use everyday: examples of this exponential need are commercial (e.g. WiFi b/g/n band, LTE) and research (e.g. optical fibre link). In these regards, especially for research purposes as in this thesis, it is important to have a front-end suitable for the application: it has to sample, buffer and store the information quickly and correctly enough to respect the requirements (Figure 1.1).

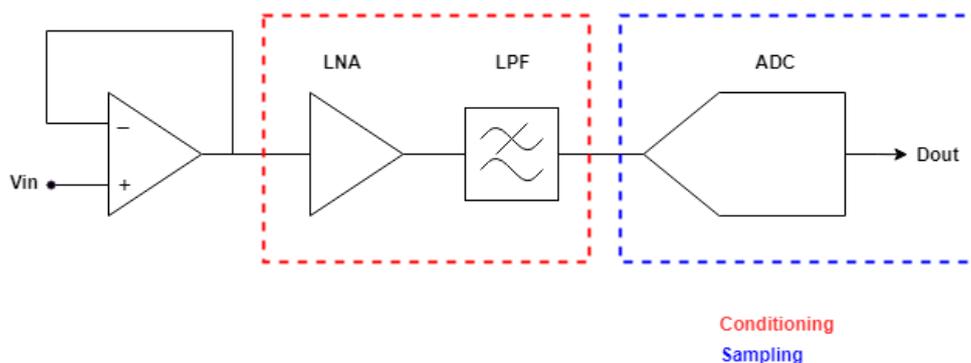


Figure 1.1: Input front-end schematic

## 1.2 Thesis outline

---

The main stage to be considered is the Analog to Digital converter: in fast data acquisition systems, when both speed and accuracy are required, there is a need of a fast and precise ADC; the typical drawbacks are high power consumption and large on-chip area occupied.

When precise electronic is required as in time - frequency metrology, analog approach is often used to test components. Inconveniences of this approach are the low reliability due to the need of tune the instrumentation and difficult repeatability: in order to test a converter for example, many devices are needed and an automatic control is often difficult. Then, the transition between analog and digital equipment has allowed lighter and faster measurement techniques to be developed and used. Last years' FPGA (Field Programmable Gate Array) boards have been used as the on-the-fly hardware reprogram capability mixed with next-generation ARM architecture have allowed researchers to be faster and ductile: data can be processed and filtered by the PL (programmable logic) and then analysed with PS (processing system) on the same board. These systems can guarantee high specifications: FPGA with million of gates, DSPs, multi channel ADCs and DACs for I/O; PS with GHz frequency CPU often ARM technology and many processing and communication modules (e.g. GPIO, UART, I2C).

When using these devices however, the limitation can be the amount of noise injected by the board in the processed signal. Then, when fast sampling rate (input) or high signal stability (output) are required this situation can be seen as a performance boundary. Every board can be evaluated by referred internal noise:

- PSU has to be stable to avoid moving thresholds across the circuit;
- circuitry of PS has to be electrically isolated from PL and vice versa;
- components through the information chain has to guarantee an high Signal to Noise Ratio;
- clock generation and distribution has to be compensated and low-jitter affected.

## 1.2 Thesis outline

The objective of this thesis is to characterise principal figures of merit of the main front-end input stage, the AD converter, exploiting theoretical ways to measure them and verify those techniques on a commercial board. A limiting factor in ADC performances is evaluated, summing up all those contributions

## 1.2 Thesis outline

---

in a valuable element in the time domain and test it also in the frequency domain. The thesis is organised as follow:

- in Chapter 2 theoretical aspects of noise and ADC figures of merit are analysed;
- in Chapter 3 measurement techniques to test the converter are shown, with a brief introduction to the board used to test those procedures and an overview on hardware projects implemented;
- in Chapter 4 results of techniques implementation are presented, analysed and commented.
- in Chapter 5 then conclusions and suggestions for future development are presented.

## 1.2 Thesis outline

---

# Chapter 2

## Theoretical background

### 2.1 The concept of noise in electronics

Broadly speaking, the noise is defined as a disturbance overlying on the signal to be processed, limiting the maximum amount of information carried. Practically it is a fluctuation following a Gaussian distribution with mean equal to zero over an infinite bandwidth (*white noise*, figure 2.1).

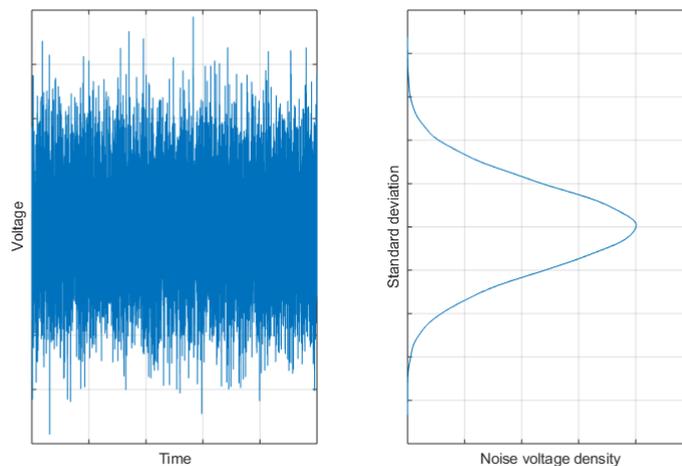


Figure 2.1: Example of noise amplitude and related probability density.

Noise sources in electronics can be summed up in three main categories[1]:

- *electric thermal noise* or *Johnson noise*, related to the casual motion (*Brownian*) of carriers in a conductor, always present. As verified

## 2.1 The concept of noise in electronics

---

by Johnson a resistor in open circuit condition presents a voltage at its terminals, whose effective amplitude follows the formula:

$$V_{eff}^2 = 4kTR\Delta f \quad (2.1)$$

depending on the observation frequency, with  $k$  equal to the Boltzmann constant;

- **shot noise** or **Schottky**, related to the carrier going through a potential barrier as in p-n junction, resulting in a random impulsive noise;
- **flicker noise** or **1/f**, whose amplitude is related to the inverse of the frequency (that is the reason of the name **pink noise**).

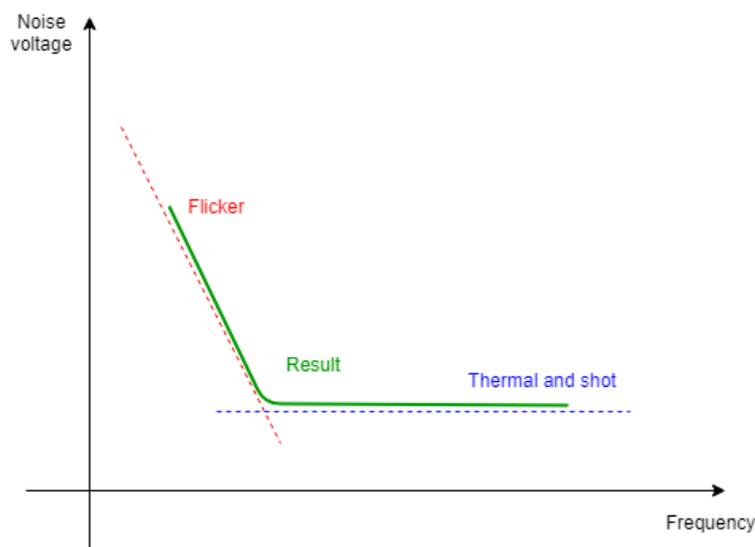


Figure 2.2: Resulting noise bandwidth.

All these contributions are collected under the effect called **additive noise**, resulting in unwanted phase and amplitude modulation on the signal.

The phase noise is a random fluctuation of the phase of the input signal, and it can be represented as a time fluctuation (jitter) following the formula[2]:

$$\chi(t) = \frac{\phi(t)}{2\pi\nu_0} \quad (2.2)$$

with  $\phi(t)$  representing the phase fluctuation and  $\nu_0$  the input frequency. It is evident that the transposition from  $\phi$  to  $\chi$  noise type is independent from frequency: it can be seen as a jitter on the clock distribution affecting

## 2.1 The concept of noise in electronics

---

the signal. Eventually, the visible result on the signal is another spurious amplitude modulation summing to the other one: every time-based change is connected to an instantaneous variation of the signal value, following

$$\delta v = A2\pi f\delta t \quad (2.3)$$

The sum of all these contributions lead to a time-independent modulation worsening the ratio between signal power and noise power (SNR, exploited in section 2.3). This effect can be injected and/or worsen through the typical signal chain: the information can be bad conditioned (at the input), poorly sampled and saved (ADC stage), wrongly analysed and reconstructed (PS/-DAC stage).

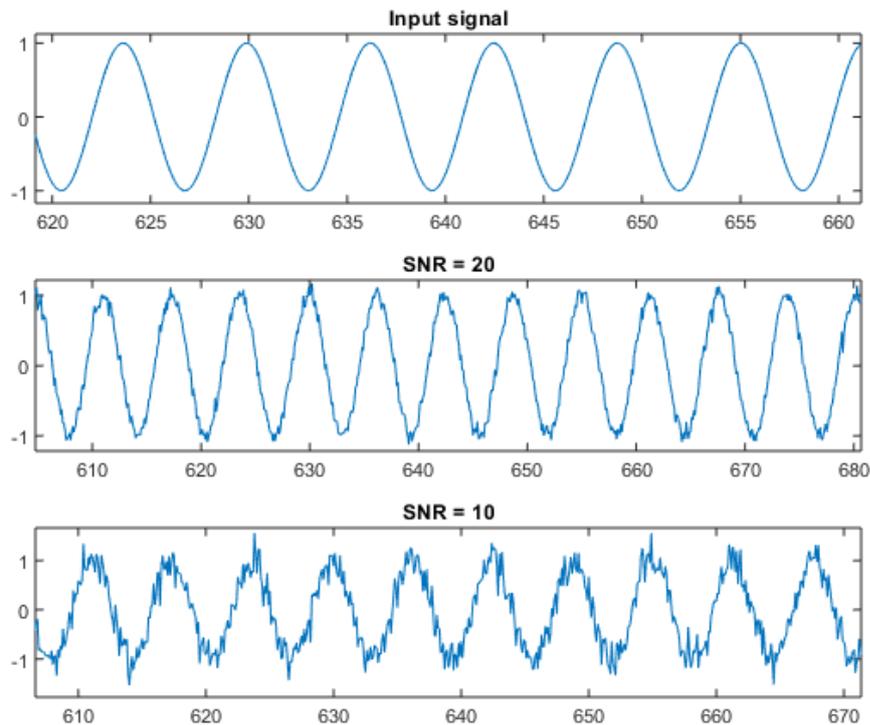


Figure 2.3: Effect of noise on a sine wave, evaluated with SNR.

## 2.2 Analog to digital conversion

AD converters are the fundamental block in every circuit front-end: from continuous amplitude and time-varying signal the information has to be made discrete in time domain (sampling) and in amplitude (quantization). A voltage based signal is then mapped on a sequence of bits (symbols). Every converter has different specifications, as number of bits, sampling period, settling time: the choice has to be done considering requirements but also drawbacks, as power consumed, supply stability needed, on-chip surface occupied.

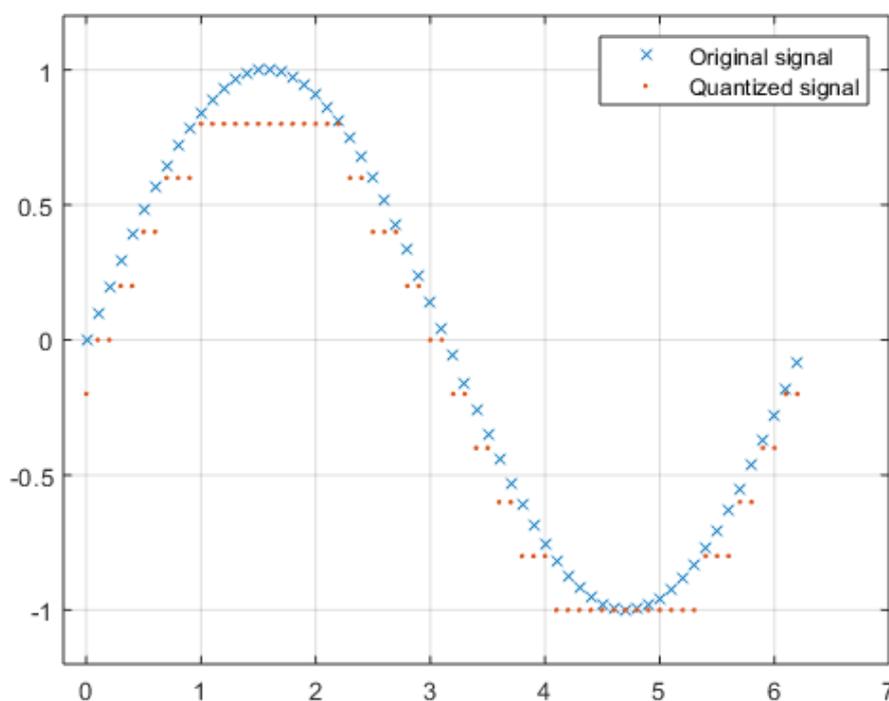


Figure 2.4: Example of sampled and quantized signal to digital domain.

The general equation followed by ADC is:

$$V_{in} = V_{FS} \cdot \sum_{i=1}^N \frac{D_i}{2^i} + \epsilon \quad (2.4)$$

where

- $D_i$  is the ADC output code;

## 2.2 Analog to digital conversion

---

- $V_{FS}$  is the full-scale analog input value;
- $\epsilon$  is the offset given by the quantization error.

Ideally, an AD system continuously encodes an analog input voltage into  $N$ -bit words respecting the (2.4). At first in order to study every converters performance two figures of merit have to be taken into account, **resolution** and **accuracy**: the former refers to the theoretical accuracy achievable by the device; the latter describes how close the device is to its theoretical resolution [3]. Accuracy is basically how much the converter is limited by noise sources. Each of the converted codes correspond to a range of voltages at the input, called the **code width**. Logically, higher is the resolution, smaller will be input voltage range associated to every code.

Since every conversion system is not ideal, every input is associated with the nearest **code centre**. Here it is important the converter resolution, because the range of possible symbols is divided in  $2^N$  intervals: the distance from every voltage value and its symbol is related to a certain amount of noise on the output. The maximum quantization error is then:

$$|\epsilon_q| \leq \frac{A_D}{2} = \frac{S}{2^{N+1}} \quad (2.5)$$

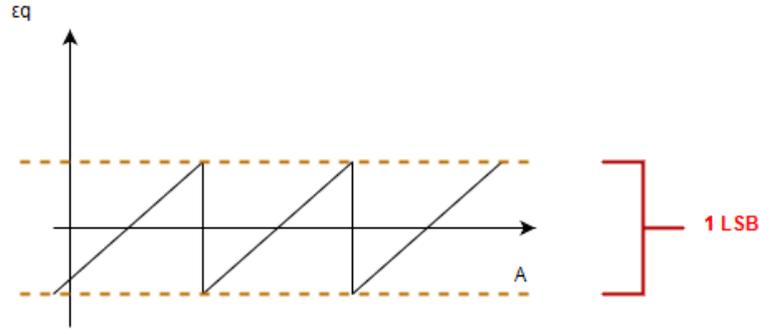


Figure 2.5: Maximum quantization error referred to every symbol.

where  $A_D$  is the minimum quantization step and  $S$  the symbols range: the amount of average quantization error affects directly the conversion formula, effect evaluated with  $\epsilon$  term.

Eventually, in figure 2.6 can be understood the sampling effect on the signal: correlation between the input and a Dirac comb is applied, following the equation:

$$X_s = x(t) \cdot \sum_{n=-\infty}^{\infty} \delta \cdot (t - nT_s) \quad (2.6)$$

## 2.2 Analog to digital conversion

---

where

- $x(t)$  is the input signal;
- $\delta$  is the Dirac comb applied every  $T_s$ ;
- $T_s$  is the sampling period, equal to the reciprocal of the sampling frequency  $F_s$ .

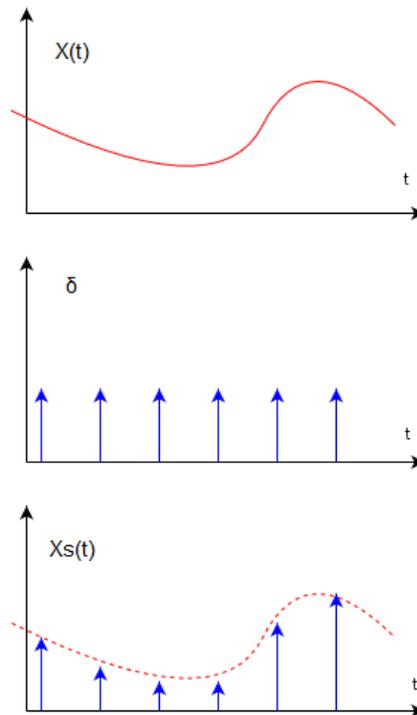


Figure 2.6: Sampling in an analog to digital system.

The actual limitation of AD system can be then appreciated: sampling is an operation that creates infinite replica of input signal in the frequency domain. The input signal then has to respect the *Nyquist criterion*: in order not to lose information, a signal must be sampled *twice the signal bandwidth*, otherwise aliasing may occur (permanent information loss, reference figure 2.7). At the end, talking about bandwidth, another limitation is given by sample and hold need of converters: it is a circuit, often a capacitor with a switch (MOSFET), that retains the information for the amount of time sufficient to the circuitry above to process the information. Holding signal value for an amount of time worsens bandwidth capability of the converter itself, as the hold time has to be sum to the sampling period to obtain the maximum sampling frequency in the worst case.

## 2.3 ADC figures of merit

---

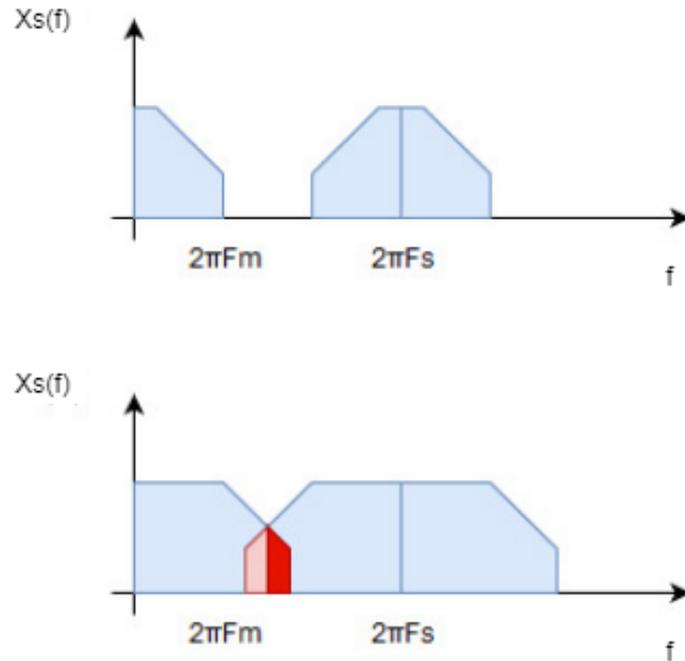


Figure 2.7: Aliasing effect arising if the input signal frequency  $F_m$  does not respect Nyquist criterion.

## 2.3 ADC figures of merit

Main figures of merit of the AD converter are now analysed. They've been divided in two categories, underlining the different converter behaviour (frequency dependence or not).

### 2.3.1 Static parameters

Static parameters are the specifications that can be tested at different speed or with constant voltage, defining the ADC behaviour in a statistical way [4] and are related to its transfer function.

- the **dynamic range** is the ratio between the smallest output value and the full-scale one;

$$DR = 20 \cdot \log_{10}\left(\frac{2^N}{1}\right) \approx 6N \quad (2.7)$$

- the **offset error** is the difference between ADC ideal behaviour and actual one: the first transition for example should be at  $\frac{1}{2}$  LSB, where

## 2.3 ADC figures of merit

offset effect makes it to shift towards ideal value. The **gain error** is the different slope of the real ADC from the ideal one, computed as  $\frac{2^N}{V_{FS}}$ . Gain and offset uncertainty can be corrected by converter calibration (reference figure 2.8);

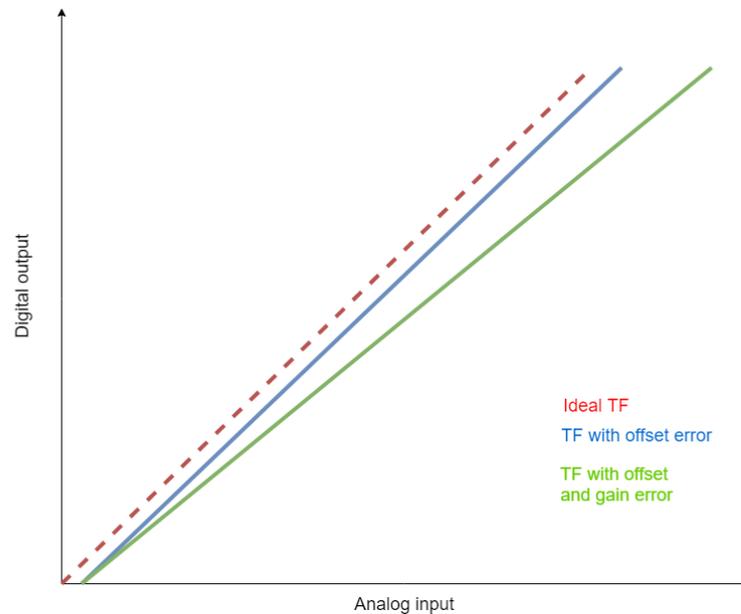


Figure 2.8: Offset and gain error in ADC transfer function.

- the **DNL** (differential non-linearity) is the difference between ideal code width and actual one; all code widths of an ideal AD converter are 1 LSB. The **INL** (integral non-linearity) is otherwise defined as the distance between the code centre lying on the ideal AD transfer function and the non ideal one (reference 2.9). Those error definition are bond: INL value can be defined as the sum of all previous DNL values up to a local point. Defining those distances is important because it allows the evaluation of a problem affecting most converters, the "missing codes" error: if locally DNL value overcome  $\pm 1$  LSB, the code nearby the affected one is basically suppressed, not allowing it to be saved and reconstructed; when the signal is rebuilt in the analog domain that missing code won't allow the information to be correctly evaluated, giving a random voltage value and subsequently invalidating the information, as a noise source.
- the **analog noise**  $V_n$  (also known as input-referred noise) [5] is the noise generated in the transition between one code and another conditioned by the board circuitry: a certain amount of noise is produced manly due to

## 2.3 ADC figures of merit

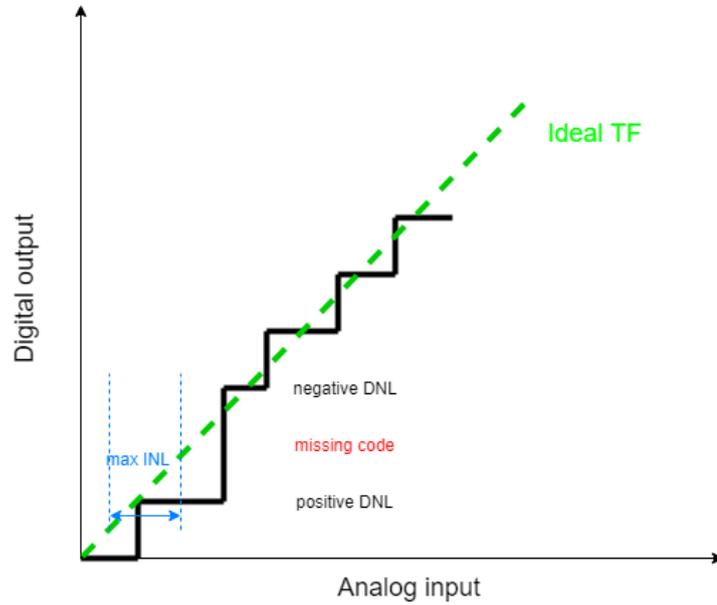


Figure 2.9: DNL and INL effect on ADC transfer function.

thermodynamic one, therefore the transition widths change dynamically, producing a certain amount of noise on sampled signal that sums on DNL - INL effect (reference figure 2.10).

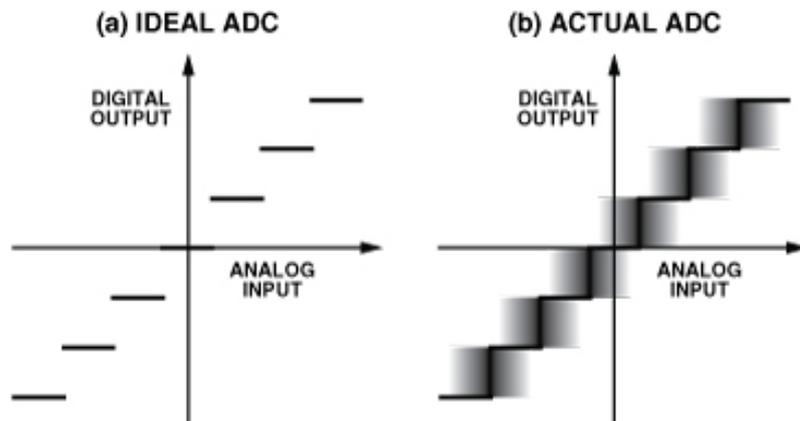


Figure 2.10: Input-referred noise effect on ADC transfer function. [5]

### 2.3.2 Dynamic parameters

Dynamic parameters define frequency performances of the converter, evaluating the distortion induced on input signal [4]. These values are related to the

## 2.3 ADC figures of merit

---

information sampled, and are calculated through a spectrum analysis.

- the **Total Harmonic Distortion** (THD) is the ratio between sum of all higher harmonics in the output signal's Fast Fourier Transform spectrum divided by the fundamental one. All harmonics are included in the definition, but just up to the third - fourth component are considered (those with not negligible power). In formula:

$$THD = 10 \cdot \log\left(\frac{V_2^2 + V_3^2 + \dots + V_n^2}{V_1^2}\right) \quad (2.8)$$

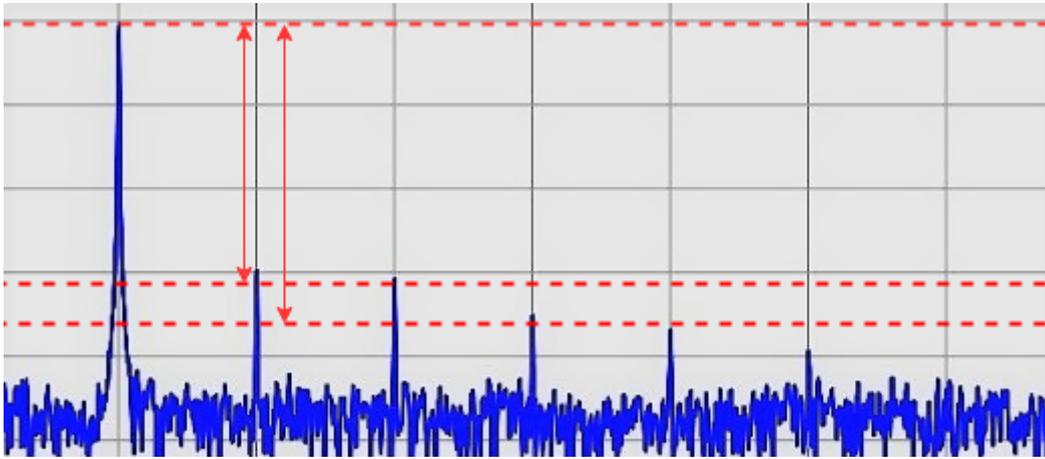


Figure 2.11: Representation of THD in respect to the fundamental (on the left).

- the **sample jitter** is the variation of the period of two consecutive samples, related to the input signal slew rate and the converter clock error. The internal source of jitter noise is mainly related to the Sample and Hold aperture time variation, called *aperture jitter*, and the main board clock.
- the **SNR** value evaluates the ratio of the signal amplitude in respect of the noise level; it gives an idea of how much the noise level contribution, sum of all previous figures of merit, influences the signal. Logically, higher is the SNR, higher will be the converter reliability in frequency terms.

An expansion of the previous concept, aim of this thesis, is the so called **SINAD** [6]: it evaluates the SNR plus the noise and the distortion effect due to the converter. Often calculated as *maximum SINAD*, it is

## 2.3 ADC figures of merit

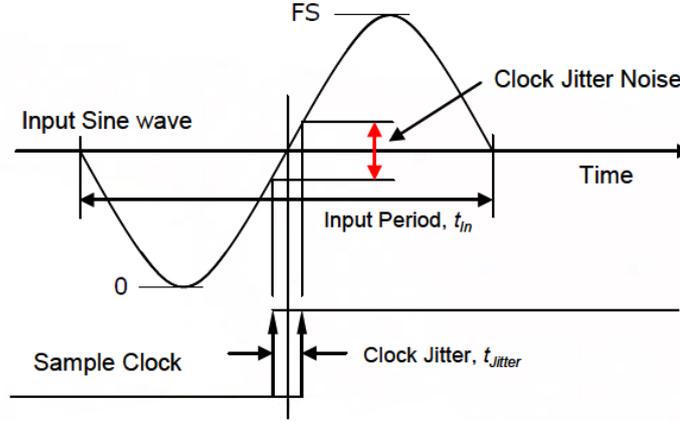


Figure 2.12: Sample jitter representation. [6]

the parameter that tells how much the signal is eventually condition by the overall noise sources through the conversion chain. In formula:

$$SINAD = -20 \log \sqrt{\frac{2}{3} \left( \frac{\sqrt{\frac{BW}{100}} (1 + DNL)}{2^N} \right)^2 + \left( 2\pi \frac{T_j}{10^6} \right)^2 + \left( \frac{2\sqrt{2}V_n}{2^N} \right)^2 + \left( \frac{THD\%}{100} \right)^2} \quad (2.9)$$

where

- $BW$  is the fraction of the Nyquist bandwidth used;
- $N$  is the resolution in bits;
- $DNL$  is the differential non linearity, in LSB;
- $V_n$  is the analog noise evaluated in LSB;
- $THD$  is the total harmonic distortion, in percentage;
- $T_j$  is the RMS jitter of the signal normalised to its period, in PPM.

In ideal conditions (i.e.  $BW = 100\%$ ,  $DNL = 0$ ), SINAD becomes equal to ideal SNR value, computed with the rule-of-thumb equation

$$SNR = 6.02N + 1.76 \quad (2.10)$$

### 2.3 ADC figures of merit

---

- eventually the ***ENOB*** is a parameter computed starting from the previous one that analyse the converter effective resolution taking into account noise and distortion levels. In formula:

$$ENOB = \frac{SINAD - 1.76}{6.02} \quad (2.11)$$

# Chapter 3

## Testing an ADC

In this chapter the Red Pitaya board system will be introduced, and several measurement techniques will be discussed. Then, the hardware implementations of those techniques will be shown and discussed.

### 3.1 STEMLab Red Pitaya

Red Pitaya is an open source embedded platform composed by a PL (programmable logic) by Xilinx (Artix-7) and a PS (processing system) part, the Zynq platform (ARM instruction set). The good aspects of the board are the embedded instruments on board, loaded by the custom Linux operating system, and the out-of-the-box equipment provides:

- oscilloscope function and signal generator, two input-output channels with external trigger capability, able to generate and analyse signals up to 50 MHz;
- a DFT spectrum analyser able to measure signals up to 62.5 MHz and export data;
- a Bode analyser able to measure gain and phase from 1 Hz to 60 MHz;
- a logic state analyser that can analyse binary signals (GPIO outputs) and different encoded buses (I2C, SPI, and UART) and decode received data;
- a LCR meter able to test those components with an additional module.

The powerful configuration of this board can be easily seen[10]: a dual-channel 14-bit Analog to Digital converter working at 125 MSa/s and a dual-channel 14-bit Digital to Analog converter working at 125 MSa/s are present for fast

### 3.1 STEMLab Red Pitaya

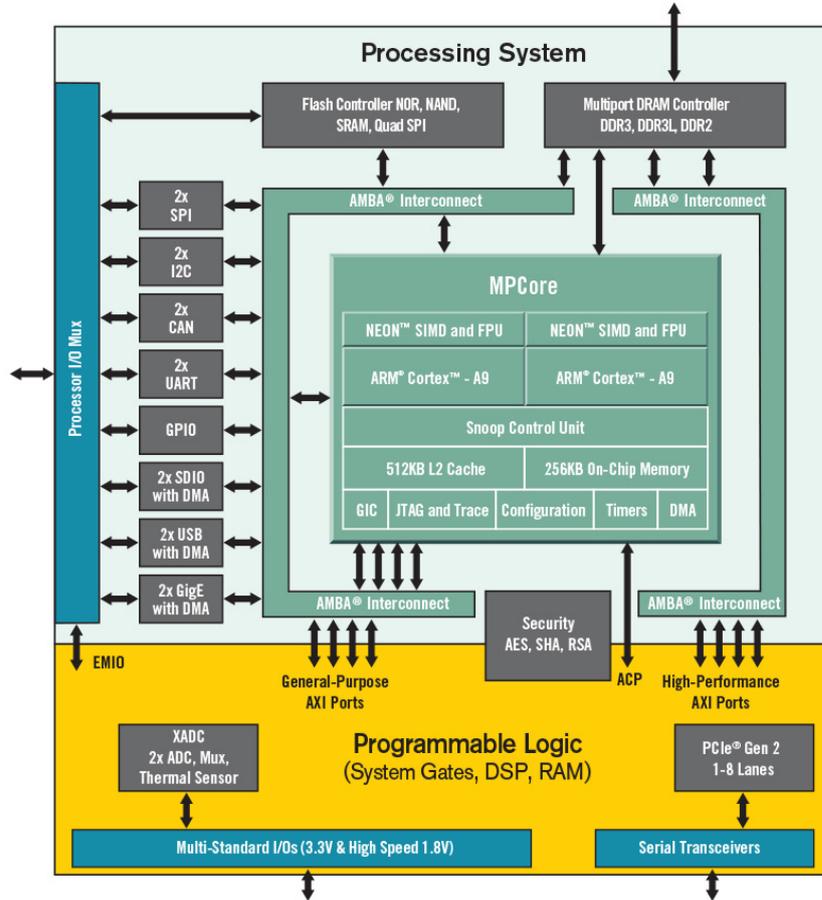


Figure 3.1: Zynq platform overview[11].

Input/Output, with input amplitude that can be set to 2 V or 20 V; two GPIO expansion port are present, one of them fully programmable; the System on a Chip integrates on one hand the dual core Cortex-A9 running at 766 MHz with two PLLs for clock generation and stabilisation and on the other hand the FPGA with the following characteristics[10]:

- 28000 programmable logic cells;
- 17600 LUTs;
- 35200 flip-flops;
- 512 MB of RAM divided in 36 kb blocks;
- 80 Digital Signal Processing units.

### 3.1 STEMLab Red Pitaya

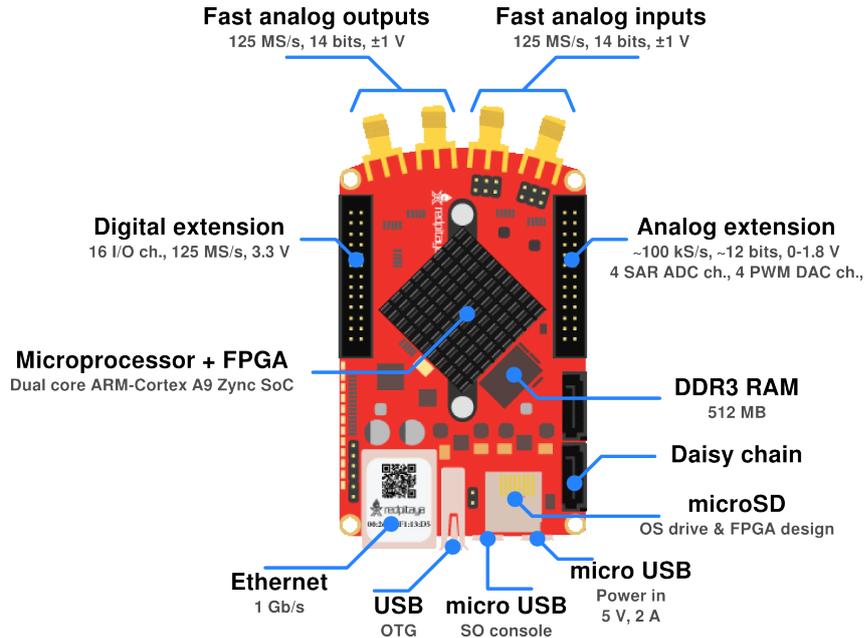


Figure 3.2: Red Pitaya hardware overview.[12]

This powerful configuration led to a high ductility while carrying out several measurement techniques: redefining the logic part with Xilinx VIVADO editor, input data could be elaborated and sent to the processing system; data packets then had been formed and sent to the main computer to be elaborated with MATLAB using Secure Copy Protocol over Ethernet.

As Red Pitaya is an open source project, VIVADO IDE and SDK don't fully recognise the device via Ethernet nor USB; then, a full custom configuration project (.bit file) has been compiled then sent over Ethernet to the board to be loaded from custom OS from line terminal. This operation has allowed a deeper knowledge of Linux programming techniques but all the SDK monitoring systems were shut off.

#### 3.1.1 Linear Technology 2145

Red Pitaya board is equipped with LT 2145 dual channel ADC: it is a flash converter 14-bit resolution designed for high SNR, low jitter and low cross-talk application with maximum sampling frequency equal to 125 MHz[13]. The power supply voltage has to be stable and equal to 1.8 V, and the frequency performances are in the suitable range for no missing code and low harmonic distortion condition, to be characterised.

Analysis results of several tests are provided, carried out in extreme conditions

### 3.1 STEMLab Red Pitaya

---

to rally the component ( $T = 25\text{ }^\circ\text{C}$ ,  $f = 70\text{ MHz}$ ,  $A = -1\text{ dBFS}$ ):

Parameter	value
DNL (max)	0.9
INL (max)	2.6
SINAD	72.8 dBFS

Table 3.1: Figures of merit from Linear Technology datasheet.

Those results have been verified in chapter 4 in different test conditions (different frequency span, 1 - 20MHz, different amplitude values, 2 - 5V<sub>pp</sub>): they fit within a good tolerance, as the INL and the DNL stay below the maximum value (reference section 4.5) and the SINAD value is coherent (reference section 4.9), as it has to be adapted to the maximum input amplitude (20 V) and re scaled in dBFS with the formula:

$$A_{dBFS} = 20\log_{10}\left(\frac{V_a}{V_b}\right) \quad (3.1)$$

with

- $V_a$  is the amplitude used in the test;
- $V_b$  is the amplitude of the test carried out by the Linear Technology (17.82 V).

## 3.2 Measurement techniques

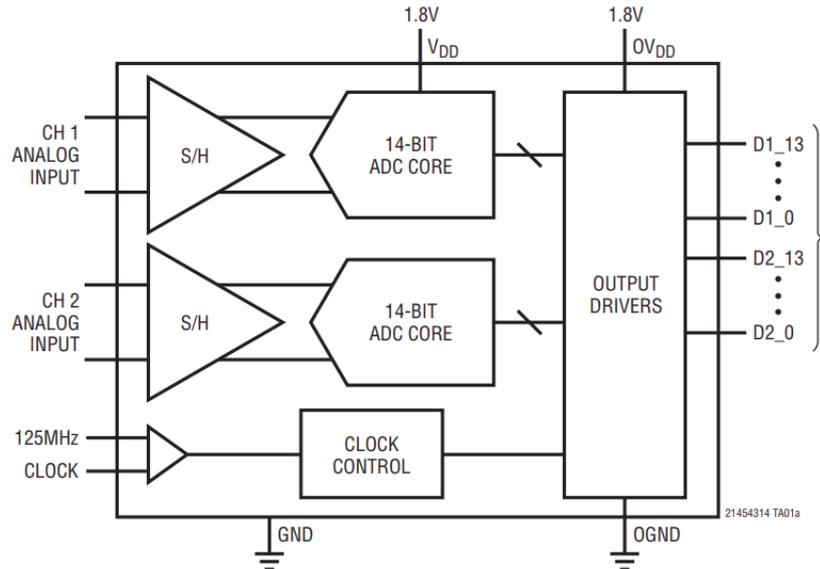


Figure 3.3: LT2145 core block diagram[13].

## 3.2 Measurement techniques

### 3.2.1 Histogram test for linearity

The histogram method is used to evaluate any kind of non linearity behaviour of a converter: the integral and differential non-linearity were this work starting point. This test compute the density of every available code from  $2^{N-1} - 1$  to  $2^{N-1}$  in 2's complement with N bits. During the test a repetitive signal is applied to the converter, generating the corresponding constellation of symbols at the output; in this case a sinusoidal signal has been chosen due to its bathtub-like statistical distribution.

Any deviation from the ideal output is underlined as evident spikes in the distribution, while the worst case of the ADC, the missing code case, can be easily found once collected all data and investigating if a recurrence bin is fixed at zero.

During conversion operation, the histogram test shows how many times a certain code word appears at the output starting from the analog input: the information is stored in different bins, representing every code recurrences. With N bits there will be  $2^N$  code bins and ideally their width should correspond to  $\frac{V_{FR}}{2^N}$ , where  $V_{FR}$  represents the full-scale range in volts of the converter. That is the point of the test: practically the code width is not fixed due to linearity errors, and this test counting recurrences is able to recognise this deviation.

## 3.2 Measurement techniques

---

Due to the test nature the number of samples to be collected to high confidence can be quite large[4]:

$$M = \frac{\pi 2^{N-1} Z_{\alpha/2}^2}{\beta^2} \quad (3.2)$$

where

- N is the number of bits;
- $\beta$  is the DNL resolution;
- $Z_{\alpha/s}$  is a parameter for 99% confidence resolution.

Since the converter on the board is 14 bits resolution, M is equal to 17077785, rounded to 20 million of samples for higher reliability.

Then the transition voltages between codes has been evaluated:

$$V_j = -A \cos\left(\frac{\pi}{M} \sum_{k=0}^j H_k\right) \quad (3.3)$$

where

- A is the sinusoidal amplitude;
- $H_k$  is the corresponding recurrence bin of a word.

Eventually DNL and INL has been computed through:

$$DNL_j = \frac{V_{j+1} - V_j}{1LSB} \quad (3.4)$$

$$INL_j = \frac{V_j - V_1}{1LSB} \quad (3.5)$$

where

- $V_j$  is the corresponding threshold voltage;
- $V_1$  is the first threshold voltage computed.

For sake of simplicity, A has been set to 1 V, in order to normalise the final result and reduce the sensitivity from this linear factor.

### 3.2.2 Histogram test for analog noise

Same configuration has been used to test the input referred noise (analog noise)[8]: converter input has to be terminated to a dc value, in this case set to 0 V and recording a number of output samples. Differently from bathtub-like graph from the previous case, the effect of the analog noise is a Gaussian-like output, as the converter keeps elaborating data around a fixed value with higher mean: then the noise is referred to the standard deviation of the distribution.

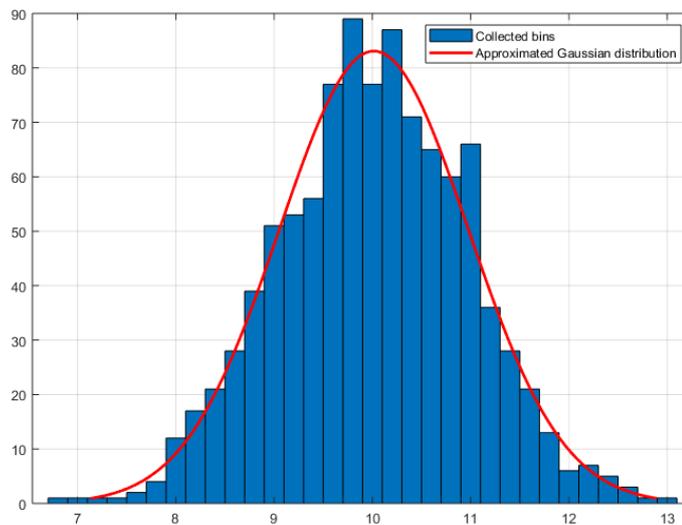


Figure 3.4: Example of noise referred output bins.

The standard deviation has been computed with MATLAB and normalised to the number of bits.

### 3.2.3 Histogram test for jitter evaluation

The structure previously used has been adapted also to evaluate jitter impact on sampled signal. The main idea was to sample the signal at every 0 V positive-edge crossing (reference figure 3.5), collecting all the output symbols and computing the standard deviation of the Gaussian distribution. The output in this case has been similar to the analog noise evaluation, but the meaning is totally different: the output is the code sampled oscillating around the theoretical value (0), and the output is evaluated in seconds to understand the difference between subsequently sampling period.

The jitter is then evaluated with the formula[9]:

## 3.2 Measurement techniques

---

$$J = \frac{\sigma}{\pi f_{in}} \quad (3.6)$$

Although the formula in literature included the normalisation in respect to the number of bits it has been decided not to normalise the result to respect needed of the SINAD Intersil formula (reference 2.9).

The sampling jitter is a combination of external clock jitter and internal aperture jitter of the converter: one task of this evaluation was possibly to understand which could be the jitter limiting performances, the jitter of the clock oscillator or the jitter caused by the sample and hold unit in the converter, better analysed in Chapter 4. Jitter value has been evaluated also through an Agilent frequency meter analysing the board equipped with a hardware structure (reference figure 3.12) to buffer and measure it on GPIO port using an oscilloscope with infinite permanence function.

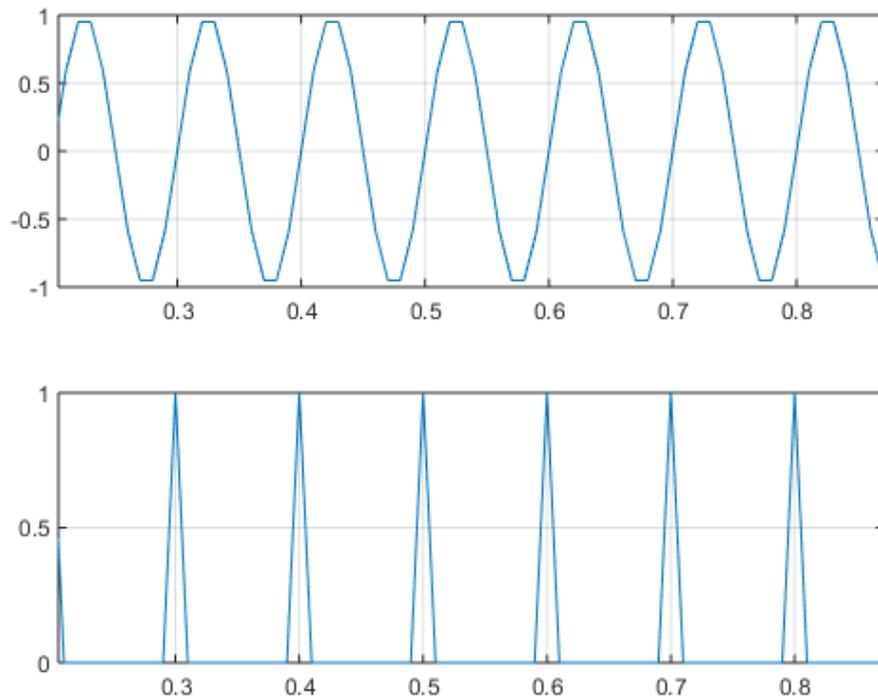


Figure 3.5: Example of 0V crossing sampling to jitter evaluation. On the top, the signal to be sampled; on the bottom, Dirac comb representing sample period.

### 3.2.4 THD evaluation

The Total Harmonic Distortion has to be then evaluated. Thus the measurement could be performed on the sampled data, Red Pitaya tool has been chosen to test also the custom OS functions. The output is a spectrum analysis (reference example figure 2.11) and cursors can be used as every spectrum analyser to measure voltage peaks and to implement THD evaluation following the formula 2.8.

### 3.2.5 SINAD evaluation in frequency domain

In conclusion another way to evaluate SINAD value has been found: computing this term in frequency domain is simpler, because a formula is applied on Red Pitaya spectrum analyser output[4]:

$$SINAD = 10 \log \left[ A_m^2 \left( \sum_{k=1}^{m-1} A_k^2 + \sum_{k=m+1}^{M/2} A_k^2 \right)^{-1} \right] \quad (3.7)$$

where

- $m$  is the data related to the fundamental: since the sampling period is not precisely a multiple of the input frequency, data around the fundamental have been also taken (limit set to 2 maximum);
- $A_m$  is the fundamental amplitude;
- $A_k$  is the data processed amplitude;
- $M$  is the number of data samples.

The main problem related to this approach is the lack of deepening of single distortion term in noise analysis, even if the frequency domain approach is quite simple and fast.

In the output file of Red Pitaya tool the two nearest values to the main frequency have been considered (reference figure 3.6), as the signal power is not perfectly centred on the principal line: this speculation seems working well evaluating the power in this specific case.

The built-in tool is not stable: the tool often doesn't give anything in output, and with some windows changes and tweaks some files have been collected (with few points). The output .csv file function is then glitchy and is dependent on the frequency windows observation setting. In order to proceed to the analysis, output .txt file have been created unifying different observation files and proceeding with MATLAB analysis.

### 3.3 Hardware implementation

---

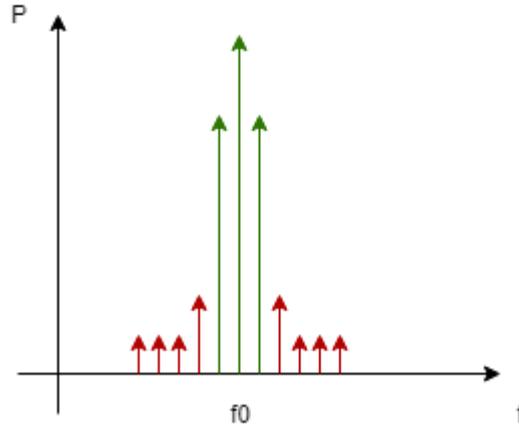


Figure 3.6: Evaluating SINAD in the frequency domain, only bins with relevant power have to be considered (green ones) in respect to noise (red ones). As the sampling rate is not a perfect multiple of the input signal, the power is spread over the nearest bins in respect to the main frequency.

## 3.3 Hardware implementation

One goal of this thesis was to implement those different techniques on the ductile structure of a FPGA board. Two projects have been implemented in Xilinx VIVADO in order to collect data processed from the converter and send through the PS to the main computer, and analyse board clock.

Due to the lack of documentation regarding this open source platform, a custom configuration file for Zynq SoC has been written, in order to have the processing system part functioning while reprogramming the hardware side. Thus this precaution worked, Xilinx SDK didn't recognised the board, so the configuration file has to be copied on the Red Pitaya in the root folder and then loaded with *cat* command in the */root/dev/xdevcfg* configuration folder.

### 3.3.1 ADC data converter

Several requirements and problems have been encountered:

- high amount of sampled data from the converter has to be stored then transferred to be elaborated, but the temporarily store device has to be chosen: every on board solutions have not enough bandwidth;

### 3.3 Hardware implementation

---

- input data has to be converted from raw to be elaborated and adapted to be sent on AXI communication bus;
- Ethernet scp protocol has not enough buffer to store temporarily data packet to be sent to the main PC.
- all of these requirements didn't allow to implement a real-time system: all data has to be analysed later.

Since in the histogram test the matter of interest is the statistical data of the periodic input signal and not the real-time stream, the project (reference figure [3.7](#)) has been implemented to solve previous requirements and problems.

### 3.3 Hardware implementation

---

The system works as follow:

- the raw input data are store in a FIFO at first. As the maximum sampling rate is 125 MSa/s and every code is composed by 14 bits, the maximum bandwidth required is 218.75 MB/s. Xilinx FIFO generator has the required bandwidth, maximum up to 600 MB/s[14]. The custom state machine asserts *wr\_en* signal and the FIFO starts to collect samples directly from ADC after the reset state;
- when FIFO is full, it asserts *full* signal; the state machine asserts *rd\_en* to starts transfer symbols to AXI4 ADC interface, until is senses *empty*: then it starts over to refill the FIFO. The queue has another function: as it has been decided to write samples in data packets to the SD card to store them temporarily, the FIFO is used as an interface between different clock domains (125 MHz - 500 kHz) as the class-10 SD card has a maximum write speed of 10 MB/s;
- the ADC interface converts the raw symbols to hexadecimal 2's complement and adapt it to 32 bits width as AXI protocol requires[15], inserting in free bits a hexadecimal control code ("*feed*") to monitor the good sample operation; then, handshaking signals are asserted as AXI protocol requires to let PS through C code to read value and store it in SD card packet in a .txt file;
- eventually, the packet is transferred through scp protocol via Ethernet to the main PC and opened in MATLAB.

The file opened in MATLAB is converted and remapped to integer value, converted to bins to be counted and then formulas are implemented on data.

#### 3.3.2 Timing and ST diagrams

The timing constraints are related to the Xilinx IP cores: since there was the need of implement a custom state machine that had to manage all the signals related to the Xilinx IP rules and following the AXI communication protocol. At first, at startup, the FIFO has to be reset: according to the Xilinx constraints, the *rst* signal has to be asserted at least for 3 clock cycles of the slowest clock of the domain (500 kHz). Write and read enables have to be set to 0 during all this period (figure 3.8).

Then, data starts to be collected in the queue, with *wr\_en* asserted. Then when the FIFO is full, *full* is asserted for a single clock cycle. The state machine set the enable to 0, and switches to the second clock domain, preparing AXI device to be turned on (figure 3.9).

## 3.3 Hardware implementation

---

In conclusion, when the last symbol is sent at the PC, the signal empty is asserted for a single slowest clock cycle; then the C code checks how many symbols have been collected, and eventually the state machine starts again the writing cycle at 125 MHz. If the operation is over however, the hexadecimal control code turns to all zeros: the C code evaluate this condition and then put the whole system in idle mode (figure 3.10).

The State Transition Diagram is presented in figure 3.11, and sums the previous transitions made by hardware and led by the custom State Machine between filling the FIFO and downloading it transferring data to the main computer.

### 3.3.3 C code development

During the project development, C code has been written to be run on the ARM architecture. AXI ADC component writes data to a specific memory register a valid flag and data output: the code checks the flag and saves the symbol to a local buffer. As the PS sees the PL part as a memory-mapped device, the code sets the precise address (0x43c00000) using *mmap* function. Then, when the counter hits the requested number the PS has to detach the shared memory and writes the .txt file opened in append mode.

The communication between the board and the main PC has been resolved using a fixed IPv4 on the Red Pitaya and disabling the automatic IP linking on the main computer. The, with the program PUTTY, a communication over Ethernet via command line has been established, allowing to execute Linux commands directly on the custom OS.

### 3.3.4 On board clock analyser

As jitter evaluation has been difficult (explained in section ...) due to the low value and subsequently difficult instrument evaluation, another project has been developed to analyse it (reference figure 3.12).

It works as follow:

- an utility buffer sets the ADC clock from differential to single-ended;
- the Clocking Wizard IP boost the signal and divide it to different ratios (1 MHz, 5 MHz, 10 MHz, 20 MHz) to be able to measure them directly;
- the Combiner IP combines all the signals to fit the expansion port (GPIO) for the measurement.

### 3.3 Hardware implementation

---

The direct measurement on the oscilloscope was another confirmation of the values obtained from the indirect analysis, thus limited by instrument resolution: it has been carried out to verify the order of magnitude of the values, as unexpected good performances were calculated.

### 3.3 Hardware implementation

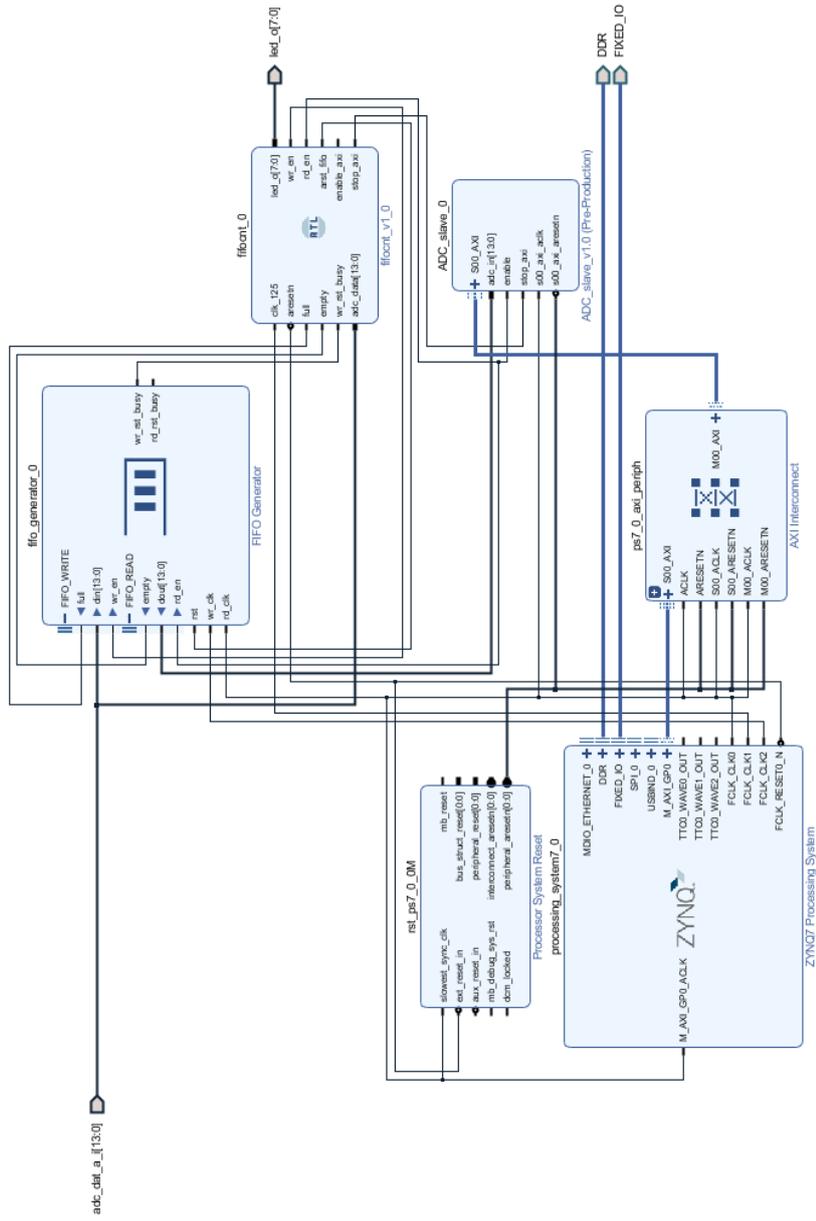


Figure 3.7: Scheme of the first project implemented. Main blocks are ADC AXI interface, custom state machine, FIFO for data storage, processing system to set Red Pitaya PS.

### 3.3 Hardware implementation

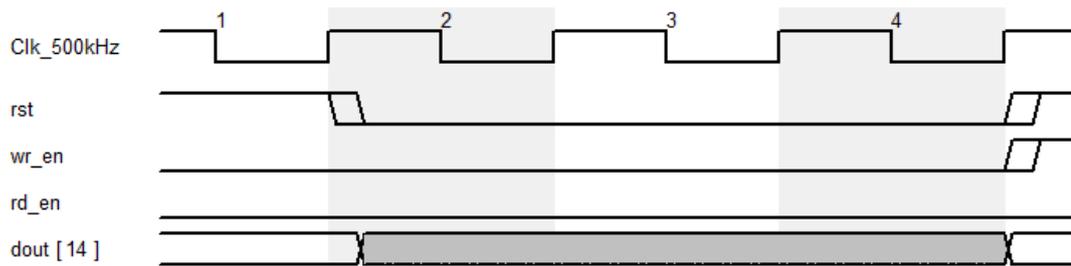


Figure 3.8: Reset state transition.

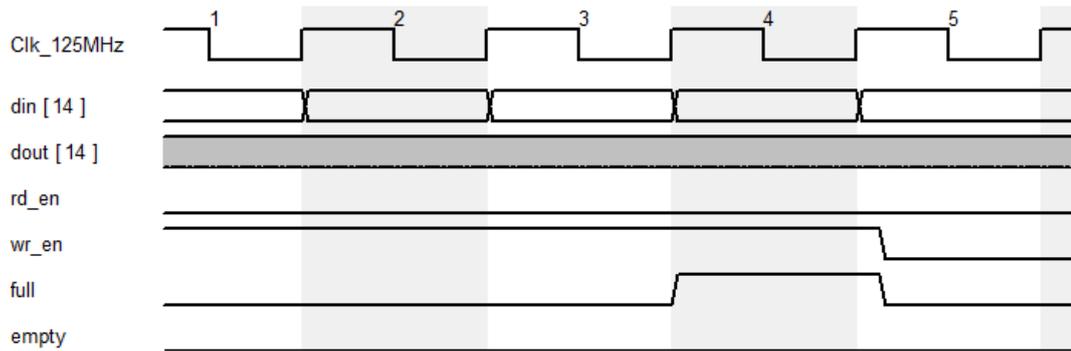


Figure 3.9: Write full transition.



Figure 3.10: Read empty transition.

### 3.3 Hardware implementation

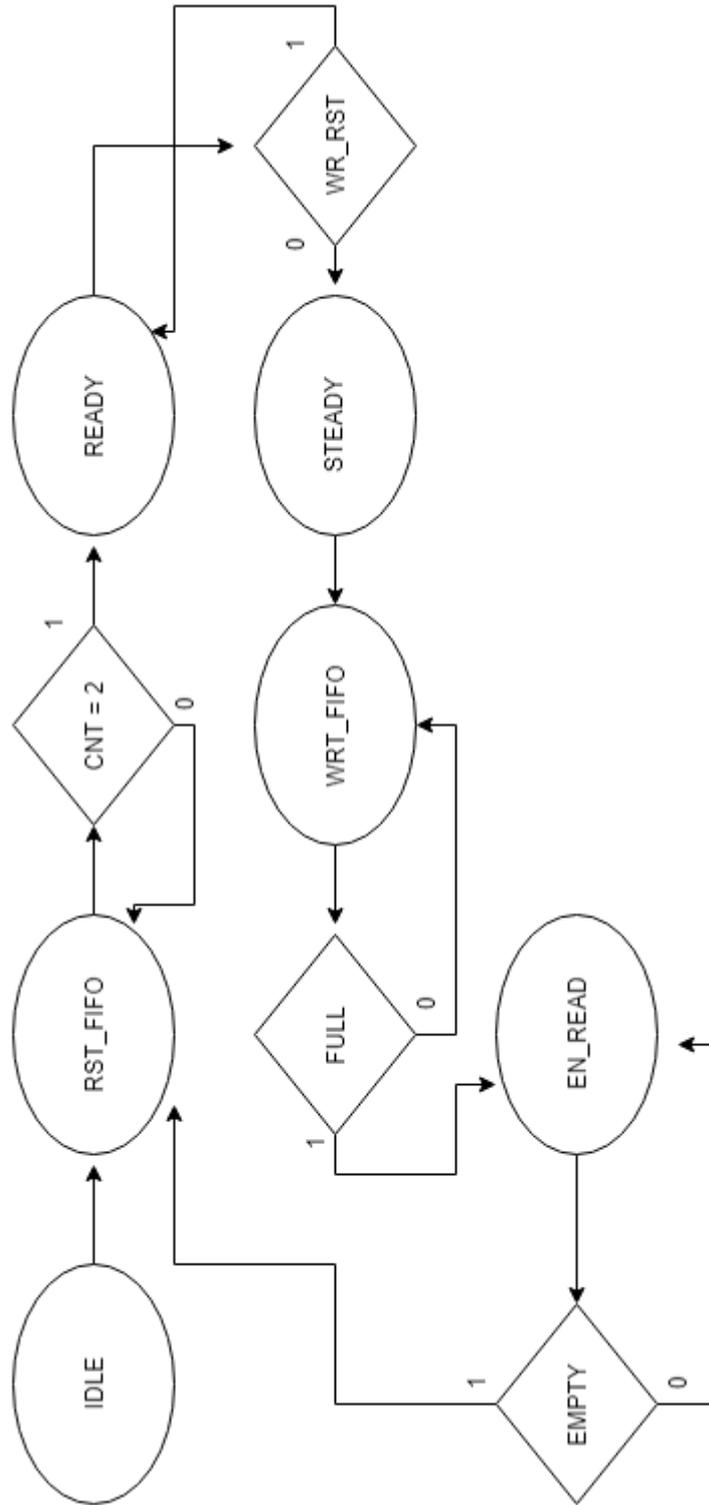


Figure 3.11: State transition diagram.

### 3.3 Hardware implementation

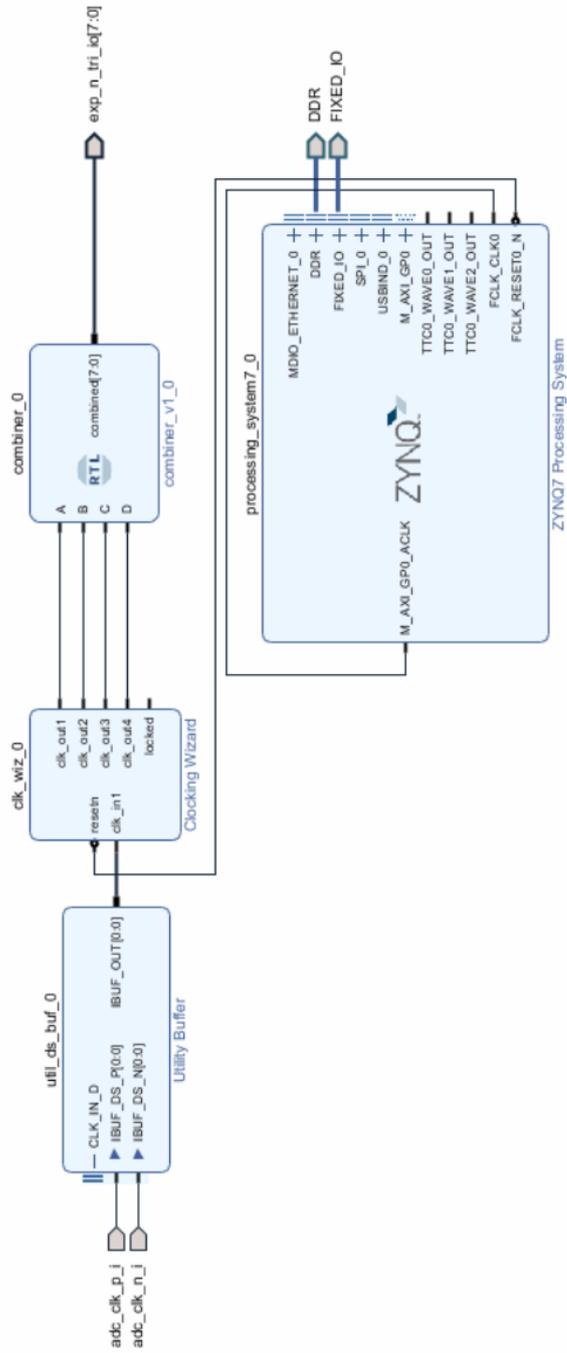


Figure 3.12: Scheme of the second project implemented. Main blocks are ADC clock buffer and wizard to make clock signal from differential to single ended and to stabilise it, GPIO port and Red Pitaya processing system.

# Chapter 4

## Results

In this chapter results of previous techniques are presented and analysed.

### 4.1 DNL and INL evaluation

#### 4.1.1 Board defect

When evaluating non linearities for the first time the bathtub distribution was present, underlining the correct sample operation, but another problem arose: even with the signal generator set to the maximum Red Pitaya input amplitude many symbols at the limits were missing (reference figure 4.1). Thus the DNL at the amplitude limits overcame 1 LSB, going against the no code missing declaration[13].

A solution has been found: setting the amplitude of the input sine wave to  $2.05 V_{pp}$  with board input limit set to  $2 V_{pp}$ , 25 mV per limit, has fixed the problem. Then, a theory has been speculated: although the converter can guarantee no missing code, the reference voltage is not stable as requested. The histogram test is then able to detect also non linearities of the board.

## 4.1 DNL and INL evaluation

---

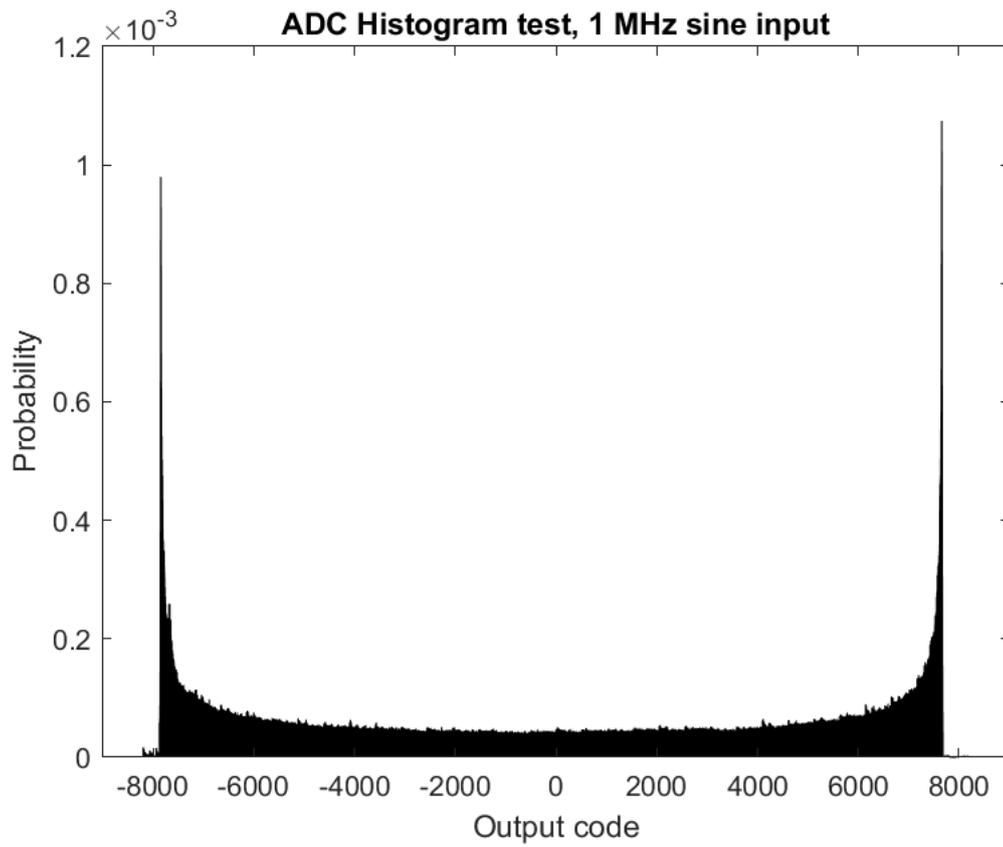


Figure 4.1: First evaluation of histogram test.

## 4.1 DNL and INL evaluation

---

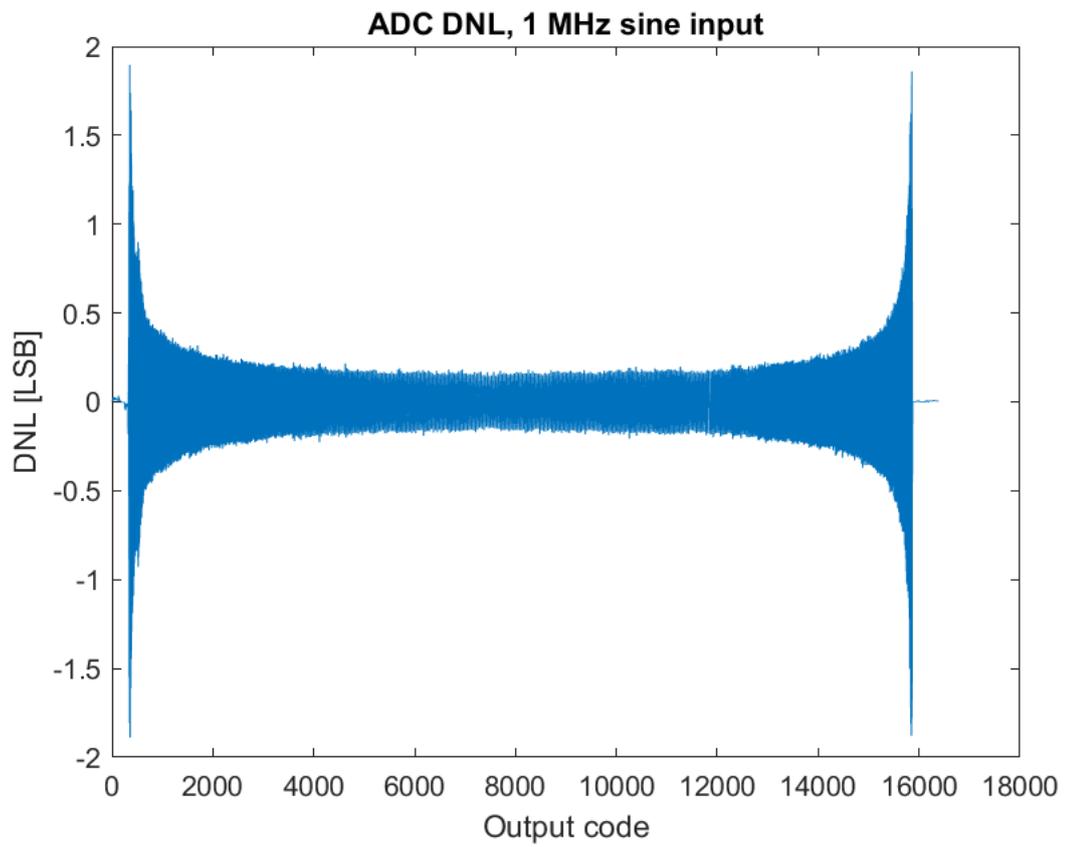


Figure 4.2: First evaluation of DNL.

## 4.2 Bathtub distributions

Correct results are then presented (reference figures 4.3 - 4.6). This output verifies hardware correctness in two ways: the first one is the output distribution, that has to be bathtub like as the input signal has no derivative at its boundaries, and then higher sampling probability (or slower slew rate); the second one is the presence of all symbols, in particular the spikes at the limit, evidencing a good sampling system and the fully saturated input to have a good conditioned experiment. As frequency increases, the probability of middle inputs has to lower, as the signal applied has faster slew rate in non-zero derivative points.

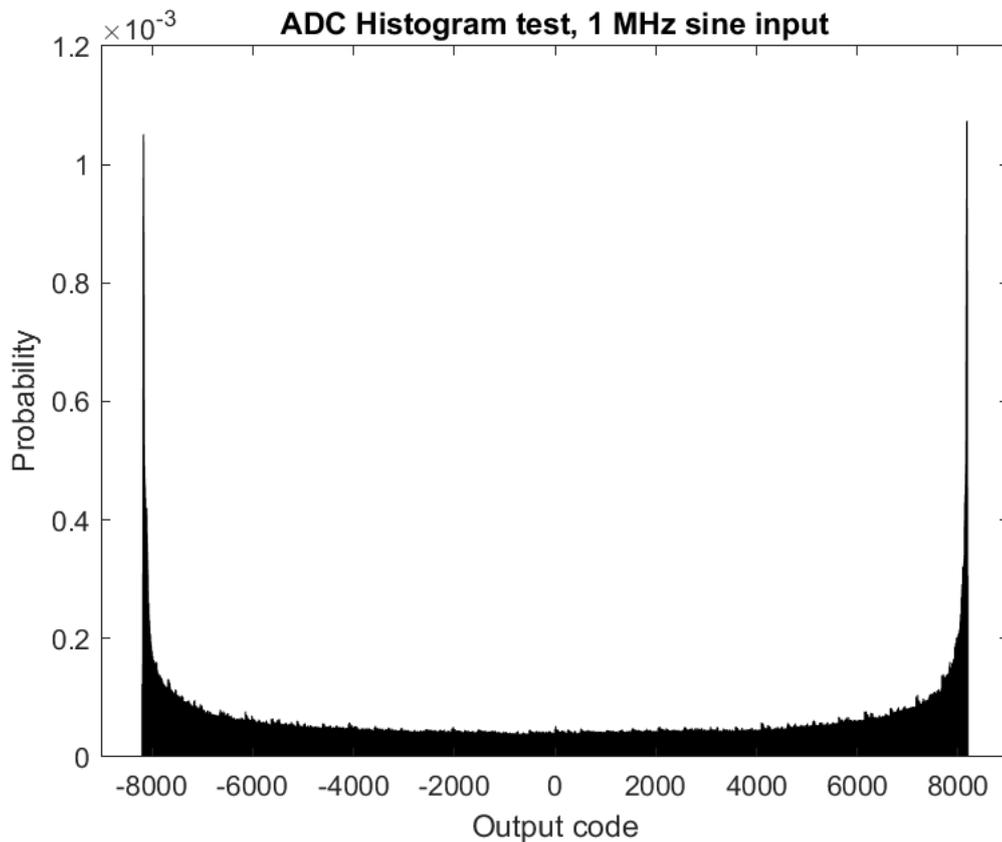


Figure 4.3: Histogram test, 1 MHz

## 4.2 Bathtub distributions

---

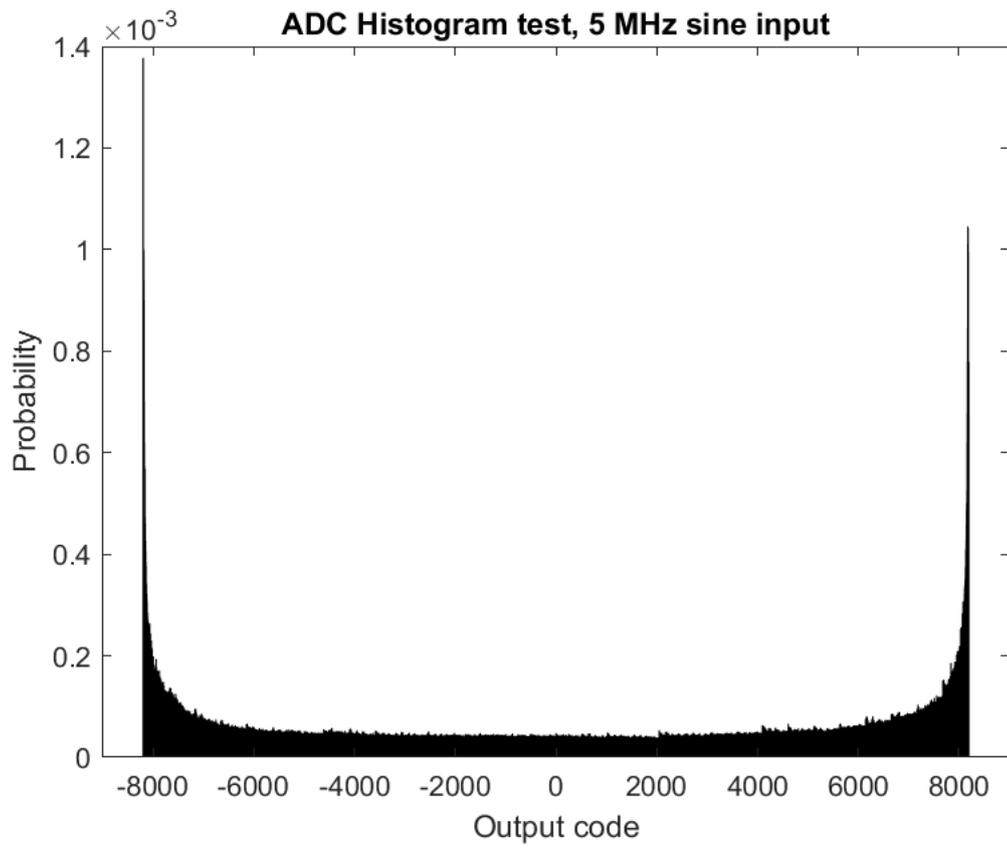


Figure 4.4: Histogram test, 5 MHz

## 4.2 Bathtub distributions

---

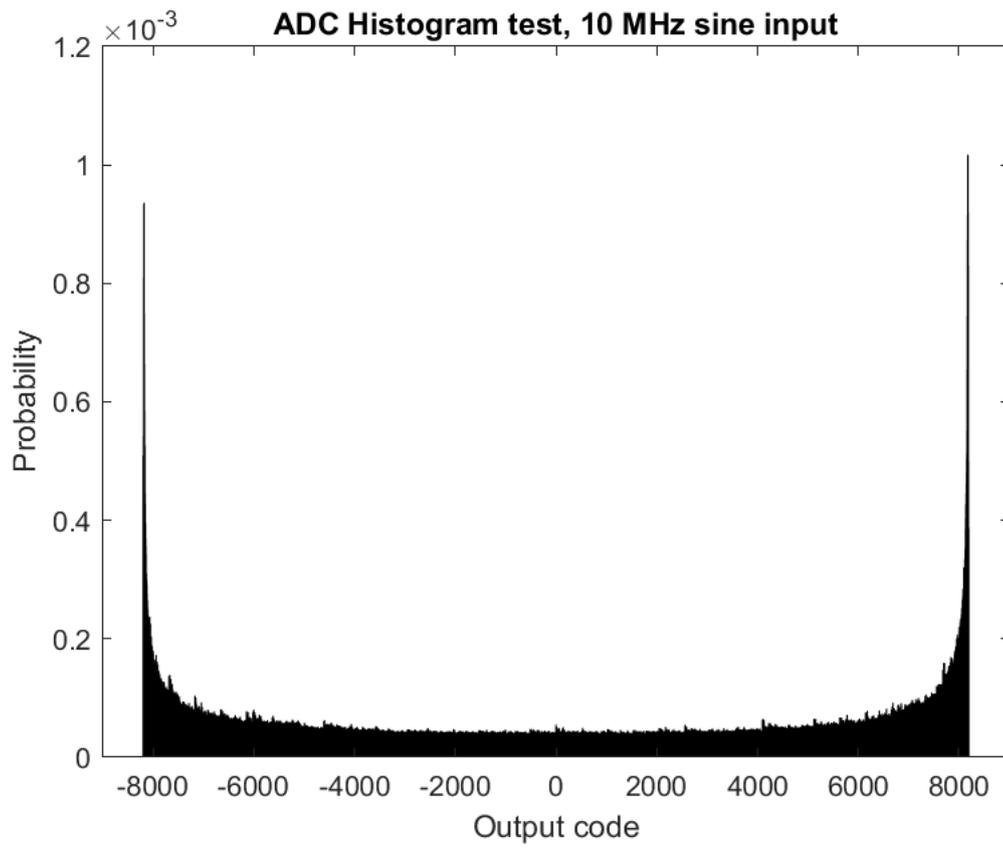


Figure 4.5: Histogram test, 10 MHz

## 4.2 Bathtub distributions

---

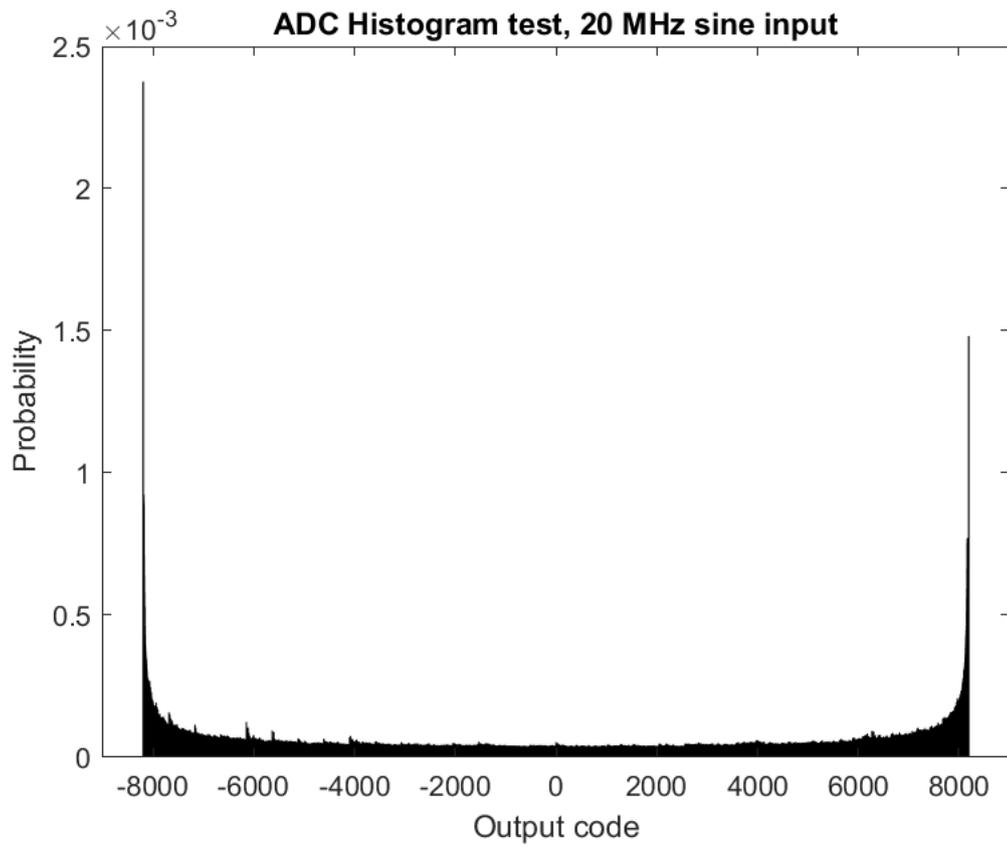


Figure 4.6: Histogram test, 20 MHz

### 4.3 DNL results

DNL results are then presented (reference figures 4.7 - 4.10). In normal conditions the DNL value is nearly negligible, except for input amplitude limit: according to previous speculation, due to the non-stable reference voltage and the slower signal transition, the deviation from one code to another worsens, though it never overcomes the 1 LSB boundary.

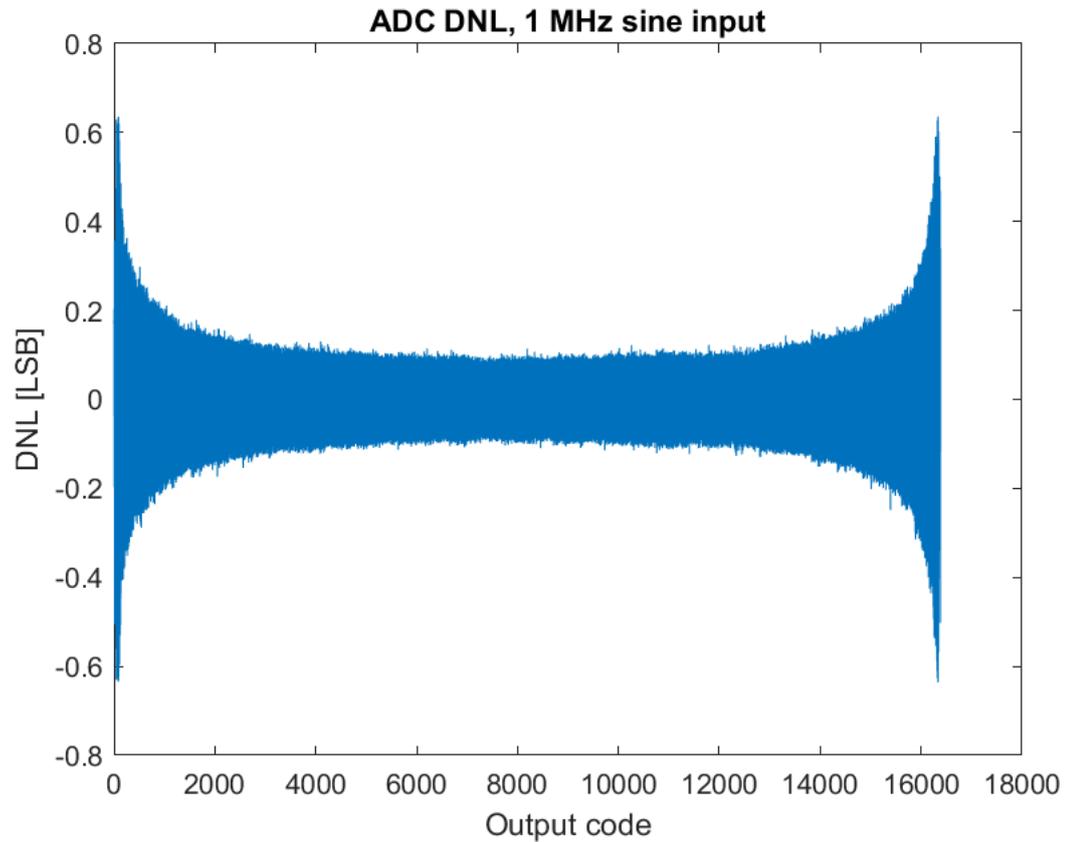


Figure 4.7: DNL evaluation, 1 MHz

### 4.3 DNL results

---

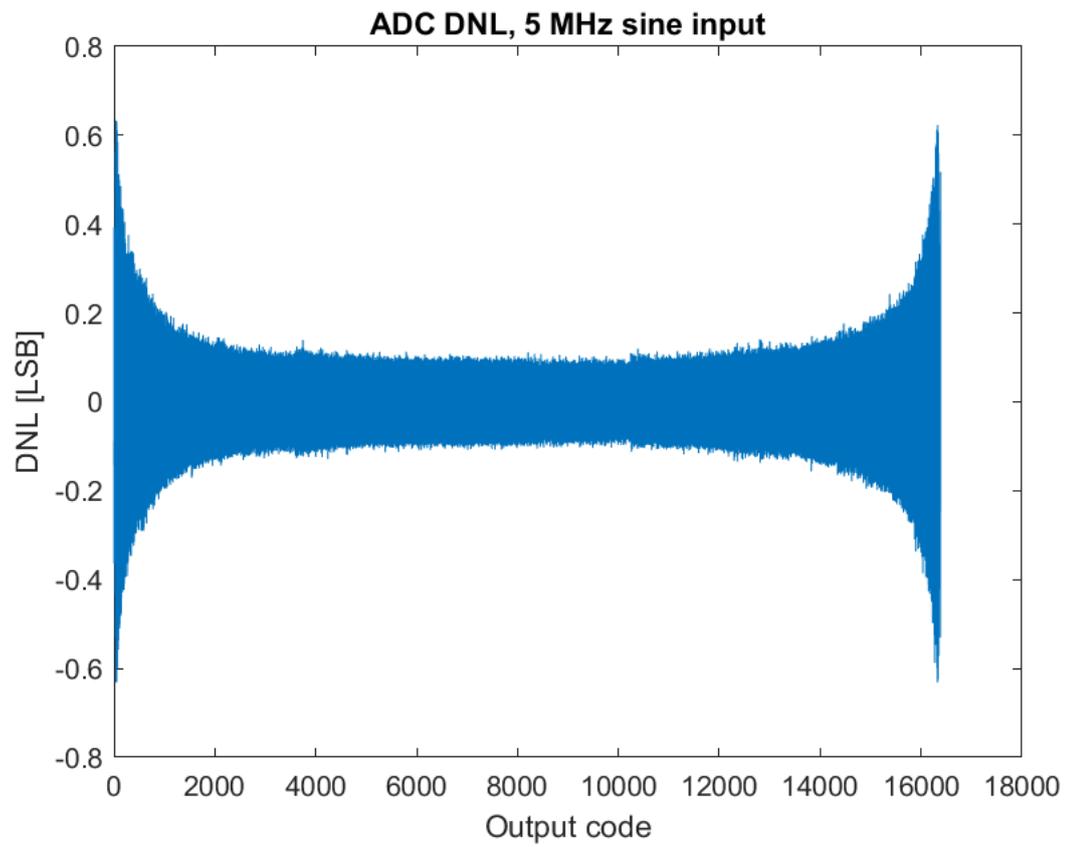


Figure 4.8: DNL evaluation, 5 MHz

### 4.3 DNL results

---

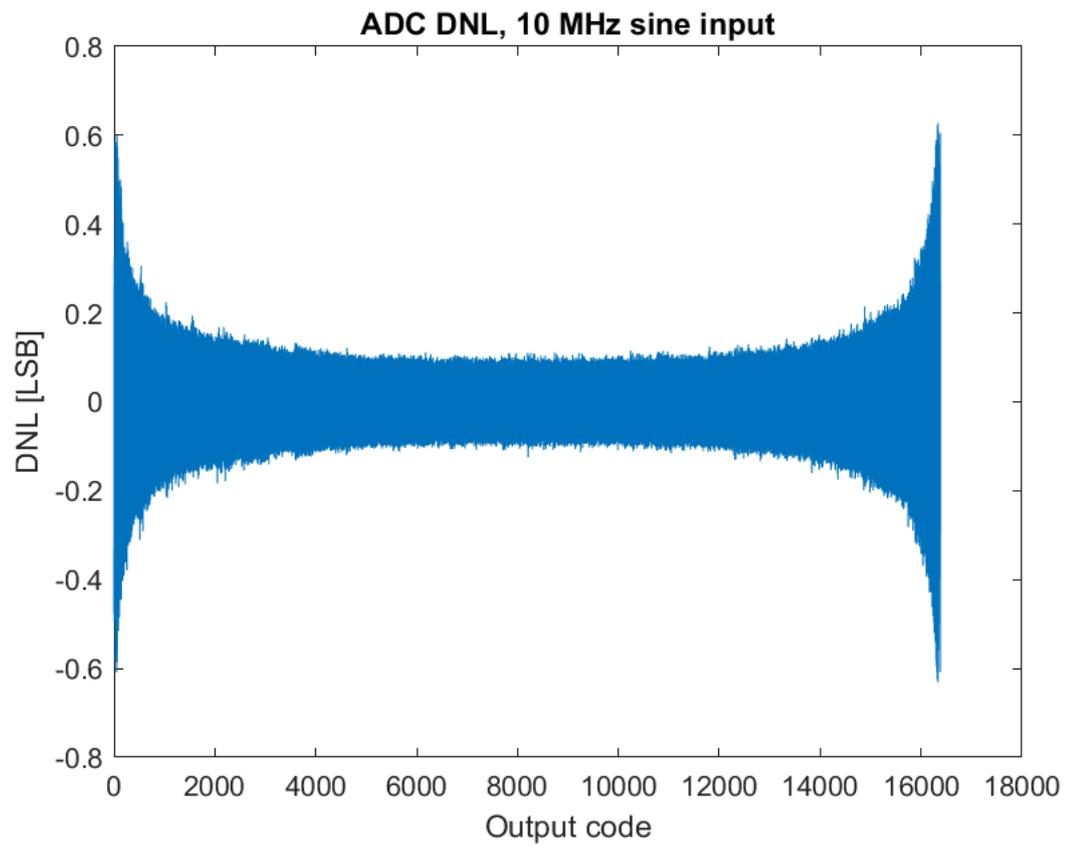


Figure 4.9: DNL evaluation, 10 MHz

### 4.3 DNL results

---

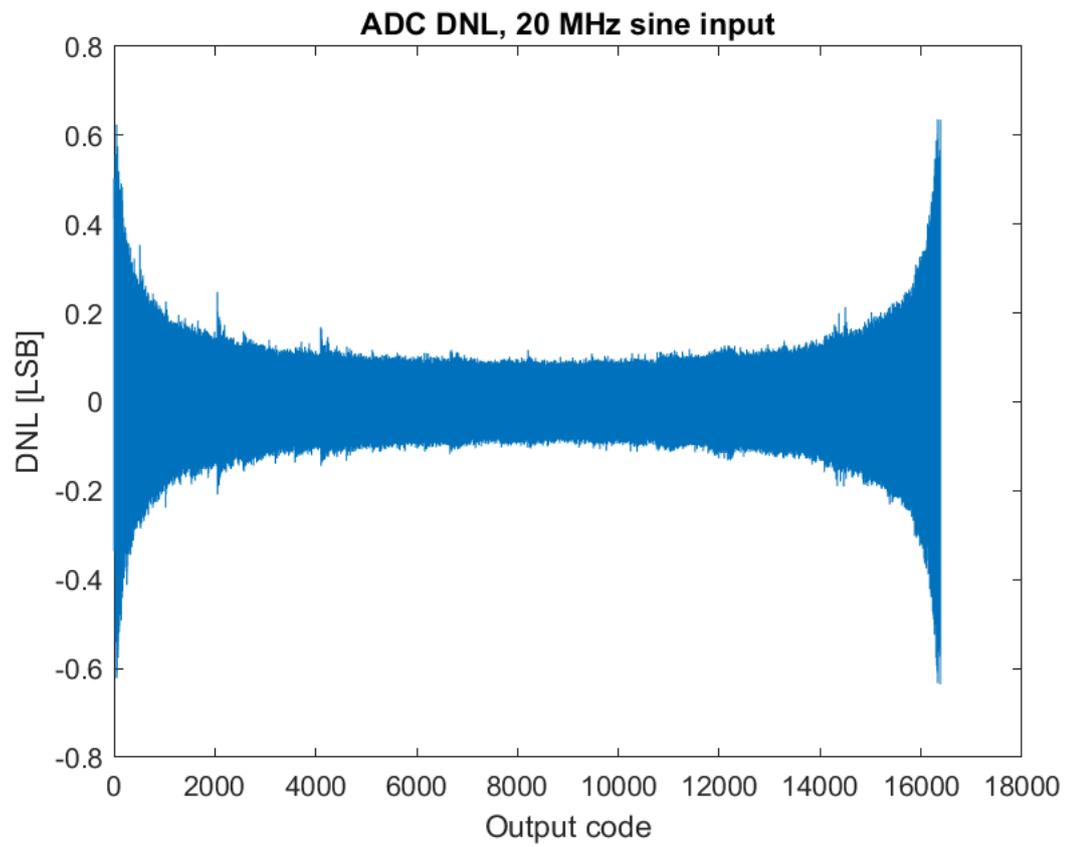


Figure 4.10: DNL evaluation, 20 MHz

### 4.4 INL results

INL results are then presented (reference figures 4.11 - 4.14). As INL and DNL are strictly connected, INL verification is one step to investigate DNL analysis correctness: as the previous case the distance from the ideal transfer function degrades near signal limits, but it's stable in the middle.

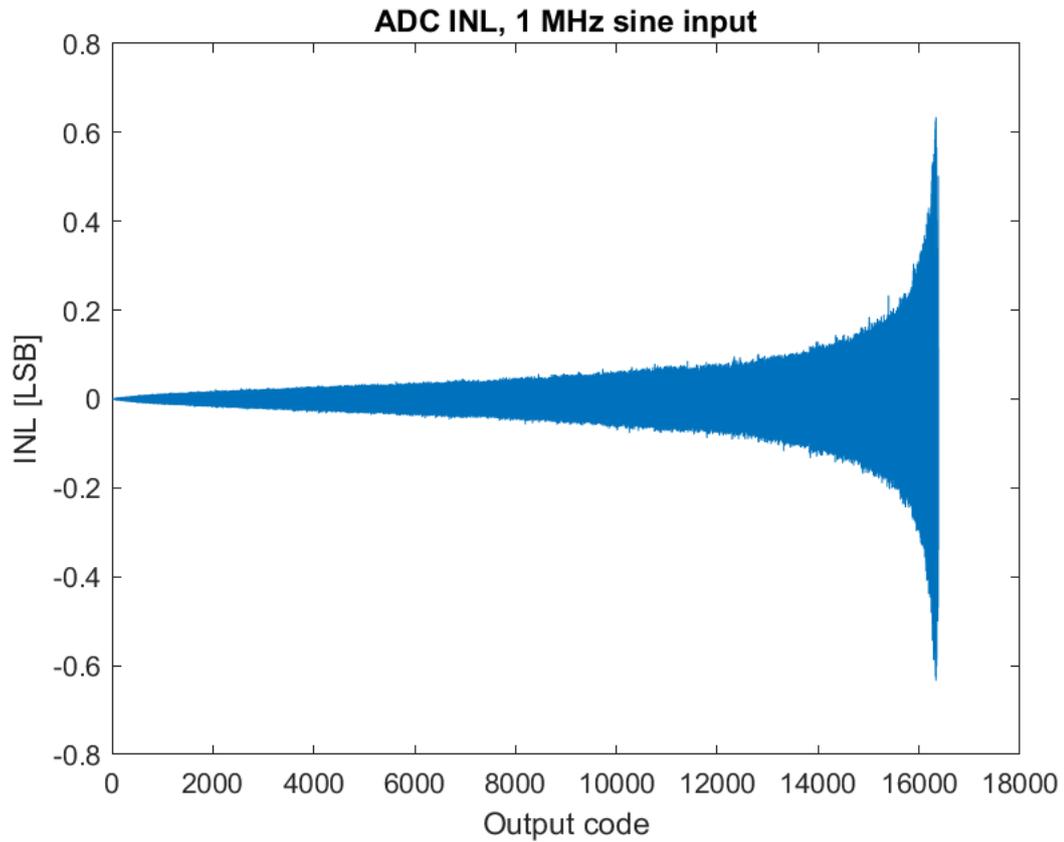


Figure 4.11: INL evaluation, 1 MHz

#### 4.4 INL results

---

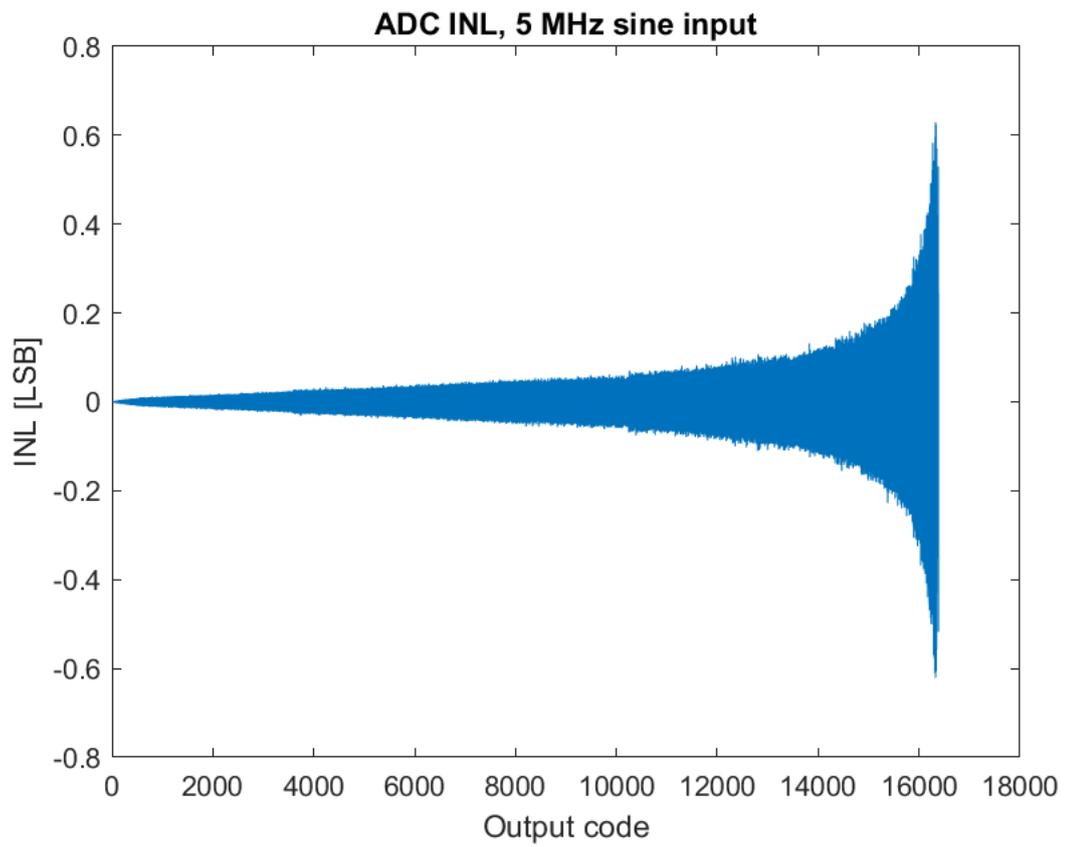


Figure 4.12: INL evaluation, 5 MHz

#### 4.4 INL results

---

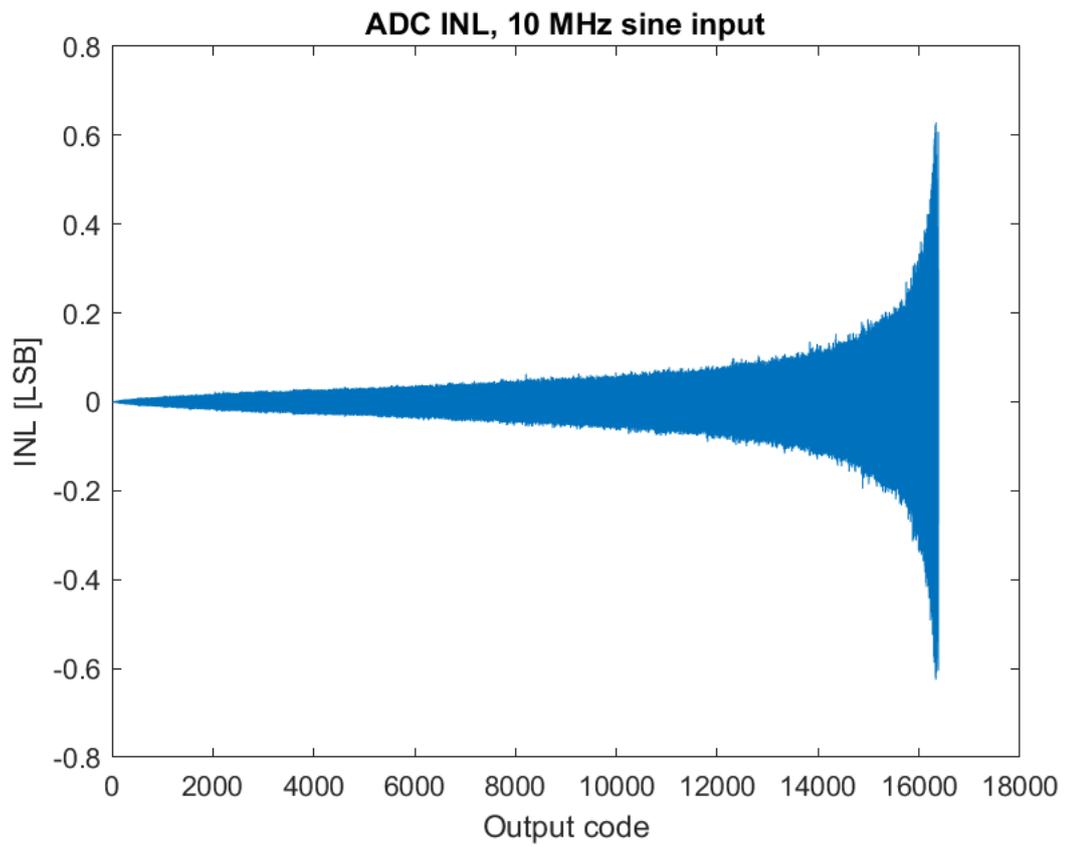


Figure 4.13: INL evaluation, 10 MHz

#### 4.4 INL results

---

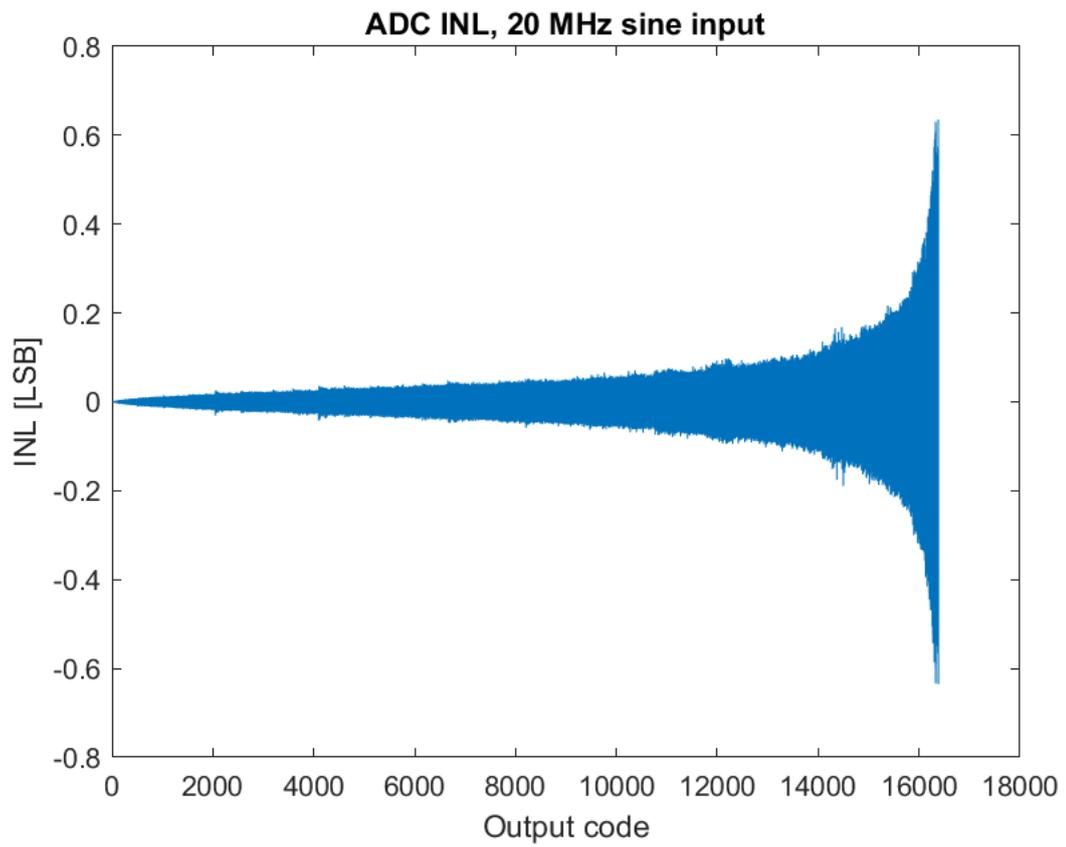


Figure 4.14: INL evaluation, 20 MHz

## 4.5 Maximum DNL - INL

Frequency [MHz]	Maximum DNL [LSB]
1	0.64
5	0.63
10	0.62
20	0.63

Table 4.1: Maximum DNL values in respect to the frequency.

Frequency [MHz]	Maximum INL [LSB]
1	0.63
5	0.63
10	0.63
20	0.64

Table 4.2: Maximum INL values in respect to the frequency.

As it can be seen, in the frequency span of the analysis INL and DNL doesn't heavily affect the signal: the maximum value of the non linearities has been reported and used to evaluate the distortion in the worst case. In normal operations however the DNL value, real limit to performances, is computed with mean value of 0.2 LSB. As declared, no missing code have been found and non-linearities values are in the tolerability range.

The problematic parts of the signal have been evidenced: the points with null derivative of the sine wave are the more probable and the more noisy due to the major overall noise impact on lower slew rate and the higher recurrence during noise evaluation. DNL has a direct impact on lowering ideal SNR value: the fact that, in this case, it doesn't depend directly on converter's performance, it is a good way to separate board and device characterisation, evidencing that the same converter on another commercial board can be more reliable.

## 4.6 THD evaluation

Total harmonic distortion has been then evaluated with technique exploited in chapter 3 (reference figure 4.15 - 4.18) and a plot of computed values is presented (reference figure 4.19). Peak values have been evaluated with embedded tool of Red Pitaya directly on the graph, computing then on MATLAB. As frequency increases, the harmonic distortion is not negligible anymore: the speculation is that the more the frequency gets near to the limit, the more upper harmonics distorts input signal and worsens all the frequency values, especially SINAD one.

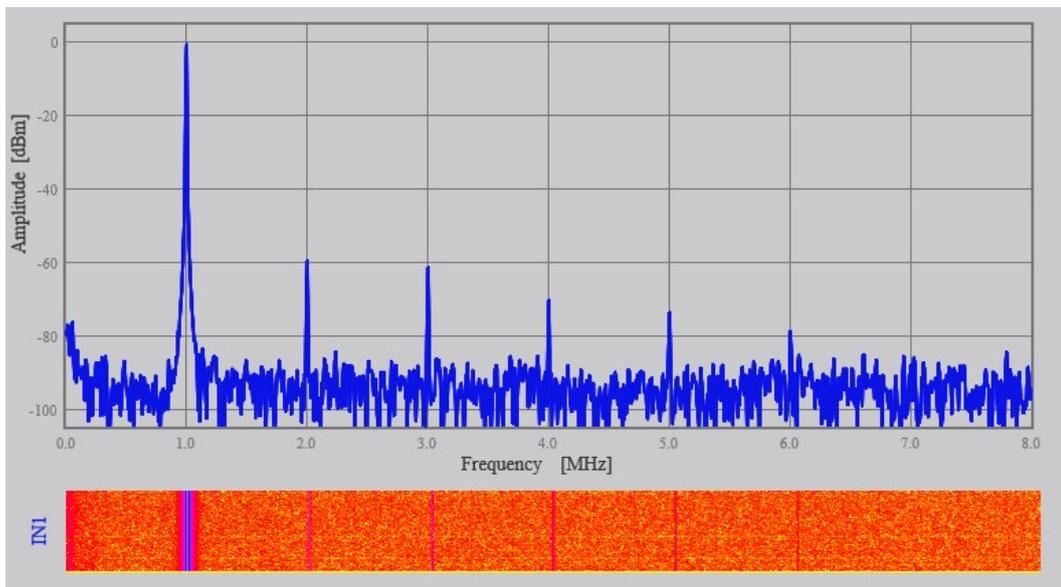


Figure 4.15: THD evaluation, 1 MHz

## 4.6 THD evaluation

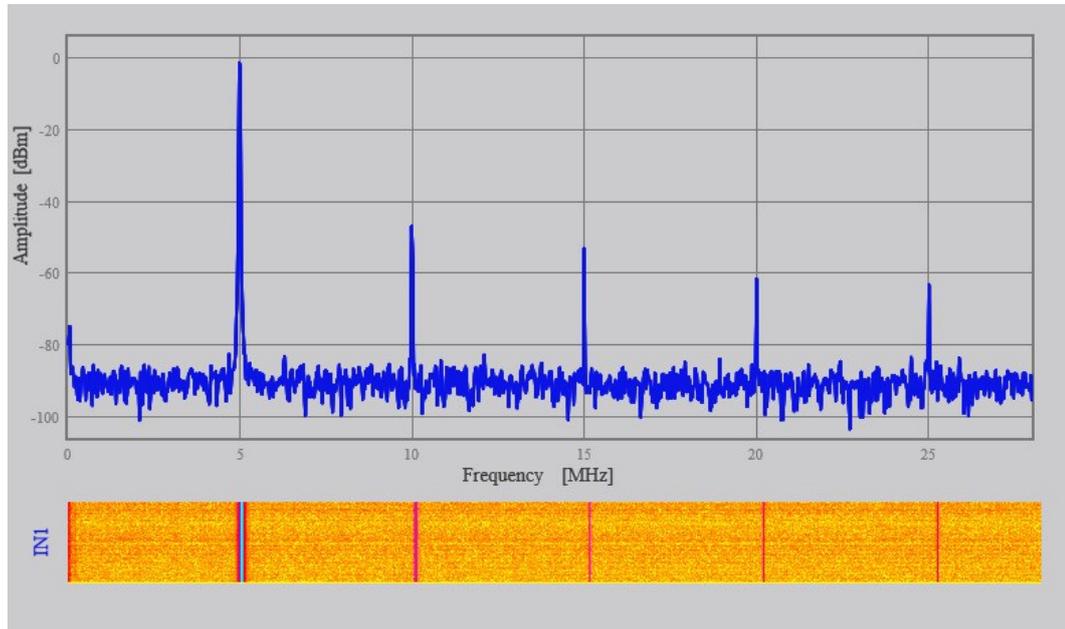


Figure 4.16: THD evaluation, 5 MHz

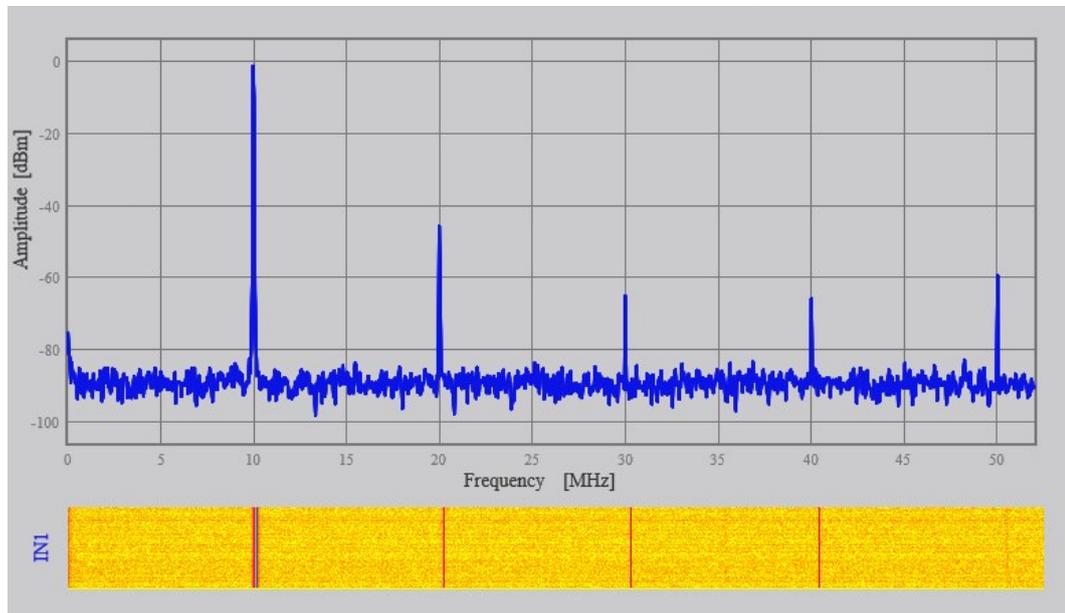


Figure 4.17: THD evaluation, 10 MHz

## 4.6 THD evaluation

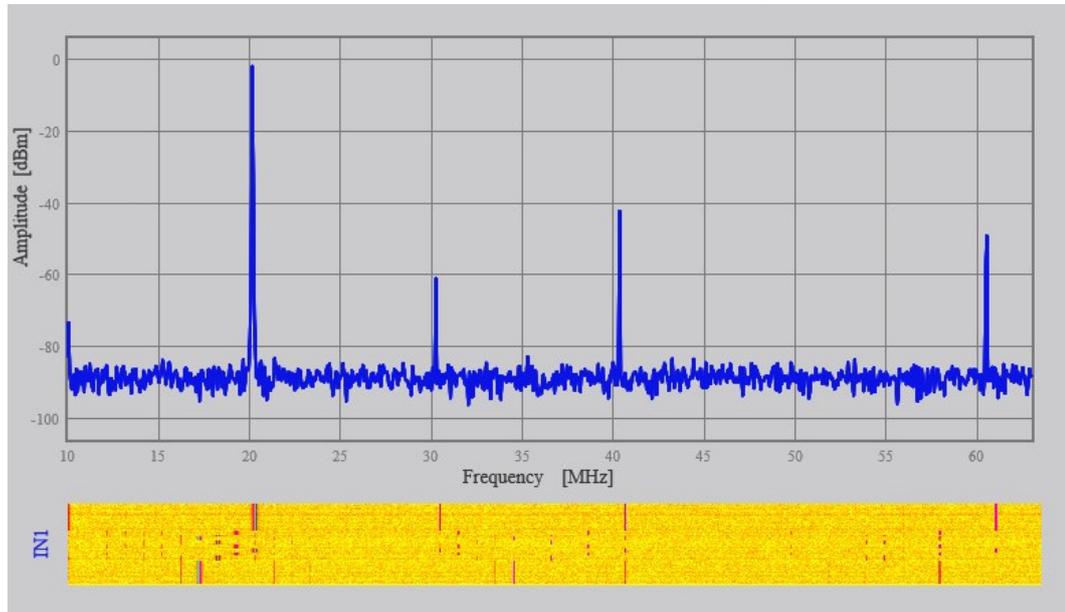


Figure 4.18: THD evaluation, 20 MHz

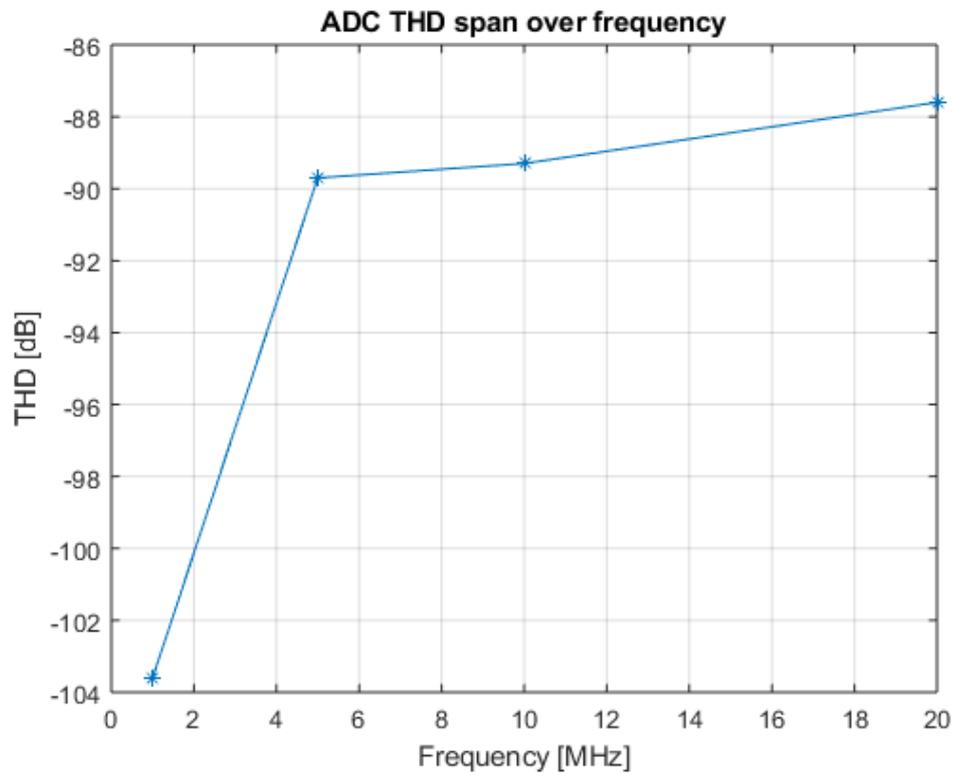


Figure 4.19: THD values over frequency.

### 4.7 Analog noise evaluation

The analog noise has been evaluated: the effect is a spurious driven voltage generator, summing up all the noise contributions presented in section 2.1. What is to be understood is if the overall input noise is negligible in respect to other contributions or not.

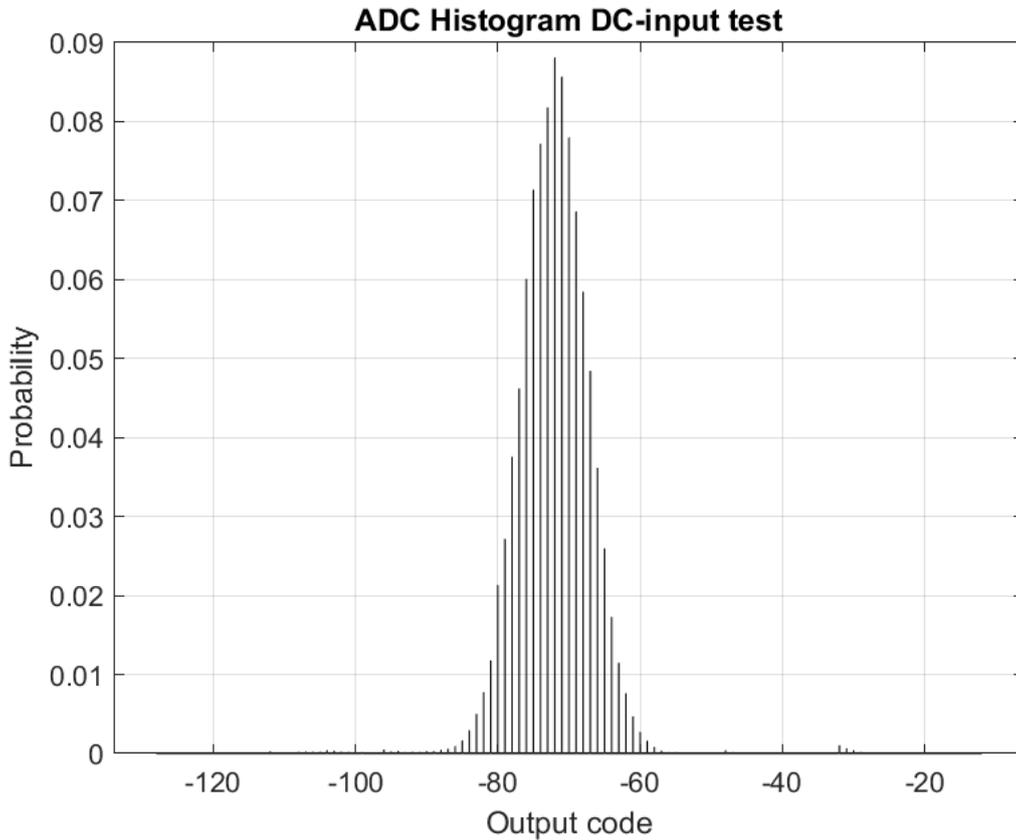


Figure 4.20: Analog noise evaluation, input grounded.

The result has been normalised to the number of bits, and it is evaluated as 0.12 LSB. Resorting to the formula 2.3, and evaluating that 0.12 LSB are equal to 7.32  $\mu\text{V}$  this term can be considered negligible for frequencies of interest.

## 4.8 Jitter evaluation

Overall jitter has been computed with the technique discussed in the previous chapter. Since the object of study was a commercial board, a jitter in the order of ns over a period of  $\mu\text{s}$  was expected: since after the analysis results in the order of the ps, another test with the second hardware project has been carried out.

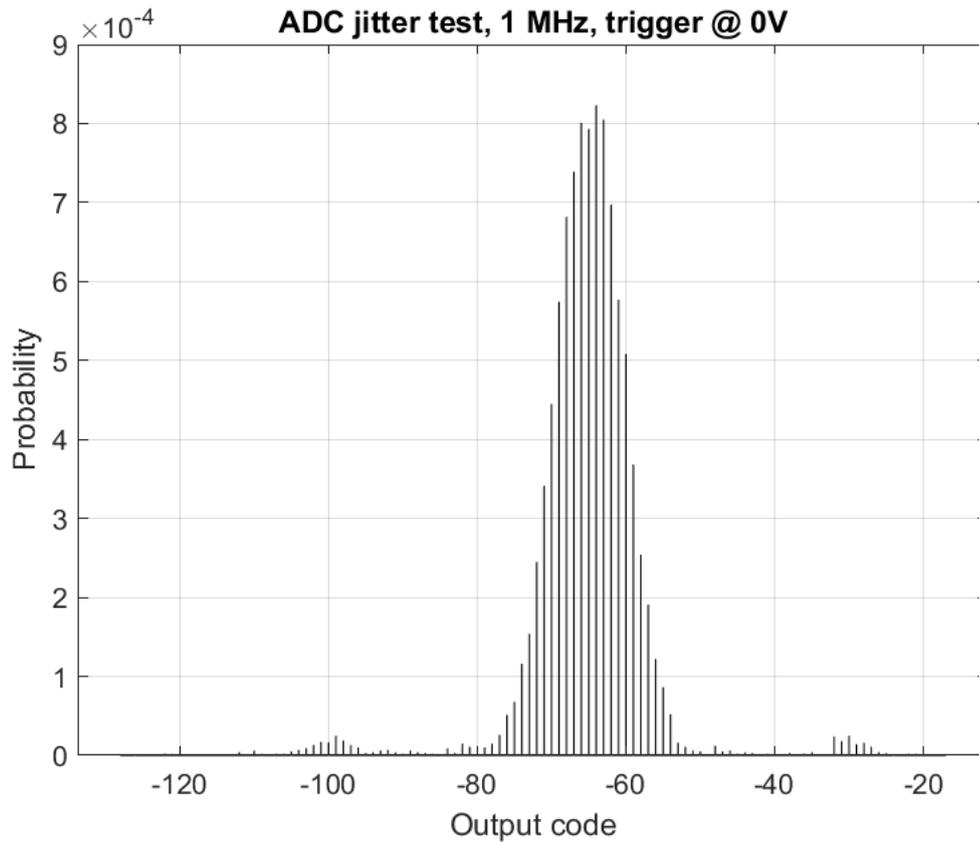


Figure 4.21: Overall jitter evaluation, 1 MHz

## 4.8 Jitter evaluation

---

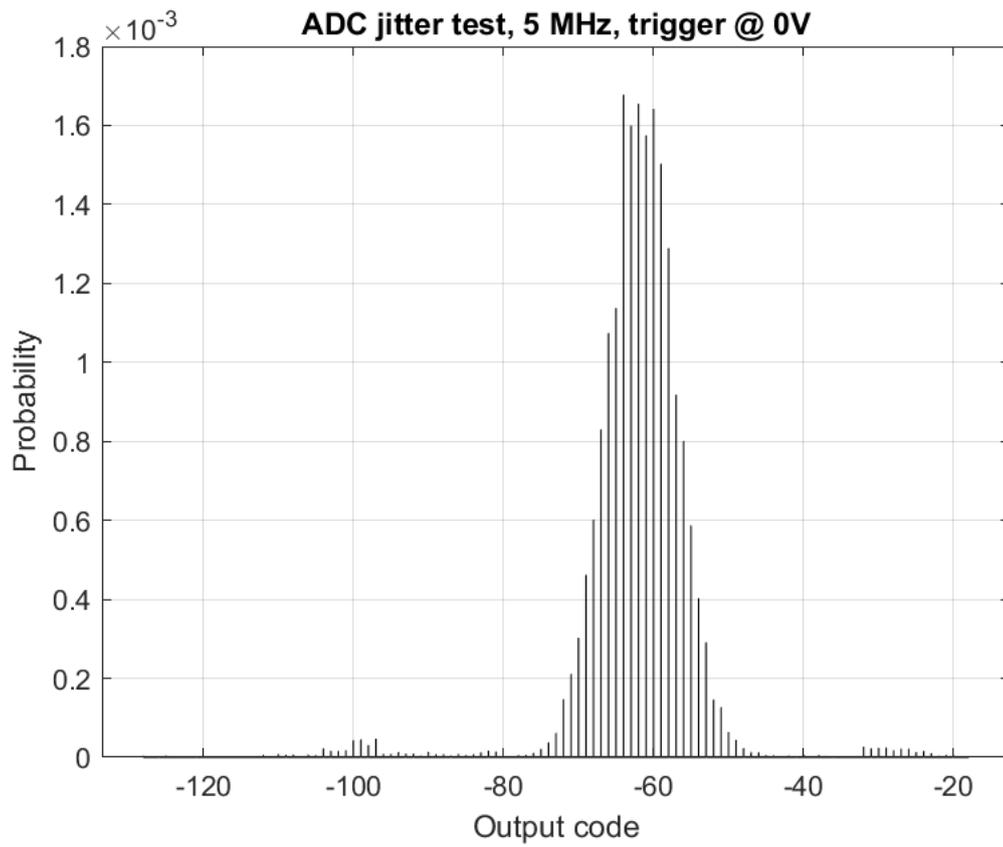


Figure 4.22: Overall jitter evaluation, 5 MHz

## 4.8 Jitter evaluation

---

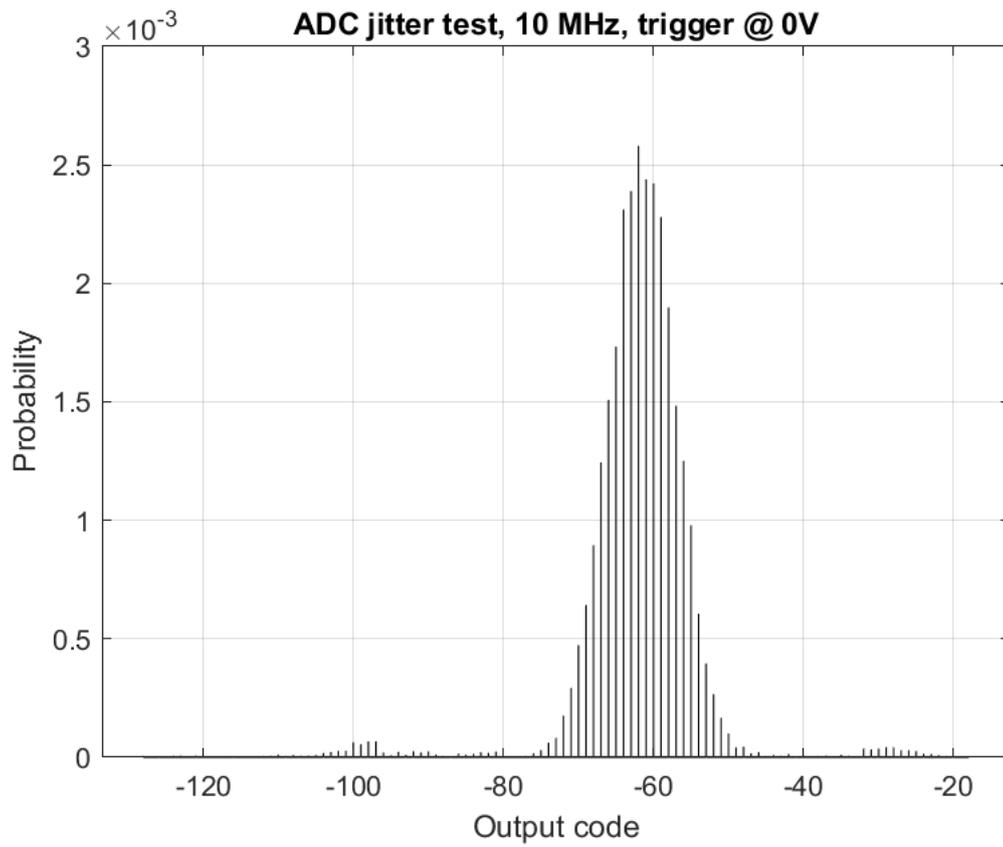


Figure 4.23: Overall jitter evaluation, 10 MHz

## 4.8 Jitter evaluation

---

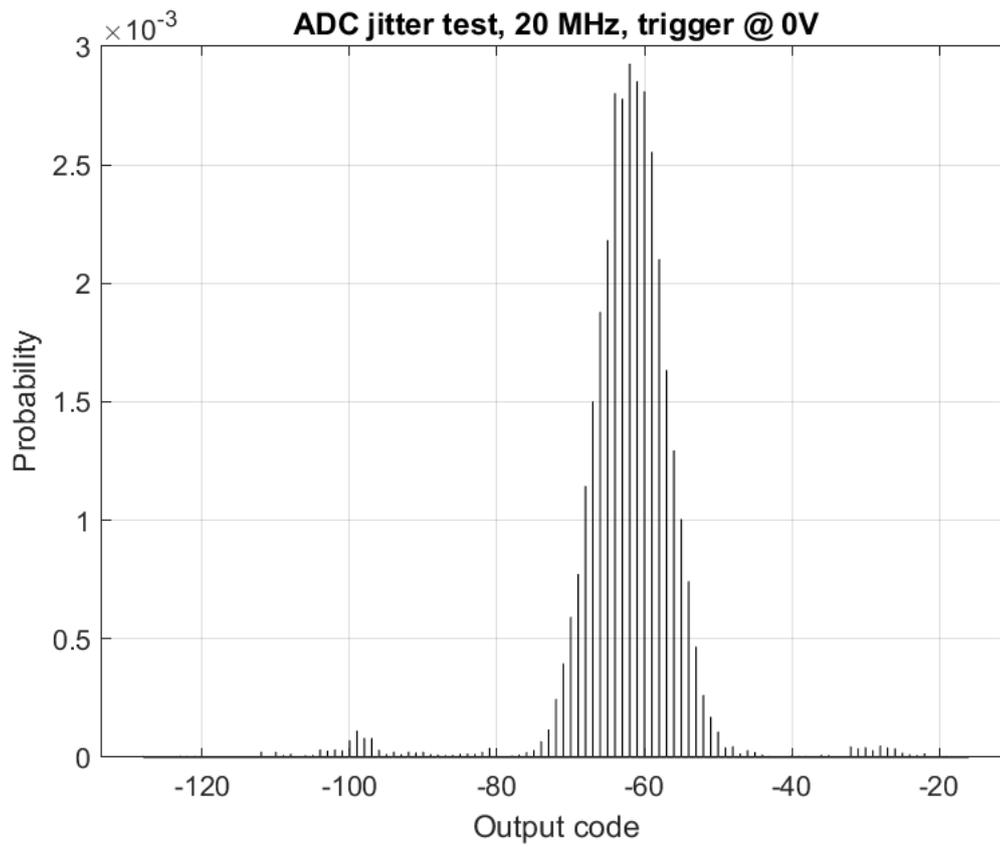


Figure 4.24: Overall jitter evaluation, 20 MHz

## 4.8 Jitter evaluation

---

Frequency [MHz]	Jitter [ps]
1	23.7
5	47.9
10	71.8
20	83.5

Table 4.3: Results of the first jitter analysis through the first hardware equipment.

Frequency [MHz]	Jitter [ps]
1	100
5	110
10	130
20	130

Table 4.4: Results of the second jitter analysis through the second hardware and the frequency meter.

The jitter measured is the result of the sum of the sample and hold unit and the clock jitter: the limit is the latter, as the S/H bandwidth declared by the datasheet is 750 MHz and thus, at the frequency object of the test, this can be considered negligible.

As it has been said, the second measurement has been done in order to understand and confirm the order of magnitude of the sample jitter: the deviation is imputable to the instrument noise bias, but verification was carried out correctly. Once confirmed, the second result has been used as worst case scenario.

## 4.9 Final SINAD and ENOB evaluations

At the end of several measurements SINAD value has been computed with formula 2.9: as it can be seen from figure 4.25 the results are in the range of tolerance for good performances, also evaluating the Effective Number of Bits (reference figure 4.26).

The tolerance has been computed for every value with the formula:

$$\delta P = 10 \log_{10} \left( \frac{P \pm \delta P}{P} \right) \quad (4.1)$$

An oscillation around  $\pm 0.5$  dB corresponds to 10% of uncertainty on SINAD value, considered as a reasonable value.

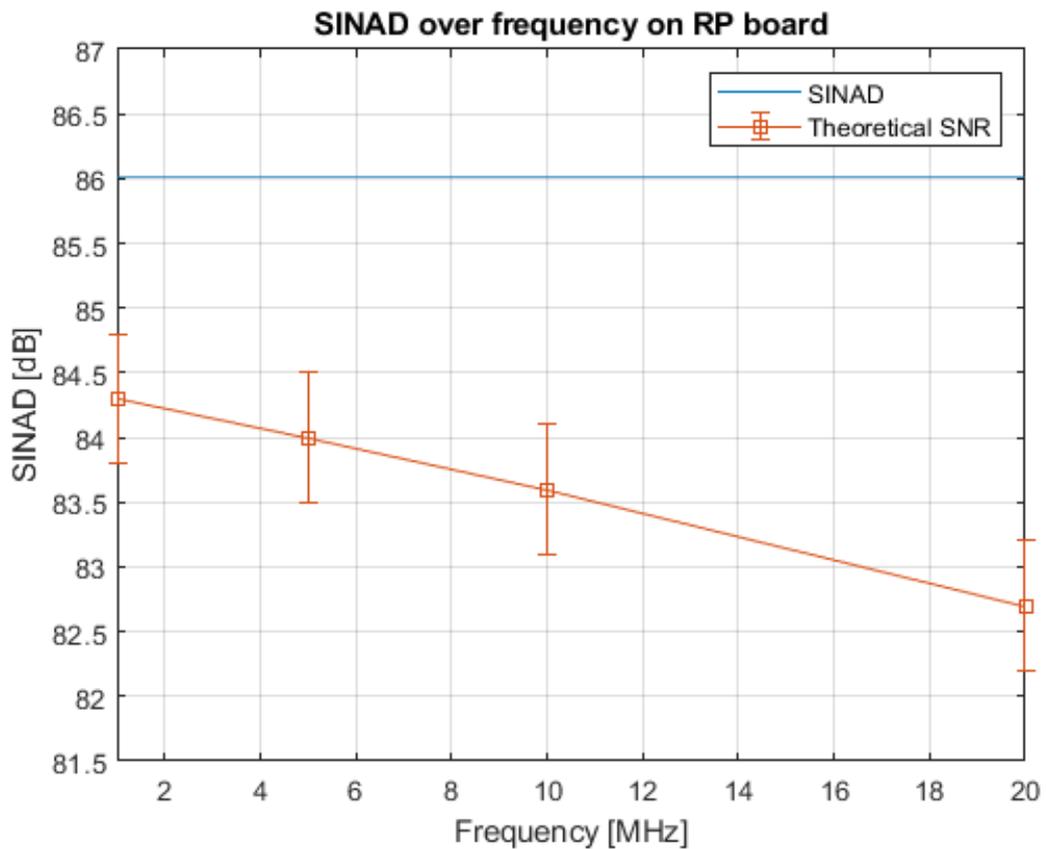


Figure 4.25: Overall SINAD evaluation.

## 4.9 Final SINAD and ENOB evaluations

---

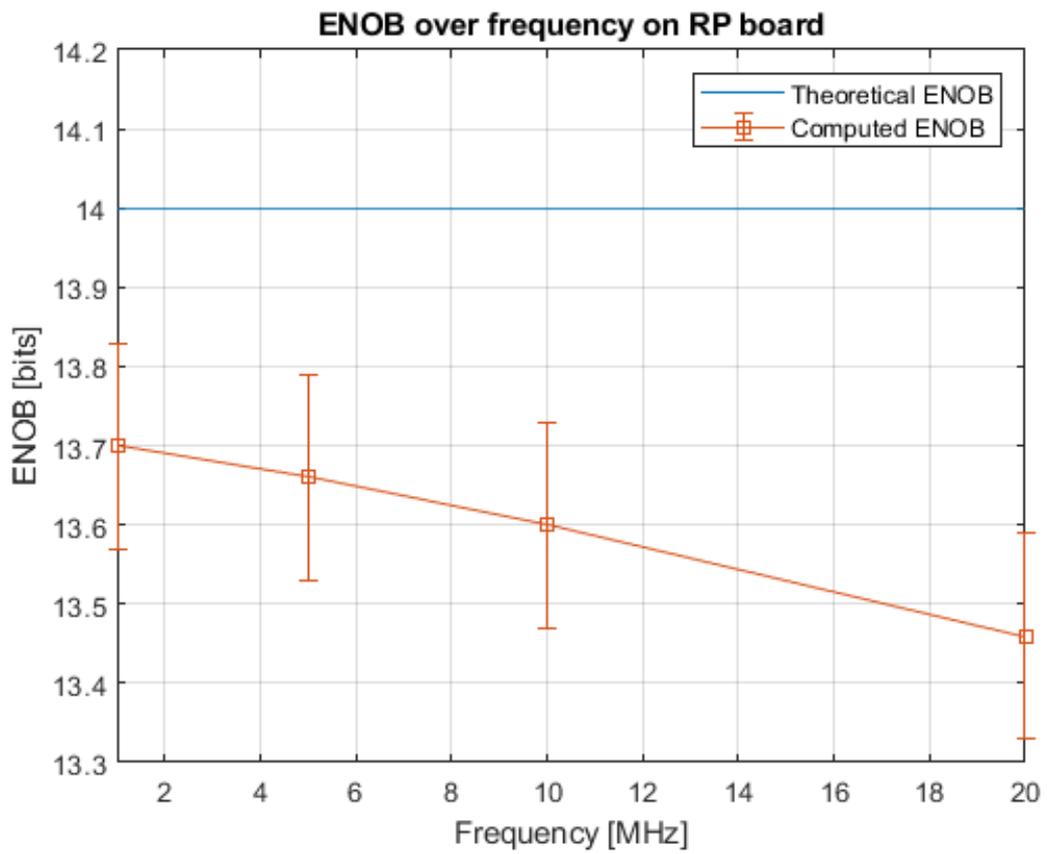


Figure 4.26: ENOB computed from the SINAD values.

## 4.9 Final SINAD and ENOB evaluations

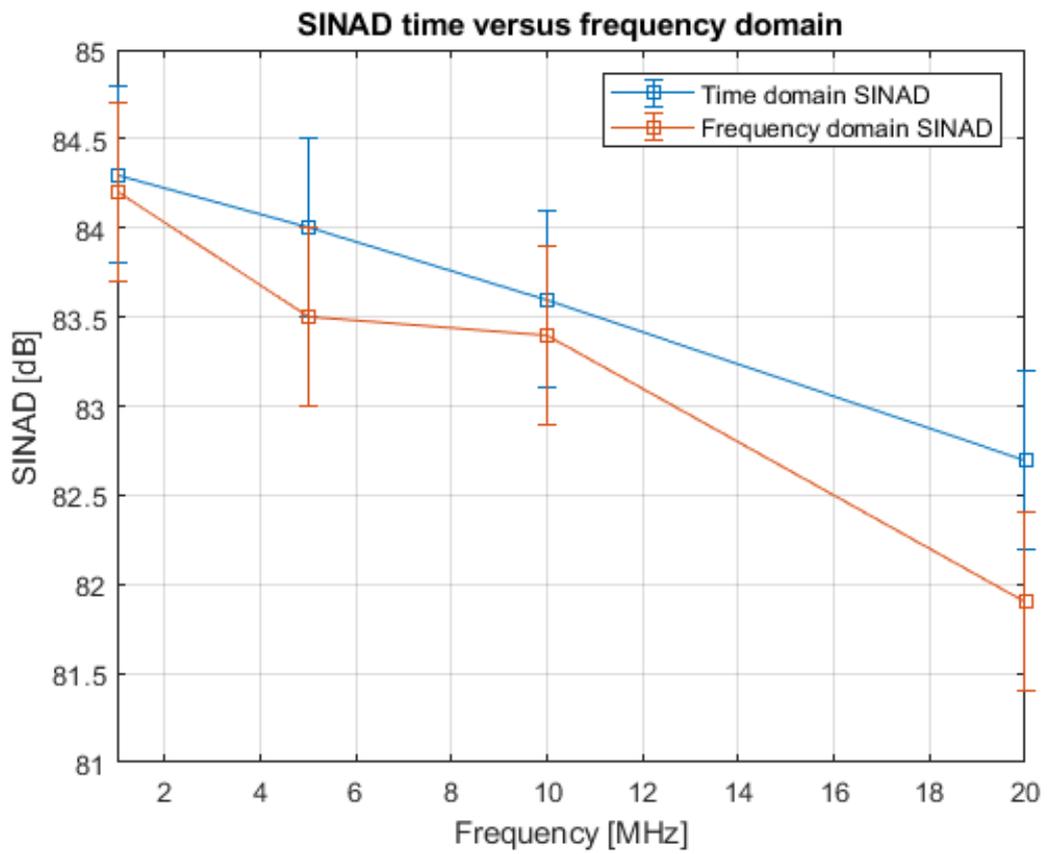


Figure 4.27: SINAD evaluation in time and frequency domain.

## 4.9 Final SINAD and ENOB evaluations

---

Frequency [MHz]	SINAD [dB]	ENOB [bits]
1	84.3	13.7
5	84	13.66
10	83.6	13.6
20	82.7	13.46

Table 4.5: Results of SINAD evaluation with ENOB values.

Time domain SINAD [dB]	Frequency domain SINAD [dB]
84.3	84.2
84	83.5
83.6	83.4
82.7	81.9

Table 4.6: Results of the second SINAD analysis in time and frequency domain.

After the analysis final values fit the results provided by the Linear Technology: testing conditions were different due to instrumentation used ( $5 V_{pp}$  instead of  $17.8 V_{pp}$ ), estimating a difference of 11 dBFS, making reasonable the results of both dissertations with the formula 3.1.

Summing up all the results presented in this chapter the limiting factors of the frequency performances in no-limit conditions are the non linearities that can not be compensated: harmonic distortion and jitter are nearly negligible in this board values. This measurement system then has been capable of weighing different figures of merit and it has been able of understanding this case's limiting factors: this experience proves that the chosen figures of merit are representative of the main noise contributions. Testing them in this way makes it possible to understand whether a board could guarantee precise specifications or not.

In conclusion Intersil formula has been verified with frequency domain analysis (reference figure 4.27) with a maximum deviation of 0.8 LSB (absolute value).

# Chapter 5

## Conclusions and future developments

The objective of this thesis was to develop a measurement sequence test for an embedded and commercial Analog to Digital converter. This work is a part of Red Pitaya characterisation for time - frequency metrology, to understand whether this board could be fit for this purpose or not. Due to the shortage of time, the sequence was not adapted to the other important part of the front-end, the Digital to Analog converter. The focus has been then moved to choose and characterise the principal figures of merit and weight them of the analog conversion. The Red Pitaya board has been proved suitable for precise electronic equipment, as the analysis on accurate signals can be carried out without noise heavy influence.

Further developments could be implemented: as well as adapt it on other board components, faster implementation could be done with Red Pitaya update in several ways:

- instead of forming data packets, direct data can be sent over Ethernet establishing a direct connection with the main workstation without using Secure Copy protocol or saving data to the Secure Disk, being limited in speed; to achieve that, AXI4 protocol has to become from Lite to Stream and high speed 64-bit port on Red Pitaya has to be enabled, maybe with configuration files provided;
- a MATLAB driver could be written, as data collection can be stored in the main computer with an algorithm, lightening the PS part of long calculations and storing operations;
- faster operations could be implemented also enabling DMA engine, trying to achieve bandwidth required from sampling saving operations.

---

The system has proven to weigh on board noise sources, to understand if they can be eventually corrected. Red Pitaya system is stable and can achieve good performances on signal generation and analysis: the next main project, after a deeper characterisation carried out by Dr. Cardenas and Dr. Calosso, is the use of this board in fibre technology as it can synchronise e.g. different laboratories over fibre diffusion of a stable and noise-free impulse.

# Bibliography

- [1] Giovanni V. Pallottino. *Il rumore elettrico. Dalla fisica alla progettazione*. Springer-Verlag Italia, 2011
- [2] Claudio E. Calosso, Enrico Rubiola. *Phase noise and jitter in digital electronics*. arXiv:1701.00094 [physics.ins-det], 2016.
- [3] National Semiconductor. *Specifying A/D and D/A converters*. AN-156, 1976.
- [4] Kent H. Lundberg. *Analog-to-Digital converter testing*. 2002.
- [5] Walt Kester. *The Good, the Bad, and the Ugly aspects of ADC input noise - Is no noise good noise?*. Analog Devices, 2008.
- [6] *Understanding noise in the Signal Chain*. [www.intersil.com](http://www.intersil.com).
- [7] *DAC and ADC Testing Fundamental*. MSIC DT Lab., Dept. of El. Eng., NYUST
- [8] Walt Kester. *Data conversion handbook, chapter 5*. 2005
- [9] Yves Langard, Jean-Luc Balat, Jacques Durand. *An improved method of ADC jitter measurement*. IEEE, 1994.
- [10] *Zynq-7000 SoC Data Sheet: Overview*. DS-190, 2018
- [11] <https://www.mentor.com/esl/vista/virtual-prototyping/zynq-virtual-prototype/>
- [12] [https://marceluda.github.io/rp\\_lock-in\\_pid/img/redpitaya\\_hardware.png](https://marceluda.github.io/rp_lock-in_pid/img/redpitaya_hardware.png)
- [13] *LTC2145-14/LTC2144-14/LTC2143-14 datasheet*. Linear Technology, 2011.
- [14] *FIFO Generator v13.1*. VIVADO design suite, 2017
- [15] *AXI4-Lite IPIF v3.0*. VIVADO design suite, 2016