POLITECNICO DI TORINO

Master's Degree in Electronic Engineering

Master's Degree Thesis

Design of an acquisition front-end robust to EMI

Supervisors
Prof. CROVETTI PAOLO STEFANO
Prof. MUSOLINO FRANCESCO

Candidate
COVOLO UMBERTO

April, 2022

to Eleonora

ABSTRACT **In this thesis, a technique for the reduction of EMI in acquisition front-end circuits is presented taking advantage of two standard, EMI sensitive amplifiers whose input stages react to radio frequencies in a complementary way. A complete micro-controller based acquisition system will be designed utilising the exposed method, based on the digital acquisition and manipulation of data. This demonstrates an increased immunity with respect to each amplifier used. The approach can be considered an alternative to the existing written solutions that, in general, require the design of specialised, non-standard, topologies with the drawback of extended time requirements and dedicated, non-conventional analysis.**
The project illustrates the current problems, then describes some state of the art solutions and, finally, exposes the design steps that have led to the final result. This result is then tested in the laboratory to illustrate the power of the technique.

Contents

Abstract **ii**

1 Introduction **1**
1.1 Typical situations 1
1.2 Target of the thesis 3
1.3 Thesis outline 4

2 Review of the state of the art **5**
2.1 EMI propagation and nonlinear effects on operational amplifiers . . 5
2.2 Nonlinear behaviour of differential pair 7
2.2.1 Small signal, high frequency analysis 9
2.2.2 Large signal models and SR 12
2.3 Possible design solutions for the EMI problem 16
2.3.1 Modified input differential pair and filtering 16
2.3.2 Cancellation topology 18
2.3.3 Advanced topologies 21
2.3.4 Complementary topologies 24
2.3.5 Comparisons 25
2.4 Development of complementary based input solutions in FE 27

3 The presented solution **30**
3.1 Theoretical base 30
3.2 The double amplifier board and the filter 32
3.3 The micro-controller board 42
3.4 The ADuC7061 45
3.4.1 The analog to digital converter 46
3.4.2 The UART interface 51
3.4.3 The memory and the control registers 52
3.5 The ADuC7061 configurations and the overall algorithm 54
3.5.1 Configuration of the oscillator and the active devices 56

3.5.2 The UART . . . 56
3.5.3 The ADC setting parameters . . . 57
3.6 The core algorithm . . . 59
3.7 The serial transfer to PC . . . 65
3.8 Functionality tests . . . 66

4 Measurement of the tested configuration 68
4.1 The configuration used for measurement, connections and the procedure . . . 68
4.2 Parameters definition . . . 73
4.3 Offset calibration . . . 74
4.4 Alpha estimation . . . 76
4.5 System response without EMI . . . 78
4.6 Induced offset measurements . . . 82
4.6.1 Alpha redefinition . . . 85
4.7 CW EMI measurement . . . 89
4.7.1 RF AM square modulated measurements . . . 89
4.8 RF AM sinusoidal modulated measurements . . . 100
4.9 RF AM exp modulated measurements . . . 103
4.10 SFDR and SNDR measurements . . . 107
4.11 Effects of frequency on the signal with and without the LPF . . . 115
4.11.1 Internal ADC digital filter . . . 115
4.11.2 The additional low pass filter . . . 119

5 Conclusions 121

6 Appendix 123
6.1 Figures . . . 124
6.1.1 RF 1kHz square-wave modulated plots 0-9dBm . . . 125
6.1.2 RF 400Hz sine-wave modulated plots at 7dBm . . . 130
6.1.3 RF 100Hz exponential modulated plots at 9dBm . . . 133
6.1.4 RF 1kHz exponential modulated plots at 0-9dBm . . . 136
6.2 SFDR and SNDR for a 1KHz square wave modulating signal . . . 141
6.3 The ADUC program . . . 143
6.3.1 The standard program . . . 143
6.3.2 The measurements program . . . 152
6.4 Matlab scripts . . . 162
6.4.1 Laboratory acquisition program . . . 162
6.4.2 Standard acquisition program . . . 164
6.4.3 Plot and shift program . . . 165
6.4.4 Offset plotting . . . 168

6.4.5 Offset vs power plotting, alpha refining 170
6.4.6 Offset vs frequency plotting, alpha refining 174
6.4.7 FFT computation single channels 178
6.4.8 FFT computation all channels 181
6.4.9 Alpha to fixed point conversion 183
6.4.10 Alpha evaluation 184
6.4.11 Alpha computation with minimization 185

Bibliography **185**

CHAPTER 1

Introduction

Operational amplifiers, (opamps), are the basic building block for the design of modern analog sophisticated circuits. One or more of them can be employed in a design. Beyond their power all of them are susceptible to the interference of radio frequency, RF. Indeed, opamps, whose inputs are subjected to spurious EMI, show output disturbances in the form of a voltage offset shift in the presence of a continuous wave, CW, RF signal, and in the form of a voltage proportional to the demodulated RF. As will be shown later, (see also [1, 2, 3, 4]), EMI can generally be modulated by low frequency signals that, due to demodulation, can be found upon output of the stage. In both situations, an error is superimposed onto the nominal signal generated from the RF input.

Looking at the analog world, amplifiers are applied to the front end of signal acquisition systems as conditioning amplifiers and in filters for the acquisition of information from sensors and from transducers. Taking into account that the acquired signals have, in general, reduced amplitude, the distortions that arise can affect all the following stages. In some cases the latter error can even completely corrupt the base band signal involved or bring the device or the subsequent into saturation.

1.1 Typical situations

With the spread of wireless communication systems in different and increasing bands of interest (mobile phones, PC networks, Bluetooth, smart gadgets. etc) and with the almost total substitution of linear analog power supplies with switching power supplies, the environment where front end circuits must operate has become more and more polluted by RF interference. This has brought about an increase in the importance of susceptibility to EMI that cannot be neglected. Moreover, considering that the power supply voltage of the integrated circuits,

ICs, has been decreased progressively from 5V to 3.3V and even lower voltages, the errors produced by EMI could be intolerable. Indeed, in cases of low power devices, the unwanted signal added to the input of the conditioning amplifiers can have a comparable or even higher amplitude with respect to the useful signal being processed. For these reasons, it has become important to find solutions to overcome the problem and to make opamps less sensitive to outside interference.

In a typical front-end system (figure 1.1),

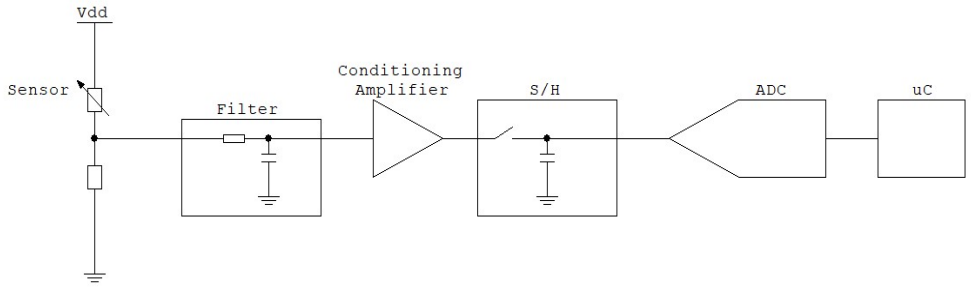


Figure 1.1: Common front end

composed of an analog stage followed by a digital acquisition stage, one could find the generator of useful signal followed by a low pass filter used to keep the input within the band of the subsequent amplifiers and reduce the aliasing phenomenon, then one or more amplification stages (that can also operate as a filter), and an analog to digital converter. The low pass filters in front of an amplifier which are supposed to be a barrier to high frequency, typically, are not and due to the unwanted presence of parasitics. The RF signal can therefore bypass the filter and reach the sensitive devices that follow. Furthermore, the EMI can couple to traces of the printed circuit board (PCB), the ground plane, or, also, to the pins of the integrated circuits and induce currents and voltages at the inputs of the amplifiers or at the power supply pins. Methods to reduce the unwanted effects, have to be considered, especially for applications where the environment is noisy or has strong RF pollution. Solutions like the introduction of filters, layout modifications, variation of the operating frequency, and dedicated shielding, can bring with them costs and difficulties and may not be a cheap way to solve the problem. At the same time, there may be conflicts with the need to use plastic materials instead of metal. The use of shielded cables or filtered connectors can also be expensive or incompatible with the space limitations required by the application. So, the design must take into account EMI effects from the beginning, and a solution should also be found also in the early stages of the project.

As will be shown below, operational amplifiers and feedback amplifiers especially (first of all the voltage follower configuration [4]), are particularly susceptible to EMI at their inputs. Different studies have been done in order to describe their susceptibility, since their behaviour is highly influenced by the applied RF and because of their predominant use in modern systems. In the literature, solutions have been proposed with the aim of having opamps more robust to EMI. These require modifications of the the amplifier circuit layout and require, therefore, a non standard design, an ad hoc circuit to limit the adverse effects of RF. This non-standard approach can, in turn, increase the time required for the project and can require non standard analysis and project guidelines. Solutions, indeed, have been presented [5, 6, 7, 8, 9, 10, 11] by changing the topology of the circuit such as non-standard input differential stages or complementary input stages, in order to compensate with opposed effects on different types of circuits (for example, a pMOS and an nMOS differential pair see [10] and [12]), or by adopting some sort of filtering effect at the input. However, each of these solutions has a negative impact, since they require a specific, non-standard design that could also be influenced by process variations and poorly controlled parasitics and may even be technology dependent. Moreover, these techniques can affect the performance of the amplifier itself (for example, varying the common mode input range, CMIR, see [10]) modifying what the opamp originally was. Nonetheless, some available methods do not adapt well to nano-scale integration (for example, due to the limited gate length [12]).

1.2 Target of the thesis

In this thesis, a new technique aiming to reduce errors caused by EMI in front-end acquisition systems will be described and validated through a complete system design. In particular, this new approach is based on the use of simple EMI-sensitive, off-the-shelf operational amplifiers, with the aim of obtaining an overall reduction of the radio frequency induced error. The technique takes advantage of two operational amplifier placed in parallel, that show complementary input behaviours (an nMOS (NPN BJTs) and a pMOS (PNP BJTs) input), to which the useful signal and the RF are applied. Because of the complementary behaviour, two signals corrupted by opposed sign errors are obtained at the output of the opamps. Through digital acquisition and with specific post-processing of the samples, it is possible to use this effect to reduce the disturbances on the useful signal.

The thesis is focused on the development of the digital acquisition, of the elaboration stage, and on the testing of the overall effectiveness of the illustrated techniques. In particular, it is the design of a system based on the ADuC7061 micro-controller demo-board, the writing of the firmware that carries out the re-

quired theory calculations on the acquired data, and the implementation of the communication protocol from the board to the PC, where the data are analysed through their frequency spectrum. The complete system is then tested to confirm the effectiveness of the theory.

1.3 Thesis outline

The work starts by describing how the RF can affect the operational amplifiers output and by introducing the results obtained in different studies. Then, it presents some of the possible solutions that can be found in the literature to make an amplifier more EMI immune, concluding with the description of the theory behind this new approach. The technique is then tested to prove its validity with measurements.

In Chapter 2, the current state of art is reviewed, highlighting how the nonlinearities can induce errors in the amplifier output and introducing the theory behind this work. Then, some solutions proposed by the literature are shown, including the studies that are the groundwork of this thesis and that take advantage of the use of complementary susceptible stages to obtain the EMI induced error cancellation.

In Chapter 3, the new technique is described in more detail, and the hardware used to perform the acquisition is shown. It begins with the description of the board that hosts the two operational amplifiers and continues with the acquisition board used to perform the analog-to-digital, A/C, conversion and the computations required. It then describes in depth, the different configurations used for the ADuC7061 and explains the micro-controller functionalities, its firmware and the way it communicates with Matlab.

In Chapter 4, the complete test-bench design is used to perform some measurements and to evaluate the results. Furthermore, the latter are compared with previous results obtained without using digital dedicated-elaboration (see [13]).

In Chapter 5 the conclusion of this work is discussed.

CHAPTER 2

Review of the state of the art

In this chapter, the mechanisms behind the EMI-induced error in opamps and their modelling will be revised. Then, some solutions reported in study articles will be discussed, giving insight into the current state of the art.

2.1 EMI propagation and nonlinear effects on operational amplifiers

Looking at the literature, it is possible to find various studies and simulations about how the EMI induced spurious error can be characterized. These studies allow an understanding of the sources of EMI propagation and obtain methods to manage the problem. This, in turn, provides some guidelines to follow, in order to reduce the susceptibility. Moreover, a link between the amplitude of RF at the inputs of the opamps and the induced output offset shift and in band disturbances is provided ([2], [12], [14] [15], [16], [17]). Easy to use closed-form expressions are proposed to help the layout designers make correct choices to reduce the susceptibility, showing the dependencies of the corrupted output upon the RF input voltage, the parasitics and the design parameters.

The effects of EMI in operational amplifiers, in general, are related to the nonlinear behaviour of the overall circuits themselves. However, the principal cause of distortions can be found in the differential pair stage ([12], [16], [18]) and not in the successive stages that can be considered linear. Commonly, an opamp is, indeed, composed of one input transconductance stage, the differential one stage, and a cascade of one or more amplification stages (see figure 2.11). A compensation network, where required for gain stability in feedback, limits the bandwidth to some MegaHertz. This compensation, generally, is applied to the output of the first differential pair. In this way, the propagation of RF is almost

stopped at the end of the first stage without propagating it through the output of the overall amplifier. The next stages, therefore, do not contribute to distortion unless a very high amplitude RF changes the operating point of the these stages. On the other hand, the differential pair can generate inter-modulation components that drop on the useful band of the amplifier and, thus, are amplified from the stages that follow untill output. In particular (see [1, 12, 15, 16, 17]), a CW RF at input generates a DC average component in the pair that is brought to the output in the form of an offset that is not directly linked to asymmetries of the pair but, instead, to the distortion generated by it.

The typical approach followed by the literature to arrive at the computation of the EMI effects assume, indeed, that the stages following the input differential pair are linear in band. So, an high transimpedance is considered in the useful frequencies and a negligible one above those frequencies. This assumption can be considered valid for all the approaches. Both [12] and [19] show how nonlinearities give rise to the EMI effects. Summarizing the results, the susceptibility to EMI is a consequence of the harmonic distortion that is present on circuits that are intended to operate as linear (see for example the MOS characteristic in saturation $i_d = \frac{1}{2}\beta_n[(v_{gs} - V_{TH})^2(1 + \lambda v_{ds})$ that is intrinsically is non-linear). At a first approximation level, in fact, the effects of nonlinearities are neglected, considering only the small signal linear equivalent when, instead these effects are always present and must be considered.

Nonlinear systems can generate spectral components that follow the so called harmonic distortion. This implies a generation of intermodulation products [16] that create spectral components as a linear combination of the input frequencies. The common way to represent the transfer function of these systems under consideration is a power series expansion (formula 2.1) where the following are present: a bias component (f_0), a linear term, the small signal components (x dependent through a coefficient), and a set of distortion terms where the input appears as a function of x^i where i is greater than or equal to 2. These are called distortion of i-order.

$$F(x) = f_0 + a_1 x + a_2 x^2 + a_3 x^3 + ... \tag{2.1}$$

In theory, it is furthermore possible to demonstrate that an i-order distortion can generate a set of spectral components that depend on that order. So distortions can be generated with a frequency that is a linear combination of the input harmonics, (f_j), so that the sum of the frequency coefficients are exactly the i value.

$$f = b_1 f_1 + b_2 f_2 + b_3 f_3 + + b_i f_j \tag{2.2}$$

with $b_i \in \mathbf{Z}$.

$$\sum_{j=1}^{+\infty} |b_j| = i \tag{2.3}$$

A third-order distortion can make unwanted disturbances at the signal frequency and at three times the signal frequency (for example, a signal at $\omega_s = \omega_s + \omega_{RF} - \omega_{RF}$, frequency that drop in the signal band and whose amplitude depends on the RF amplitude). Second-order distortion can, instead, produce a distortion with frequency equal to zero or to double that of the signal, (linked to the RF component as shown here $\omega_{RF} - \omega_{RF} = 0 = DC$, a component that is related to RF signal amplitude but is present at zero frequency). The latter is one of the most important issue, since it generates a variation at DC. Similar considerations can be made with a modulated EMI at the input, where the demodulated signal in band can be found at the output of the non-linear system . In [4], indeed, various experiments have been conducted showing that the DC component at output is exactly an even order harmonic that depends on the behaviour of the amplifier. What this reveals about harmonic distortion, and in particular about the even order distortion, explains why the DC offset and the demodulated RF signal are present at the output of an opamp.

Of note is that a fully balanced circuit, as the differential pair should be, where the output is a function of the input signal subtracted by a function of the complement of the same signal ($Y(x) = g(x) - g(-x)$) should not generate even distortions, since it is a odd function. Considering the non-linearities this effect is not true for opamps.

Another effect that is relevant is taken into consideration in [4, 20, 21]. In particular, these works is describe the involvement of the non-symmetric slew rate, SR, in the emergence of the DC offset for moderately high RF frequencies. In this case the RF must propagate until the output of the opamp and the unequal positive and negative SR do not compensate making a DC voltage come out.

In summary, two are the most relevant effects that can be observed and are responsible for the EMI effects in the operational amplifier. The first is the rectification due to the mixing effect related to the non-linearities, and the second is the effect of the asymmetric SR that makes a DC shift arise in the event of propagation until the output of the RF. Both of these effects will be discussed next.

2.2 Nonlinear behaviour of differential pair

In a front-end system, the printed circuit board (PCB), the traces, the wires and, also, the pins of the devices can couple with electromagnetic interference and generate voltages and currents at the inputs of the circuit. The concern about the operational amplifier is due to the fact that it is one of the most sensitive

components and that, at the same time, it is employed in multiple applications from automotive to avionics. These environments are, often, highly polluted. For example, in aircraft wireless communications can involve a wide electromagnetic spectrum ranging from kHz to various GHz, and the susceptibility can generate serious problems, (some insight can be found in [14], where the problem of RF presence at the power supply pins is also taken into consideration).

In the literature, different studies were conducted with the main aim of finding an opamp model (although knowledge of the internal structure is required, it can be useful to obtain a design guideline) or a macro model [1] (that does not require the details of internal structure and parasitics and can, therefore be used in practical applications) that could explain and predict the observed behaviour of the circuit during laboratory tests done on a real amplifier. Realistic models are, indeed, required to make the analysis of the complex circuit as simple as possible. Macro-models, instead, can reduce the simulation time with respect to a complete circuit model and, at the same, can point out the link between EMI and the output without knowing the internal circuit structure (see [1]).

Looking at these studies ([1, 13, 15, 16, 17, 18, 19, 20, 21, 22]), it is possible to find various solutions that have been tried to explain how the EMI is propagating and how they influence the behaviour of operational amplifiers in different configurations. In summary, the results have highlighted a susceptibility of the amplifier in feedback configuration that demodulates through even-order distortions and asymmetries of the slew rate (SR), the RF at its input. The amplifier does give indeed, an offset DC shift as output in the case of CW RF signal at input and, more commonly, an error as a superimposed in-band signal. These results have also introduced an important classification of the voltage follower configuration, where the gain is +1, considering it to be the worst case for EMI susceptibility. The latter conclusion allows consideration of this topology as the reference case for this and other studies, permitting easy evaluation of the RF common mode and the differential mode. (see [12]).

In [3, 20], some models of the opamp are presented and compared with test results on commercial models. As observed in these studies and in the measurements in chapter 4, the amplitude of the resulting offset starts at low frequencies and increases to a maximum before, finally, reducing at higher frequencies, see figure 2.1 and 2.2. In the above cases, the impedance of the RF source, in addition to parasitics at the input also caused by packaging and connection configurations (see also [23]), behaves as a low-pass filter that makes the EMI-induced voltage decrease for high frequencies, in contrast to what was obtained in the on-chip test experiments (see [3, 16, 17]).

Of note is that, due to non-linearities, the application of a CW RF sine-wave with no DC bias at the input generates an output with a different offset with

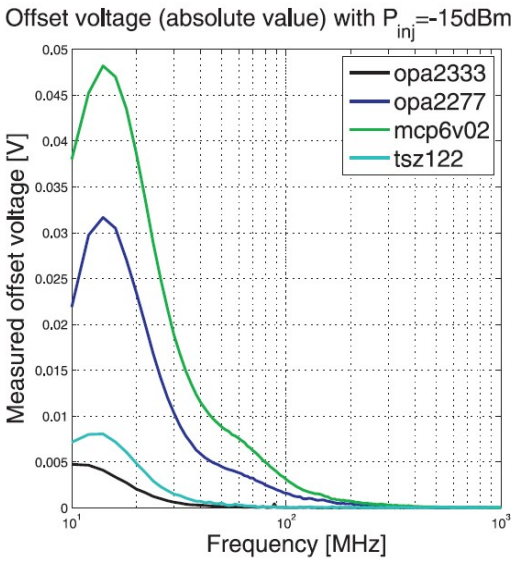


Figure 2.1: Offset in modulus vs frequency for different types of amplifiers, taken from [20]

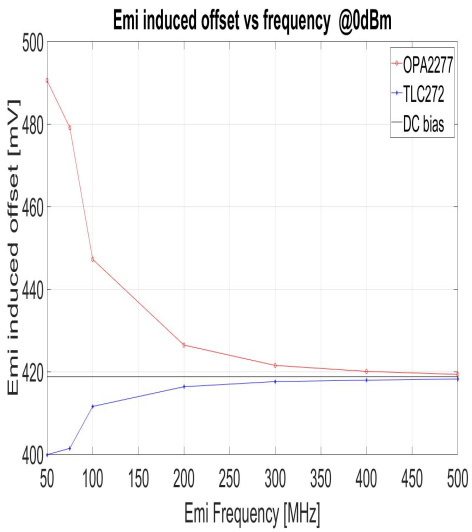


Figure 2.2: Offset vs frequency for 0dBm incident EMI and for different types of amplifiers, from measurements

respect to the DC bias point obtained without EMI. This additionally highlights the nonlinear nature of the offset. Indeed, it is worth remembering that a linear time-invariant system with a sinusoidal input still gives a sinusoidal signal as output with zero mean value. The DC offset that comes out is, thus, to be attributed to nonlinear mechanisms (see also [24]).

Considering the illustrated effects, the variation of the output DC point cannot be underestimated, because its variation can be a relevant issue for the successive devices and, even, for the amplifier itself that can go into saturation (looking at the figure 2.2 a relevant offset shift is induced at 50MHz).

2.2.1 Small signal, high frequency analysis

In [12], thanks to the Volterra series expansion, in [15], and in [1, 16, 17, 19, 22], the importance of the variation of the bias current of the MOS pair (similar results for BJTs) on the input (output) DC offset variation is demonstrated. These studies show that an input RF common mode, (CM), variation, in combination with a simultaneous RF differential mode, (DM), variation, are responsible for a mixing effect (an even order distortion) that demodulates the input RF. This behaviour is also obtained in the case of perfectly symmetrical differential pairs and can even predominate

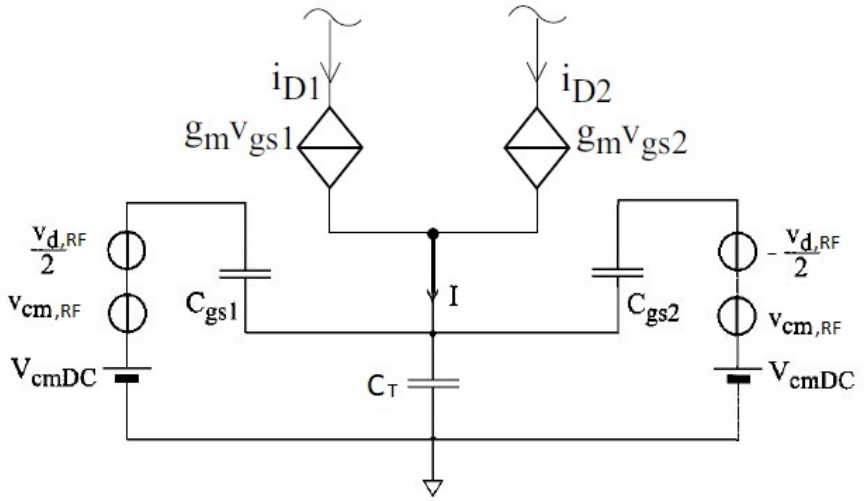


Figure 2.3: Rf induced current variation

over the effect of the asymmetries.

The mixing issue clearly emerges if a second order Taylor expansion of the expression of the differential current, $i_d = i_{d1} - i_{d2}$, (see eq. 2.4 and figure 2.3, where $\beta = \frac{\mu C_{ox}}{2}\frac{W}{L}$, is the technological parameter) is performed. Here, contrary to what is done when the circuit is analysed as a perfectly linear differential circuit, the bias current is not considered to be constant. The i_d is, as such dependent on both the differential voltage v_d and the non-constant bias current (i_{bias}). This assumption permits highlighting of the even-order distortions that would not emerge if i_d were considered dependent only on v_d (the i_d is, indeed, an odd function with respect to the single differential voltage v_d only).

$$i_d = \begin{cases} -I_{bias} & V_d \leq -\sqrt{\frac{I_{bias}}{\beta}} \\ V_d\sqrt{2\beta I_{bias}}\sqrt{1 - \frac{\beta * V_d^2}{2I_{bias}}} & |V_d| \leq \sqrt{\frac{I_{bias}}{\beta}} \\ +I_{bias} & V_d \geq \sqrt{\frac{I_{bias}}{\beta}} \end{cases} \tag{2.4}$$

Computing the Taylor expansion (for a bias point $V_d = 0$ and $I_{bias} = I_0$) and rearranging the terms, what is obtained is equation 2.5, (a similar expression that still depends on common mode and differential mode is obtained in [22] for the whole opamp)

$$i_d^{(2)} = g_m v_d + g_p v_d i_b \quad \text{from [15]} \tag{2.5}$$

where the variables are considered to be as RF small-signal variations and the trans-conductance $g_m = \sqrt{2\beta I_0}$ and $g_p = \sqrt{\frac{\beta}{2I_0}}$ (the result is still valid for bipolar

transistors considering the differential current $i_D = \alpha_f Itanh(\frac{V_d}{2V_T})$ and so the $g_m = \frac{\alpha_f I_0}{4V_T}$ and $g_p = \frac{\alpha_f}{4V_T}$).

The fluctuation of the bias current, i_b, in eq. 2.5, in turn depends on the variation of the RF common mode, giving a dependence of $i_d^{(2)}$ on the product of the differential (from v_d) and the common mode component of RF (from i_b). The current variation i_b is, in fact, linked to $v_{cm,RF}$ (see [15] where $i_b(s) = v_{cm,RF}Y(s)$ is computed in the hypothesis of perfect transistor matching, and [24]) through the finite impedance of the tail current source. Aside from the latter, at RF it further reduces due to the parasitic capacitance $C_t = C_{db} + C_{Al}$, (see figures 2.3 and 2.4 where the C_{db} is the reverse junction drain to body capacitance and C_{Al} is the P-Well to n-Isolation capacitance of a typical twin-tub nMOS, which arise if the transistor has been designed in a well to limit the body effect).

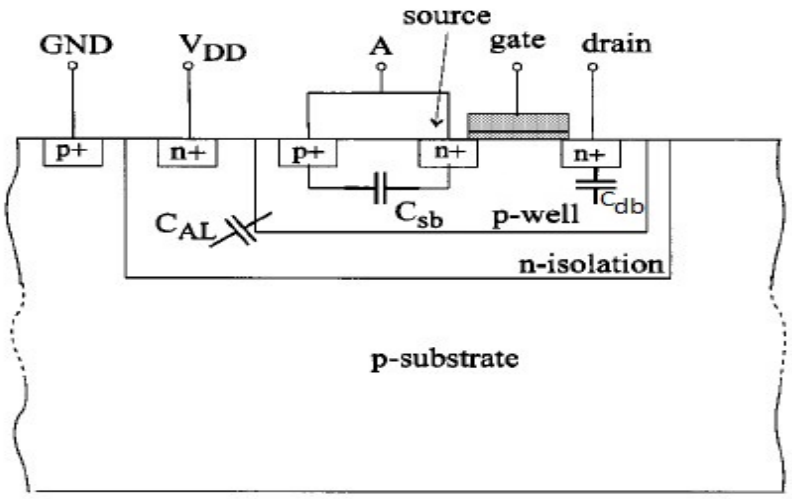


Figure 2.4: Cross-section of a device from [15]

Overall, the simultaneous variation of v_d and i_b determine an even-order distortion, i.e. for isofrequency variation of v_d and i_b a DC offset is generated on the differential current (eq. 2.5, in addition to the components directly linked to v_d, i.e $v_d \cdot g_m$) that could not be highlighted considering the bias current as fixed (see [15]). This current offset could, then, be considered at the input or at the output as a voltage shift ($\Delta V_{off-In} = \frac{\Delta I_d}{g_m}$, with g_m the transconductance of the stage).

The offset, in this exposed situation, must be computed by evaluating the RF common mode and differential mode at the input for a specific feedback configuration, knowing the circuit parameters. As noticed in [15], the evaluation of the RF common mode and differential voltage requires, in general, information about

all the components of the circuit and the applied V_{RF}, the differential amplification and the common mode rejection ratio. However, in the hypothesis of passive feedback and high frequencies, (the frequency is above the first stage cutoff),0 the amplitude of the RF at the output and returning to the input through the feedback is negligible and considered zero. In this way the differential and common mode can be easily computed considering $v_{cm} \approx v_{in}/2$ and $v_d \approx v^+ - v^- = v^+ = v_{in}$ since $v^- \approx 0$.

The limit of this exposed approach is its validity only for the application of small signal, and for the high frequency condition. On the other hand, it shows clearly how the EMI can have effects on the differential pair. Further studies have been done to eliminate these limitations and find a frequency-dependent and large signal model (see next).

2.2.2 Large signal models and SR

In [16], following the technique of [20] valid for small signal amplitudes, (based on this [12] and [17]), a rather different method to evaluate the disturbances induced by EMI in a negative feedback configuration, valid also for large signal, is presented, and shows a link to design parameters and parasitics. The method exposed in [20] is, indeed, adopted, and modified to be valid in general, however, a closed form expression for the offset is not provided. The link between the error and the RF peak voltage of the disturbance obtained in [20], has been proven to have a quadratic form, but the validity of this result is limited to the case of small signal input ($\Delta V_o ff = \frac{\Delta \bar{i}_D}{g_m} = K V^2_{RF,peak}$). In this study, the transistors are, indeed, not driven out of saturation and when this happens, the deduced relations are no longer valid (similar considerations and results have been obtained for BJTs input stage where the relation has a similar form). The [16] has overcome this limitation.

In contrast to the approximated computation, like the above studies and with respect to the Volterra method shown in [12] and [22], that diverges for large amplitude input, the presented model in [16] is also valid in cases of large signals where transistors can go out of saturation and can turn off. Hence, a useful method that can be applied in either a weak or strong nonlinear situations is presented. The benefit of this last approach is that the demodulated RF at output is connected to the differential drain current mean value and can also be computed in case the transistors are switched off.

The article shows that the drain current variations of the differential pair, directly induced by RF, are out of the cutoff frequency of the amplifier and, so, are not amplified by the stages that follow. Instead, what remains significant is the variation of the drain current mean value, $\Delta i_d = \bar{i}_{d1} - \bar{i}_{d2}$, that is inside the band of the amplifier and, thus, is amplified until the output (the individual mean values

are different because of differences in the amplitudes and phase of the applied $V_{RF\pm}$ at the input pins). The EMI, at the inputs, tries to impose a mean current value different than zero, but the feedback compensates for this effect with an output offset. The DC value is modified such as a way to compensate for the generated Δi_d and to bring it to zero. To be noticed is that a null variation of the mean differential current is obtained only if I_{bias} is considered constant (the current mean value for both transistors in this case is $\frac{I_{bias}}{2}$). Once the input voltages are evaluated, the gate to source RF voltages are computed. From these, the current mean values ($\bar{i}_{d1}$ and $\bar{i}_{d2}$) are obtained and, in turn, the mean value of the DC voltages at the inputs. Finally, for difference is determined the offset voltage generated by RF (see [16]). In case of signals that make the input transistors switch, the formulas are modified to take into account, in the current mean values computation, the off period and then obtain the RF-induced input offset voltage.

The study discussed confirms that CW RF superimposed to the input of the amplifier in a feedback configuration can induce an input offset that depends on the amplitude and the frequency of the EMI and that the distortion is mainly due to the differential pair. Moreover, it is proven that the opamp in a negative feedback configuration demodulates the input RF and, instead, in case of RF with constant amplitude but with a frequency variation, what is obtained is a constant output offset. This can, therefore, include general situations, for example a common AM modulated RF signal whose effect at the amplifier output is a signal that represents the demodulation of the input EMI in band.

A more recent large signal model has been proposed by [12] and revisited by [17] for a negative feedback amplifier. Here, the computation of the offset shift is obtained with an approach similar to that of the study discussed above, but providing a closed formula for the induced error. The article introduces a new exponential model for the MOS, (eq. 2.6), valid in saturation and in subthreshold (I_{D0} and v_F are model parameters that depend on the bias point of the system V_{GS} and I_D).

$$i_D = I_{D0} e^{\frac{V_{GS}}{V_F}} \tag{2.6}$$

The latter overcomes the assumption of high frequency RF, (giving a $V_{gs1,2}$ expression that depends on the EMI frequency through two α_i coefficients), and then evaluates the mean drain currents variation.

Like previous studies, the conclusion is that the feedback configuration gives rise to an input offset that compensates for the variation of the current mean values caused by the $V_{gs1,2-RF}$ superimposed at the inputs, ($V_{gs1,2} = V_{gs1,2,DC} + |V_{gs1,2-RF}|cos(\omega t)$), trying to maintaining it at zero. The offset is, indeed, obtained by equating the two mean differential currents ($\bar{i}_{d1} = \bar{i}_{d2}$) and computing $\Delta V_{off} = V_{gs1,dc} - V_{gs2,dc}$, obtaining equation 2.7 (where I_0 is the modified Bessel function of

the first kind of order zero).

$$\Delta V_{off} = V_F ln \frac{I_0\left(\frac{|V_{gs2,RF}|}{V_F}\right)}{I_0\left(\frac{|V_{gs1,RF}|}{V_F}\right)} \tag{2.7}$$

This model, although it comes to similar conclusions as those seen previously, is important to mention, since it goes beyond the limitations of small signal RF input and high frequency assumption and obtains, a new, valid input offset, EMI-frequency dependent model for large signal.

In [4, 20, 21], another concurrent effect of EMI on operational amplifiers has been pointed out. In these articles a description and some simulations have been done to show how the asymmetries of slew rate can determine the DC offset. In cases with symmetric positive and negative slew rates, SR, no DC offset shift is present, because the rising and falling transients compensate each other. When, instead, as is common, the positive and the negative SRs are different, a DC component arises whose value depends on the fact that one of the two SRs prevails with respect to the other (see figure 2.5). A possible cause of the asymmetric SR is the channel length modulation of the current source transistor in the differential pair that, in the event of a positive input step, causes an increase in the bias current and subsequently, a higher slew rate. A complementary effect is obtained in the case of a negative step.

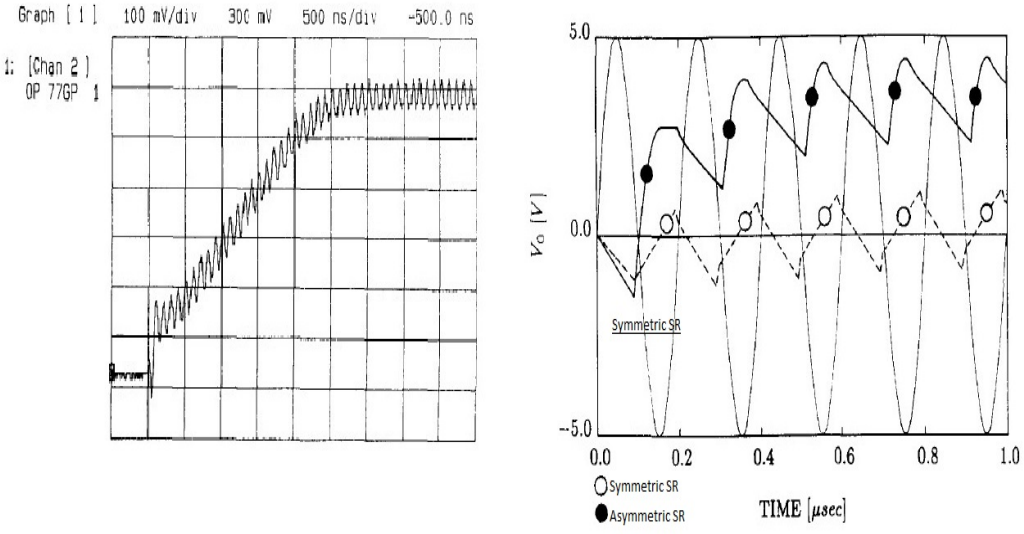


Figure 2.5: Offset induced by SR from [4]

As noticed in the articles, the SR mechanism is quite important from low to medium RF frequencies, since the RF must propagate until the output of the opamp without being strongly attenuated. For higher frequencies at the input of the differential stage, instead, the above described effects concerning the bias current must be considered.

It is also worth citing, what was found in [1]. Here a macromodel design is shown, taking into consideration the coupling of the EMI to the cables and to the PCB. This study, in particular, permits evaluation of the EMI disturbances, the demodulated RF, LF, and offset, considering only the amplifier as a box, and the extraction of the values used in the model through measurement and testing. Differently than the previous studies, the latter has a more practical application, useful for different types of amplifiers and not requiring detailed knowledge of what is inside the circuits and their parasitics.

Finally, it should be noted that the susceptibility of opamps to EMI is not limited to the presence of RF at the input of the amplifier. Studies have been conducted to characterise the effects of disturbances coupled to the power supply pins and to the ground plane (see [22] and [2, 3]). Keeping in mind that the increasing integration trend makes it possible to have, inside the same chip, analog and digital circuits that can share the same ground.

2.3 Possible design solutions for the EMI problem

Starting from what seen in the previous section, the EMI effects play an important part in the design of a front-end circuit. Due to this fact, different operational amplifier designs have been proposed in the literature that show a reduced susceptibility to the RF but, at the same time, have some drawbacks. Common solutions include low pass filtering techniques ([3, 10]), symmetrical topologies ([25]), double differential stages, replica stages or modified replica stages ([9, 26]), source-buffered amplifiers, mixed type amplifiers like replica combined with source-buffered([6, 27]), complementary input stages aimed to cancel opposed nonlinear effects, chopped amplifiers ([5]), and modification of the standard Miller amplifier with source degeneration resistances.

In general, each solution shows positive and negative aspects. The EMI reduction obtained with filters, for example, gives good results attenuating both the common mode and the differential mode of RF but, in turn, makes a reduced phase margin that can cause problems with stability. Double stages are effective on the EMI problem but, again, the price to pay is the reduction of the bandwidth of the amplifier and also of the unitary gain frequency due to the increase of the parasitic capacitance involved. Source-buffered amplifiers show a DC offset for frequencies below a hundred MegaHertz but have good results on frequencies above 200MHz ([26]). Mixed solutions give, instead, good results for a wide set of frequency ranges.

All the proposed topologies are, therefore, a trade-off between complexity, desired offset, susceptibility, band, and stability. What is important to notice is that each different layout implies a modification at the integration level that must be done at the amplifier design stage, so, these are dedicated, non-standard solutions.

In the following, will be a description of some of the solutions available today that can be found in the literature.

2.3.1 Modified input differential pair and filtering

Modified input differential pair amplifiers have been presented in [10] where three different solutions are described. Although the techniques proposed reduce the EMI effects, at they simultaneously result in cause evoke a worsening of performance in terms of phase margin or common mode input range that, in some cases, undermines the effectiveness of the methods proposed.

An initial possible solution is obtained by modifying a standard differential pair by adding an RC filter at both inputs (see figure 2.6). The low pass filter, LPF, can

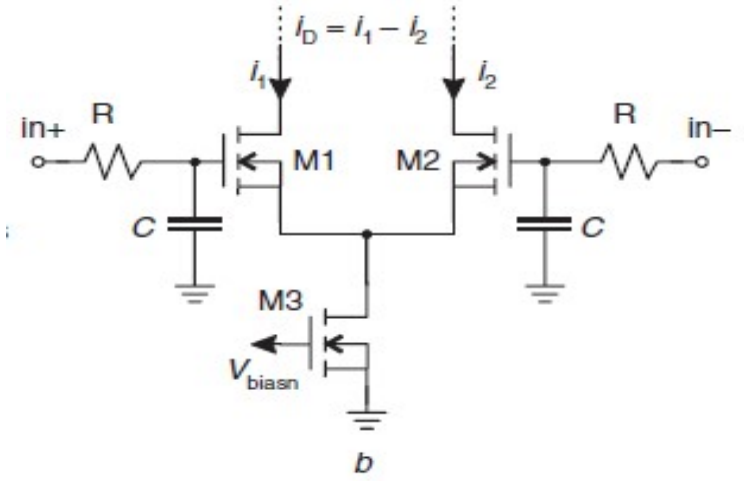


Figure 2.6: EMI resisting DP with RC filter taken from [10]

reduce the convoyed RF reaching the gate of the transistors, thus attenuating the RF common mode voltage V_{cm} and differential voltage V_d that generate the offset. This filtering effect is, however, limited to out-of-band RF disturbances since its cut frequency must allow the in-band signals to flow at the input. For this reason, the pole is placed close to the frequency limit of the amplifier (see [27]), however, it shows up in the differential gain, worsening it. In fact, the differential current and the differential amplification have expressions like those in equation 2.8 (with R the input resistor, C the overall capacitance and $A'_d(s)$ the gain of a standard non-modified amplifier, gm the transconductance of the pair) where the additional pole comes out.

$$\begin{aligned} I_d(s) &= \frac{gm}{1+sRC} V_d(s) \\ A_d &= \frac{A'_d(s)}{1+sRC} \end{aligned} \tag{2.8}$$

The pole, moreover, is responsible for a phase margin reduction, also due to the fact that the cut frequency is maintained close to the unity gain frequency. A trade-off between the two effects must be considered in this case, although (see [27]) good results are obtained in terms of induced offset reduction.

Altogether, the problems related to these type of topologies are of very poor phase margin, (PM), that makes their use almost impractical. An improvement of the PM, maintaining an equivalent frequency response with respect to a classical Miller topology, is the Miller replica amplifier, shown later in more detail.

A possible extended version of those just seen, that intrinsically implements a form of filtering effect, is shown in [5] where it compares the EMI susceptibility of a common continuous time amplifier and a chopped amplifier. This case highlights that although the chopped topology presents an offset both due to technological reasons (typical on standard amplifiers and partially eliminated by the chopping technique) and due to EMI-induced offset (determined by the mixing of EMI and the square wave produced by the switching activity, however, it is negligible with respect to the distortion produced by nonlinearities), it could still obtain a good filtering effect. The latter, indeed, arises from the presence of the modulator at the input and at the output that adds a resistance in series with the input (output) that, in combination with the input (output) parasitic capacitances, reduces the EMI through RC filtering and in turn the induced offset (see figure 2.7).

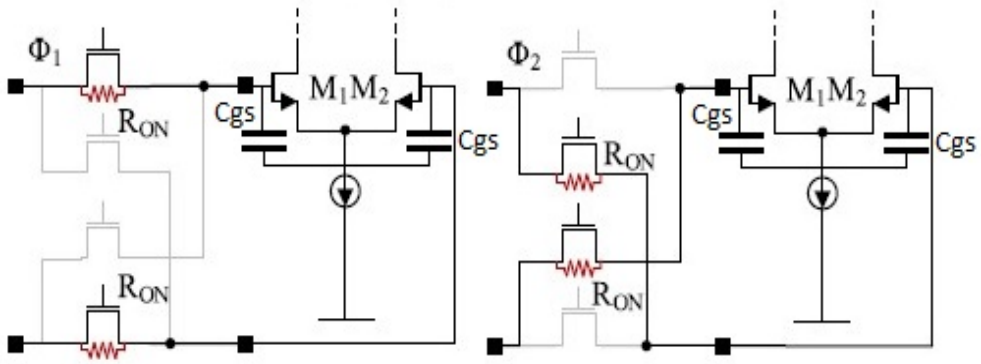


Figure 2.7: Filtering in a chopped amplifier from [5]

2.3.2 Cancellation topology

A double differential pair topology is illustrated in [25] (see figure 2.8) where two cross-connected differential pairs are present. In this case, the variation of the mean value of the differential current induced by EMI on the first pair is compensated by the second stage, due to the cross link. Indeed, the input connections are the same for the first and the second pair, but the load connections are crossed. The overall effect is a cancellation of the EMI-induced current, with little reduction of the performance in band (clearly, the dimensions of the pairs must not be exactly the same, otherwise, all the variations compensate each other).

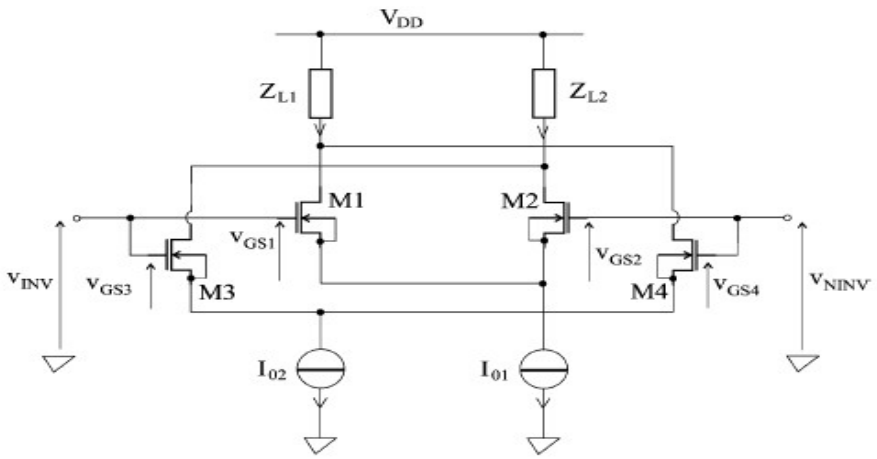


Figure 2.8: Crossed input stages taken from [25]

Another solution that aims for a cancellation effect, is further presented in [10]. The latter consists of the introduction of two complementary pairs (an nMOS pair and a pMOS pair with both the inputs connected together and whose output is the overall differential current sum of the two stages $I_d = I_{d,n} + I_{d,p}$) (see figure 2.9).

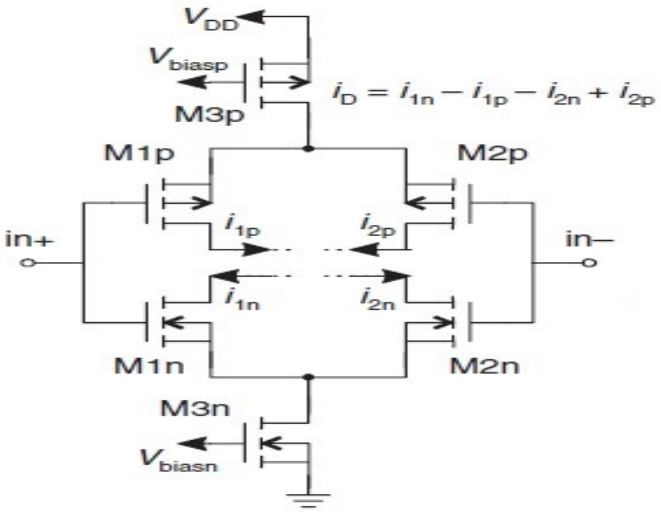


Figure 2.9: Emi complementary DP from [10]

In this configuration, in fact, the nonlinear effects of N and P MOS are compensated. A description of the mechanisms involved can, also, be found in [28]

and [12], where they highlight an opposite effect on the induced offset. In particular for an nMOS differential pair, a positive relation with EMI arises and, for a pMOS differential pair, a negative relation is obtained. The articles describe how the differential overall current variation $\Delta I_d = \Delta I_{d,n} + \Delta I_{d,p}$ still, as explained in the previous section,depends on from the V_{cm} but, here, the two components that add up have a phase difference of π that make the summation a difference whereby they compensate each other. Benefits of this configuration are that no additional poles are introduced and, so, the phase margin is preserved. On the other hand, the common mode input range, CMIR, is reduced by approximately $V_{gs} + V_{od}$ for the lower voltage and by the same quantity with respect Vdd for the higher voltage, since both the N and P stages must be maintained in saturation (input MOS and source tail MOS).

A more sophisticated design, that keeps the results seen in the latter topology, is further shown in [10]. Starting with a complementary design, this is modified biasing the inputs of the pMOS differential pair to an intermediate DC voltage (provided) through a resistor. The PMOS inputs are then connected to the nMOS input through a capacitor that decouples the input DC (so the R and C together show high pass filter behaviour). In this way, the circuit acts as the circuit described before, compensating the opposite EMI effects but, at the same time, it decouples the pMOS from the outside V_{cm} eliminating the upper limit of the CMIR that could almost reach VDD (see figure 2.10)

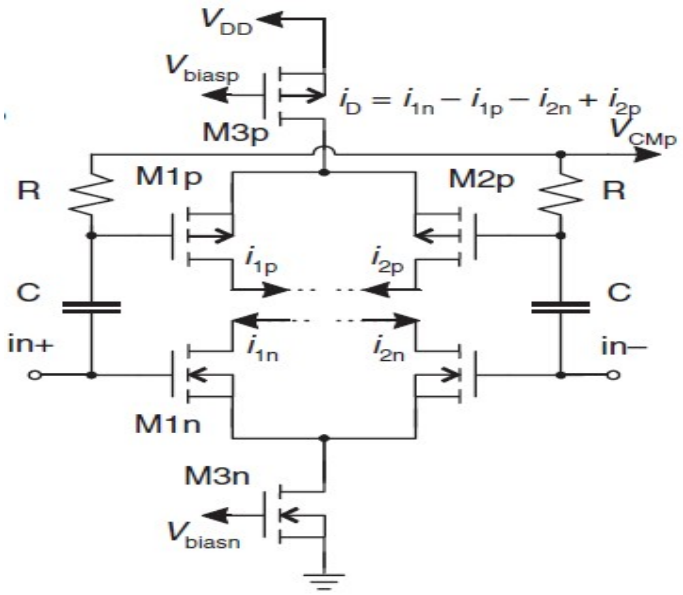


Figure 2.10: Complementary stage topology with increased CMIR taken from [10]

2.3.3 Advanced topologies

In the following, some of the most advanced topologies for the EMI effect reductions presented in the literature will be shown .

A replica Miller opamp is described in [9]. Starting with modifications of a classic Miller amplifier (see figure 2.11) they obtain a more robust reaction to EMI circuit that exhibits low susceptibility to RF coupled to the input pin in low, medium, and above the band frequencies. The proposed topology makes some

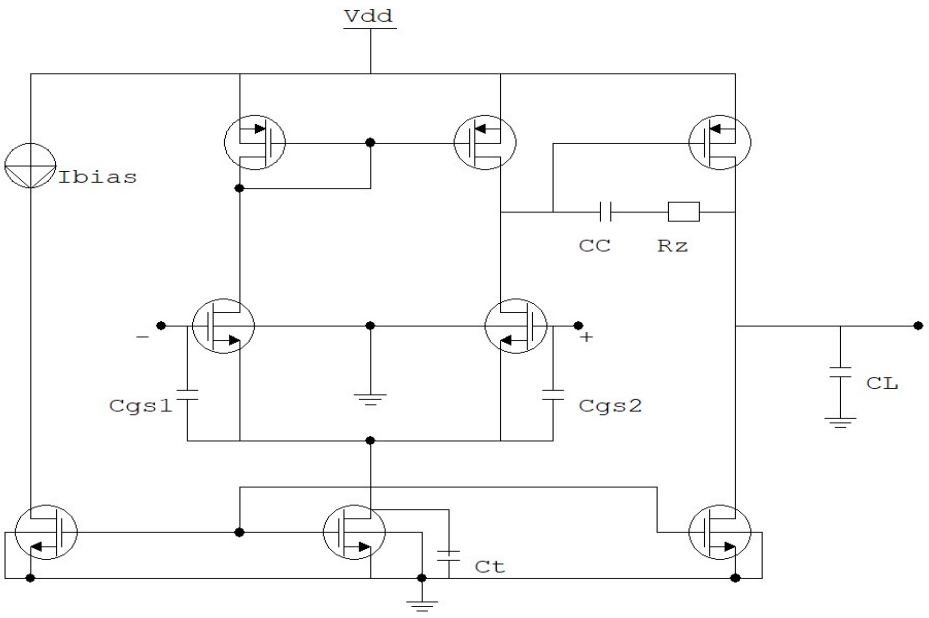


Figure 2.11: Classic Miller Amplifier

non-invasive modifications to a classical amplifier (in the article, they maintain the same dimensions as a classic opamp), keeping a large gain and a high unitary gain frequency (see figure 2.12) .

The method exploits the quality, in terms of EMI effect reduction, of an RC filtering solution but limiting its drawbacks. This improvement has been done through a replica of the differential pair. So, in this topology, it presents a first stage, the effective one, whose active load transistors are not directly driven by the same stage but, instead, are driven by another inner differential pair that has, at its input, an RC filter and whose current source is not the same as the first. In this way, this latter tail current is less EMI-sensitive (remembering that the primary effect of RF interference that causes the offset shift is due to the finite impedance of this transistor) and, in turn, this induces an overall reduction in sensitivity.

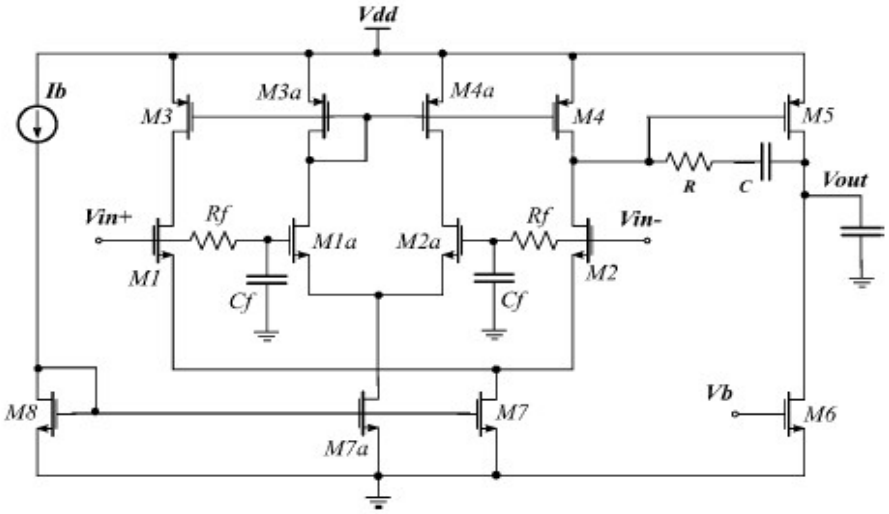


Figure 2.12: Replica amplifier proposed in [9]

Altogether, the amplifier has a comparable, to the RC model, reduction of the induced disturbance from EMI but, at the same time, still gives a good phase margin and a good frequency response similar to the classic Miller operational amplifier (in the article, it can be seen that in a frequency range from 10MHz to 1GHz the offset obtained is less than 10mV).

A possible modification of the topology just presented that introduces a source degeneration is proposed in [26]. Here, the introduction of the source resistances at the outer pair reduces the output offset voltage through a linearization of the input transconductance gm ($G_{m,eq} = \frac{g_m}{1+g_m R_s}$ that for $g_m R_s >> 1$ drop to $G_{m,eq} \approx \frac{1}{R_s}$) with good results in comparison with a classic Miller amplifier.

In [6], a modified replica Miller amplifier with source-buffered technique is shown. In this case, like in a replica Miller opamp, an inner and an outer differential pair are present (see figure 2.13). The inner one drives the outer one through the upper pMOS current mirror transistor. The inner one can, consequently, reduce the common mode input disturbances and make an uncoupling of the input CM to the output, because the current of the top mirror is driven by the inner one that is filtered because it exploits the replica Miller opamp effect. At the same time, it increases the EMI robustness with a source-buffered technique, obtaining good results for the immunity from a few MegaHertz to GigaHertz.

The article presents this mixed topology with the aim of overcoming the simple replica problems (high capacitance at the inner pair at the drains of M3a and M4a).

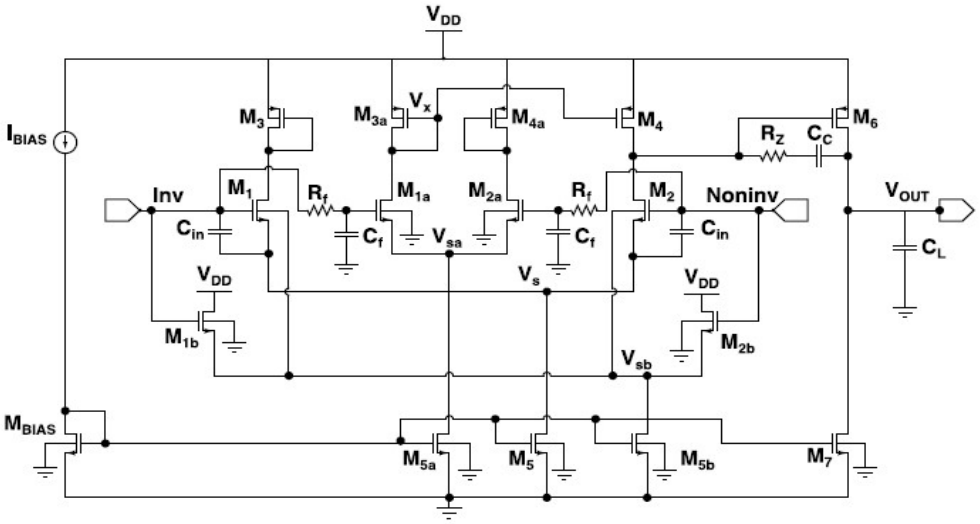


Figure 2.13: Modified replica source-buffered taken from [6]

In particular, the circuit is modified by adding a source-buffered stage and making a diode connection of the active load. The latter, indeed, reduces the node V_x capacitance (figure 2.13). The buffering technique consists, instead, in biasing the body of the N couple of the first differential pair by a second differential pair that keeps the average drain current almost constant through a body effect.

A recent study, following the articles seen above ([6, 26]), is presented in [27]. Here, after a summary of the effects of different configurations, it introduces a new topology that melds together the modified Miller replica source-buffered with the Rs degeneration. All the simulations and tests are done in a voltage follower configuration, remembering that the maximum of the generated output is obtained when the RF at the input has, both, $v_{cm,RF}$ and $v_{d,RF}$ not equal to zero and with the same phase. The results obtained in terms of gain and phase margin are comparable to the classic Miller opamp. The configuration, instead, drastically improves the performances in terms of offset. From MHz to GHz, the offset is kept two orders of magnitude below the classic Miller amplifier (from a maximum of 216mV to 5mV for the new topology). On the other hand, a reduction of the unitary gain frequency is present (from 50MHz to 15MHz [27]).

Mainly, an improvement of the susceptibility is obtained at the price of requiring a complex and non-standard layout that could, in practice, take time to adjust each single transistor well.

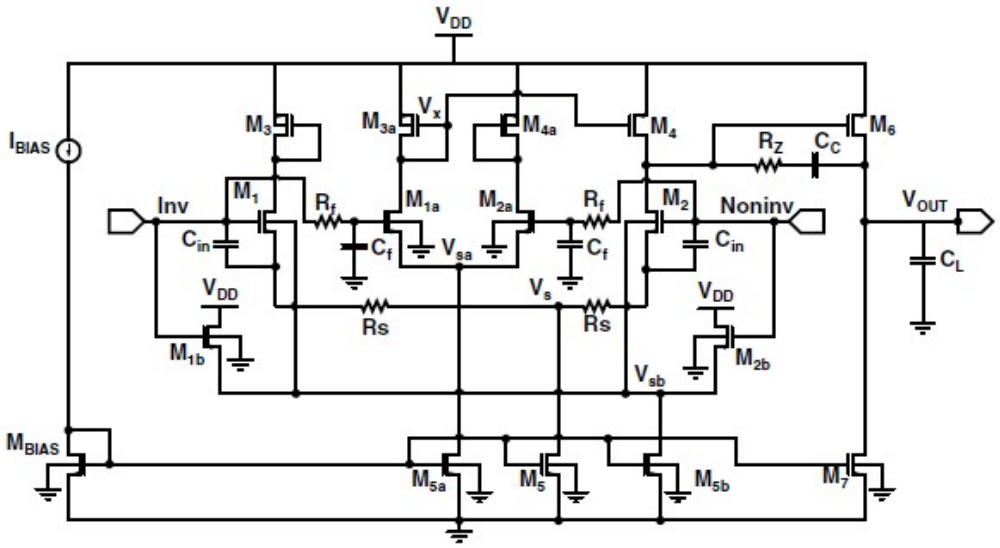


Figure 2.14: Source-buffered replica with degeneration proposed in [27]

2.3.4 Complementary topologies

Another important topology, the basis for the development of the technique explained in this thesis, can be found in [12] and [28], where a similar behaviour is exploited by taking advantage of two complementary differential pairs.

The basic idea behind these studies is to compensate for the distortions by employing two input stages, one that shows a positive RF-induced offset (an nMOS differential pair) and one with a negative-induced offset (a pMOS differential pair) (see figure 2.15). The output of that combined stage is the differential current, the sum of the differential current of the nMOS pair and the pMOS pair ($I_D = i_{d,n} + i_{d,p}$).

Due to the fact that the two types of circuits show opposite responses, the overall effect is a distortion compensation. The expression of the current, indeed, depends on the differential RF voltage and the common mode RF voltage and on two terms whose phases are, for topology reasons, shifted by π (from [12]: $\Delta I_D = \frac{V_{d,pk}V_{cm,pk}}{2}[g_{p,n}|Y_n(j\omega)| * cos(\phi_{cm,n} + \angle Y_n(j\omega)) + g_{p,p}|Y_p(j\omega)| * cos(\phi_{cm,p} + \angle Y_p(j\omega))]$, with $\phi_{cm,p} = \phi_{cm,n} + \pi$, remembering that the $cos(x+\pi) = -cos(x)$ the subtraction effect is demonstrated). The small signal equivalent shows, indeed, that an increase of v_{cm} determines an increase of the small signal tail current on the N pair and a reduction on the P pair.

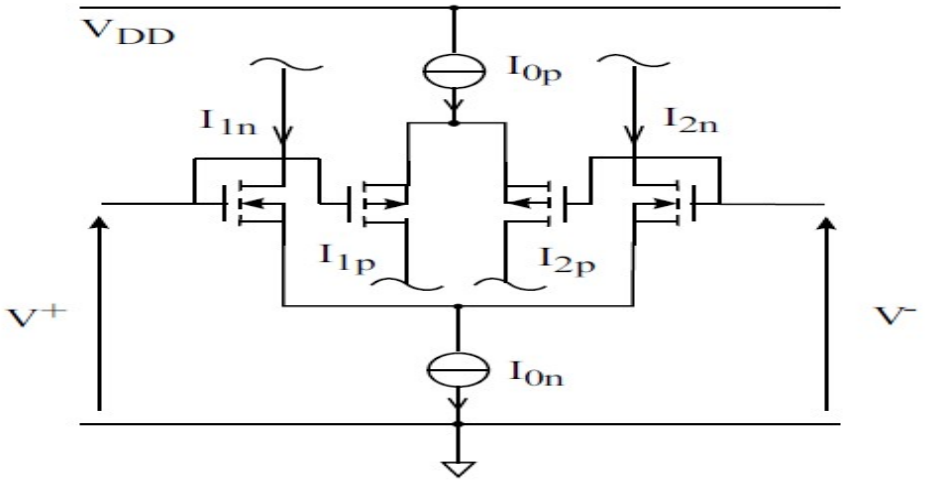


Figure 2.15: Complementary input stage from [12]

2.3.5 Comparisons

A good analysis, also in terms of slew rate, between different configurations is done in [3], where they compare some of the configurations just seen with a CMOS Miller Amplifier (input differential stage followed by a common source stage, figure 2.11). The study concludes that a classic Miller amplifier is better in terms of gain (being a two stage circuit) and in terms of voltage swing but, at the same time, can have large asymmetry on the positive and negative slew rates. In fact, it shows a relevant positive slew rate and lower negative slew rate. This is due to the nMOS output transistor that acts as a current source and also due to the channel length modulation on the tail current transistor. A positive step signal at the input, in fact, causes an increase in the bias current and, in turn, a higher SR, and a negative input step has the opposite effect ([21, 20]). Due to these critical issues, the susceptibility to RF is evidently present also at medium frequencies, due to the asymmetries (slew rate).

Another comparison is done with the source-buffered (remembering that the differences with respect to the Miller are the presence of a bulk bias network and some capacitances added to the input between the gates and the sources of the input transistor). In this case the output offset is considerably lower since source-buffered reduces the tail capacitance (the CT capacitance, see 2.11) which shorts the current at high frequency, in addition to the input capacitors that, at the same, filter the EMI at high frequency.

The comparison to the folded cascode with replica (a version with the presence of an input replica with RC, like in the Miller replica, but in the form of folded

cascode, see figure 8 in [3]) shows that because this topology is a single stage configuration, the bandwidth is better than a two-stage (where compensation is required) but, at the same time, there are limits on the output swing due to the increased number of transistors stacked on top of each other. Furthermore, showing better symmetry, the folded version is less sensitive to EMI at lower frequencies (similar SR) but exhibits the problem nearing high frequencies.

An enhanced version where the addition to the previous model of a source-buffered bulk bias is discussed, showing how this configuration further increases the robustness to EMI.The final solution presented in the article is a differential pair with the addition of a common mode cancellation circuit.

A fundamental problem can be highlighted from this comparison of different EMI immune topologies. All the configurations shown can considerably reduce the susceptibility but at the cost of an increase in complexity. This keeps the improvements finite to an integration level.

As discussed here, various topologies have been proposed to overcome the induced effects of the EMI at input. As observed and summarised in [29] a categorization of the current possible solutions can be done, valid for the out-of-band distortions. All of these can be divided in two main subcategories: distortion mitigation techniques and distortion cancellation techniques. The first of the two is considered less immune with respect to the other since it consists of adopting solutions that block the flow of the RF disturbances before they arrive at the nonlinear device and, in doing so, avoid the effects of the nonlinearities (that are always present in some form). This type of blocking technique, can, in any case, be susceptible at the out-of-band EMI because RF can reach the interested device through other paths that are not considered.

The second type of solution, instead, have a more robust immunity since they aim to cancel the effects of distortions with the use of nonlinear devices that show complementary behaviours to the EMI or with the use of the same devices subtracting the induced nonlinearity or even creating proper predistortions of the input signal before it arrives at the interested device.

A primary mitigation technique is the feedback, where the overall gain depends on a linear, passive, back-reactive network only if the forward open loop gain is high at the interested frequency. A second solution is filtering with linear devices, passive devices, at input or in the feedback loop, taking into account the fact that the entire frequency response must be unaltered i.e., the additional pole introduced must be above the dominant one and, at the same time, must give sufficient attenuation at the desired frequency (in this category there are the filter techniques previously shown).

Among the cancellation techniques there is, in primis, the differential stage,

with all the limitations already shown, considering the difficulty to obtain an ideal differential signal with no common mode. Other general solutions adopted in this category, are those already seen.

The last of the cancellation techniques that can be adopted and that has not yet been seen is the feed-forward configuration. The latter subtracts from the amplified signals with the distortion added, the only distortion.

In this way, a summary of the possible solutions has been presented. In the next section, an insight into a non-integrated solution is shown ([30, 13]). This lays the groundwork for that which follow in this thesis.

2.4 Development of complementary based input solutions in FE

A starting point for all the following work can be seen in the [30] article where a system level method of EMI effect reduction is presented, based on complementary input stages. Here, a susceptibility reduction in front-end systems is obtained through a dedicated, separated preconditioning of the wanted signal performed with standard devices, a subsequent digital acquisition, and a post processing of the data.

An acquisition system, in general, is composed of a series of stages, starting from the anti-aliasing filter, followed by the conditioning amplifier, CA, a sample and hold, S/H, and the analog to digital converter, ADC, whose output can be processed by a digital controller. The anti-aliasing filter whose aim is to keep only the baseband signal and cut the frequencies above the desired band can do little with respect to the high frequency unwanted signal present at its input. Indeed, through parasitic capacitances the EMI can easily reach the following amplifier and produce the effects seen in the previous section i.e., demodulating the superimposed RF signal. The drawback of this propagation is that the acquired data can be corrupted or even erroneous.

A proposed solution to this problem, based on standard components, is exploited by doubling the signal path and adopting two different conditioning amplifiers that show a complementary response to the EMI at their input. Two opamps with complementary differential input pairs, an nMOS input (NPN BJTs) stage amplifier and a pMOS (PNP BJTs) input stage amplifier (see figure 2.16) could be employed, as seen in the previous section. Indeed, an nMOS has been proven to have a positive response to the EMI i.e., an increase of the DC shift with a CW applied RF, instead, pMOS has a negative response to the same input (demodulation occurs if the EMI signal varies its amplitude and does not remain a continuous wave). The distorted output signals, y_1 and y_2, both corrupted by opposite sign

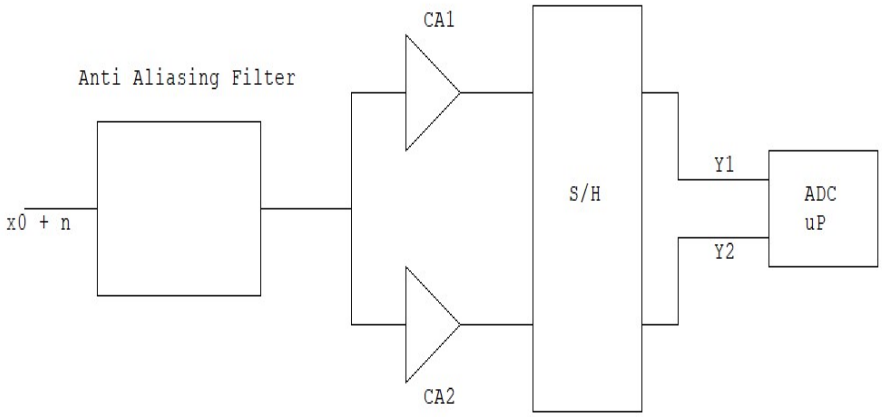


Figure 2.16: EMI immune front end proposed in [30]

EMI demodulation, are then sampled and converted. A dedicated post processing technique can, if well-tuned, extract the error from the two acquired set of data, and purge (almost, in theory) the acquired signal from it.

The proposed algorithm in [30] consists in sensing if EMI is present in the acquired data. If the signal is not corrupted, the output $y_1 \approx y_2$. In this case the y can be taken as an estimate of the two acquired values. On the other hand, in case of $y_1 \neq y_2$ the error is present and affects the y_x values in a different way but that depends on the input RF amplitude. So, the acquired samples can be expressed as 2.9:

$$\begin{aligned} y_1 &= x + n_{1,EMI} \\ y_2 &= x + n_{2,EMI} \end{aligned} \tag{2.9}$$

The original signal, not affected by EMI, can be evaluated as the weighted sum of the two acquired signals, eq. 2.10, where for a particular value of α, the EMI effect can be cancelled obtaining $y = x$ (x the original signal sample).

$$y = \alpha y_1 + (1 - \alpha) y_2 = x + \alpha n_{1,EMI} + (1 - \alpha) n_{2,EMI} \tag{2.10}$$

Because the value of the distortions, n_x, are by construction different, the value of α that cancels the error is obtained as 2.11

$$\alpha^* = \frac{n_{2,EMI}}{n_{2,EMI} - n_{1,EMI}} \tag{2.11}$$

The n_x values can be estimated in the moment when the acquired input starts to change due to the disturbances. In this way, having determined that the $y_1[n]$ and $y_2[n]$ (with n the number of the sample) are not similar, they are compared with the previous acquired values $y_1[n-1]$ and $y_2[n-1]$. If the latter were equal, the EMI would just arise, between the n-1 and the n sample. The value of the error added can be evaluated by subtracting both the previous and the following y_1 and y_2 samples (see eq.2.12), supposing that between the n-1 sample and the n sample the real value of the input x is unchanged (due to the high sampling frequency).

$$n_{i,EMI} = y_i[n] - x[n] \approx y_i[n] - x[n-1] = y_i[n] - y_i[n-1] \quad \text{with } i = 1, 2 \tag{2.12}$$

As described furthermore in [30], the same value of $n_{x,EMI}$ and so of α^* can also be used if the voltage of the EMI changes. In fact, the $n_{x,EMI}$ can be expressed as $n_{i,EMI} = n_{0i} V_{EMI}^2$ where the n_{0i} are, with good approximation, independent of frequency constants. In this way, substituting these expressions in 2.11 a value of α^* can be obtained where the v_{EMI}^2 cancels out and leaving only (2.13):

$$\alpha^* = \frac{n_{02}}{n_{02} - n_{01}} \tag{2.13}$$

From the article, it can be concluded that this value of α^* can also be employed in case of RF voltage variations, frequency variations and other general cases of modulated EMI amplitude.

CHAPTER 3

The presented solution

From the previous chapter's is studies and, in particular, starting from [30] it is developed the following design aimed to evaluate the effectiveness of the theory applied to a front-end acquisition system based on standard off-the-shelf amplifiers.

3.1 Theoretical base

In [13], the article is continuing what was introduced with [30], and that is the starting point of all the work for this thesis. Here, a similar front-end system, where the conditioning amplifiers are configured as voltage followers, is tested. The complementary EMI paths are obtained with a dedicated board where two amplifiers, with opposing EMI susceptibilities, are mounted (a BJTs input stage and pMOS input stage that show opposite behaviours to RF). At the output of both opamps, one can find the desired input signal v_x added to distortions generated by the EMI, $\Delta v_{emi,i}$ (see figure 3.1). For the two outputs, it can be obtained by: (eq. 3.1)

$$\begin{aligned} v_1(t) &= v_x(t) + \Delta v_{EMI,1}(t) \\ v_2(t) &= v_x(t) + \Delta v_{EMI,2}(t) \end{aligned} \tag{3.1}$$

In case of a narrow-band RF signal with frequency above the band of the amplifier (see the previous chapters), the effects at the output can be expressed in terms of a complex envelope of the EMI (Hilbert Transform [31]). From the above studies, (see [1]), the distortion produced can be expressed as (eq. 3.2):

$$\begin{aligned} \Delta v_{EMI,1}(t) &= K_1 |\widetilde{v}_{EMI}(t)|^2 \\ \Delta v_{EMI,2}(t) &= K_2 |\widetilde{v}_{EMI}(t)|^2 \end{aligned} \tag{3.2}$$

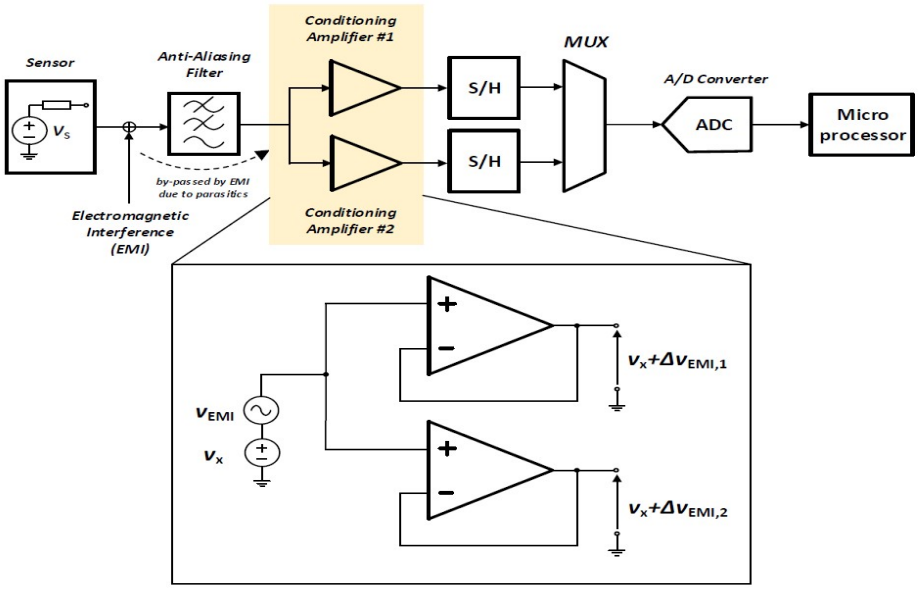


Figure 3.1: Configuration proposed in [13]

where the constants K_1 and K_2 are dependent on the circuit configurations and the devices employed, while the EMI voltage is the same in both cases. Here, amplifiers with different input stages are adopted, and this makes the constants have opposing signs, a condition required, [30], to obtain a disturbance compensation.

The following analog to digital conversion stage produces, therefore, samples ($y_{i,k}$ with i=1,2 depending on the amplifier considered and with k the acquisition number) that are the sum of a wanted value (x_k) to which it is added an EMI-dependent term ($n_{EMI,i,k}$). Following [13] and with the same approach as [30], one can obtain a weighted sum of the digitized signal plus an error. This sum, for a particular value (to be computed) of the parameter α, can achieve an error cancellation.

The obtained expression, which is the same used in the post-acquisition algorithm in the next of the chapter, is exactly (eq.3.3):

$$\begin{cases} y_k(\alpha) & = \alpha y_{1,k} + (1-\alpha) y_{2,k} = x_k + n_{\epsilon,k} \\ n_{\epsilon,k} & = \alpha n_{EMI,1,k} + (1-\alpha) n_{EMI,2,k} \end{cases} \tag{3.3}$$

The latter, as highlighted in [13] and [30], can, for an appropriate $\alpha = \alpha^*$, exhibit a cancellation of the EMI noise $n_{\epsilon,k}$. In particular, for $n_{EMI,1,k} \neq n_{EMI,2,k}$ (to avoid zero division), it is possible to compute that value in this way (eq.3.4):

$$\alpha^* = -\frac{n_{EMI,2,k}}{n_{EMI,1,k} - n_{EMI,2,k}} \tag{3.4}$$

Because $n_{EMI,i,k}$ are unknowns, α^* can not be directly computed. However, exploiting expression 3.2 and remembering that a sample takes the form of $y_{i,k} = \frac{v_i(kT_{samp})}{V_{REF}/2^N}$, the value of the disturbances can be expressed as (eq. 3.5):

$$n_{EMI,i,k} = K_i \frac{|\widetilde{v}_{EMI}(kT_s)|^2}{V_{REF}/2^N} \quad \text{with } i = 1, 2 \tag{3.5}$$

Substituting this expression that, for both i, depends on the same $\widetilde{v}_{EMI}(kT_s)$ value, one can obtain a dependence only on the K_i values (i.e., $\alpha^* = \frac{K_2}{K_1 - K_2}$), not from the specific sample k, and on the specific EMI at the input. These K_i values are only related to the specific configuration adopted.

The link of the α^* to the errors obtained can, thus, permit the simple writing of a digital algorithm to be executed by a micro-controller (like ADuC7061) or other CPUs, aimed to recover the original base-band signal purged by the errors introduced by the EMI on the operational amplifiers.

As shown in [13], the K_i values can be computed in an analytical form or by direct measurement in the presence of a CW EMI. This last method permits the direct acquisition of $n_{EMI,i,k}$, and it is straightforward. It consists of measuring with and without RF, and subtracting the sample of the original signal (known in this case) from the EMI corrupted sample for the same time instant. The method must be applied for both the channels obtaining the $n_{EMI,i,k}$ values that through eq. 3.4 gives the desired α^*. As discussed above, this parameter should be, at least in an initial approximation, constant with respect to the EMI amplitude and frequency. The value obtained and used in the following, and related to the the double amplifier board employed (fig. 3.2) is $\alpha^* = 0.3629318$.

To better estimate α^*, in different EMI conditions (amplitude and frequency), a weighted sum of different measured $n_{EMI,i,k}$ can be computed (see the measurement chapter), as explained in [13]. This technique could be exploited in an automated way like an initializing routine for the system. Hypothetically, the adoption of some sort of automatic injection in addition with a constant DC value input could be used.

3.2 The double amplifier board and the filter

This section describes the board that hosts the two operational amplifiers.

As discussed in [13], the chosen amplifiers are two complementary susceptible opamps: a laser-trimmed BJTs opamp, remembering that for bipolar transistors one can obtain similar expressions for the offset with respect to the CMOS [20, 15,

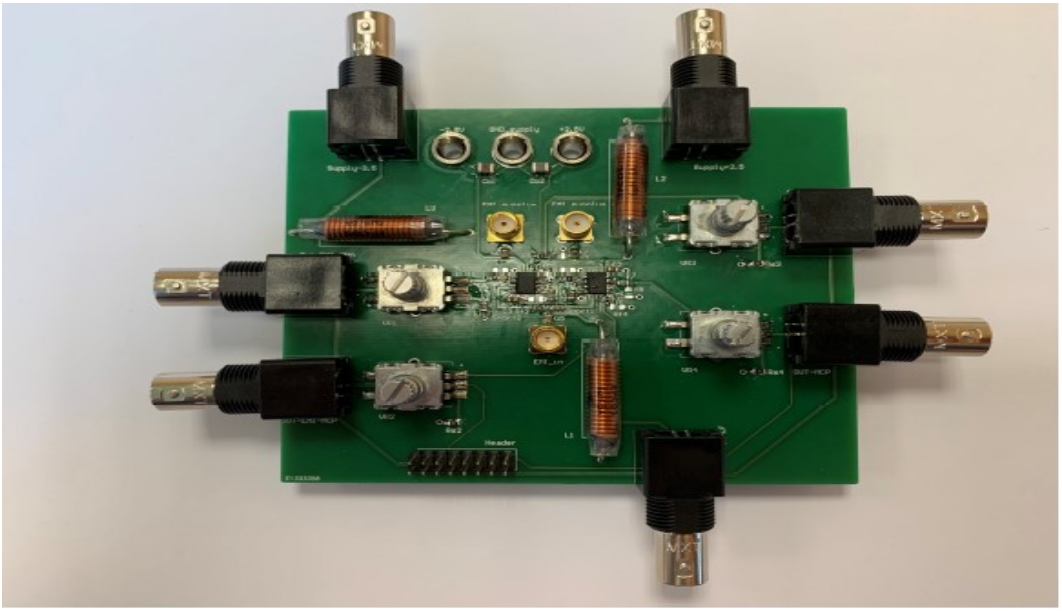


Figure 3.2: The double amplifier board [13]

19], and a CMOS. In particular, the ones installed are the OPA2277 ([32]) and the TLC272 ([33]) both from from Texas Instruments (see [20] for a characterization to EMI of the OPA2277). The supply voltage chosen is $\pm 2.5V$, still within the applicable voltage for both amplifiers. They exhibit these characteristics: an open loop gain of 134dB for the OPA and 92dB for the TLC, a GBW of 1MHz and 1.7MHz, a SR of $0.8V/\mu s$ and $3.6V/\mu s$, a common mode rejection ratio of 140dB and 85dB, and a power source rejection ratio of 130dB for the OPA and 85dB for the TLC. They have the same order of input offset, $10\mu V$ and $1.1\mu V$ and an input noise voltage at 1kHz of $8nV/\sqrt{Hz}$ and $25nV/\sqrt{Hz}$ (due to the their different technology the input bias and offset currents are of different orders of magnitude). To be noted and later seen under filter description, the OPA2277 has a limited output swing with respect to compared to the TLC272 (it clearly shown with the application of a unipolar supply). The overall configuration of the boards can be seen in figure 3.3. A voltage follower configuration is adopted since it can be considered for the worst cases (see chapter 2) in terms of EMI susceptibility.

Differently from [13] (where only a low frequency, LF, signal is injected) the inputs of the amplifiers are fed by an LF signal biased with a 600mV constant voltage and an RF generator. Both inputs are summed by a bias tee (see [34]). The input bias voltage has been required to respect the ADuC7061 input range of the Sigma-Delta converter, see next.

If required, a precise characterization of the output disturbances with respect

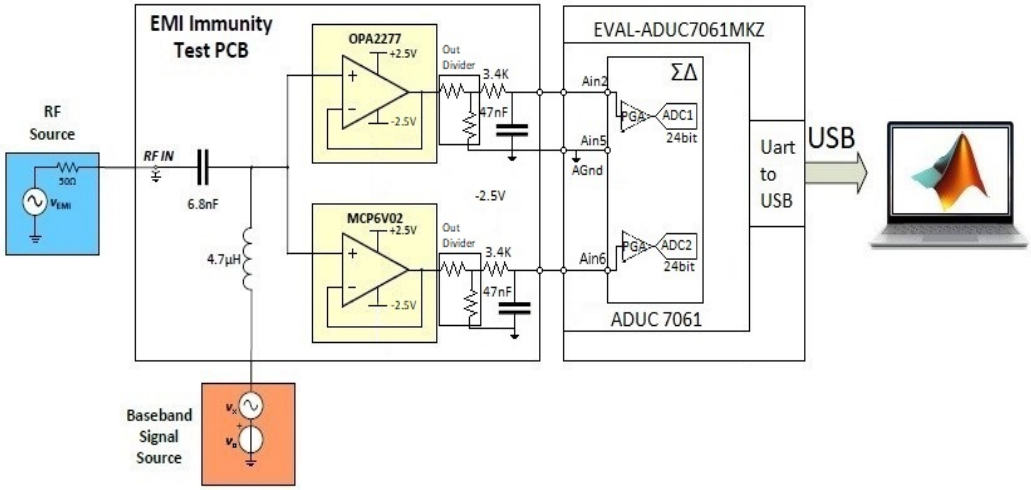


Figure 3.3: Acquisition front-end configuration

to EMI input voltages must be taken into account the reflection coefficient ($|S_{1,1}|$) present at the input of the amplifiers, due to the high impedance of the inputs with respect to the RF line, (follow [20] and [35] for details on how to proceed; to be noted that in case of S_{11} near to 0dB, high reflection coefficient almost at 1, the incident voltage seen at the opamp input is nearly doubled with respect to the voltage applied that there is in case of matched generator-load impedance, $V_{inc} = (1 + |S11|)V_{app} = (1 + 10^{\frac{|S11dB|}{20}})V_{app}$). In our case, this evaluation was not done because of our interest in the overall cancellation effect. Here is considered, so, the injected power coming out from the generator and the related open circuit voltage (equivalent to high reflection coefficient), since what is being proven is the efficiency of the method and not an exact link of the input power to the output values.

The opamps are connected to a $\pm 2.5V$ power supply. Because the RF source and the LF sources are GND referenced, an offset has been applied to the LF input. This configuration is required because the following ADuC7061 is designed to be supplied with a maximum of 2.5V, and its power voltage is fed by a USB connector that provides 5V, ground referenced. The latter is obtained by a regulator, an ADP3333ARM-2.5Z low dropout, that gives an output of 2.5V. The input of the analog to digital converter, ADC, Sigma-Delta of the ADuC, instead, only accepts a range from 0V to 1.2V and, so, the input and the output of the amplifiers must respect this range. For this reason, a DC offset is applied to the LF signal (600mV), such that the signal of interest is within the input dynamic (see measurement chapter for details). At the output of the amplifiers, to respect the input voltages,

a voltage divider is applied at the same time, (see figure 3.3 and 3.4), so that the desired input dynamic of the ADC is respected.

Of note, more will be discussed later, the ADuC ΣΔ inputs have been arranged to operate in a differential way. So, all the inputs are followed by a programmable gain amplifier, PGA, for the primary ADC, and, by a buffer, for the secondary ADC, (this is valid for both the positive and negative inputs, using the latter as a reference here). This configuration brings with it the problem of requiring a minimum voltage (not zero, from datasheet 100mV [36]) at the reference pin (Ain5 figure 3.3). This issue, not clearly highlighted in the datasheet, has been overcome, with further research, by disabling the reference input PGA (Buffer) bypassing it (see configuration section for details). This has been possible through the configuration of the micro-controller registers. After a measurement test the solution has been confirmed as valid.

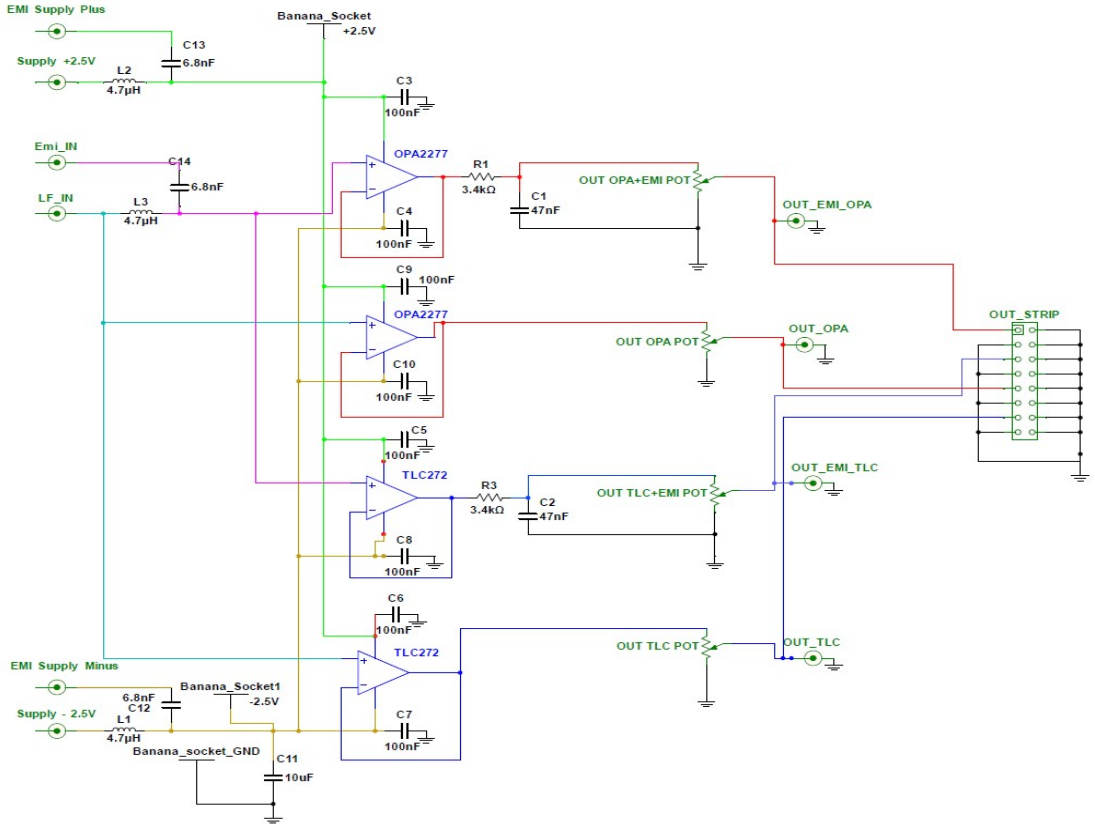


Figure 3.4: Acquisition front-end board schematics

Between the output of both the amplifiers and the input of the ADCs an additional RC filter has been placed with the aim of reducing the possible aliasing introduced by the out-of-band signals than can be present at the output of the

amplifiers and keeping in band only the harmonics desired to evaluate the proposed technique. This can avoid the the residual RF, coming from the amplifier or the feedback network, at opamp output that can reach the sigma-delta converter. As will be discussed in the measurement section, this filter is not necessary at all since the digital converter highly eliminates internally, with the digital filter, all the frequencies in the upper half of the sampling frequency (keeping in mind that the real sampling frequency is 512kHz, the Sigma-Delta, indeed, oversamples the input and what can contribute to aliasing is found above 256kHz).

The output sampling frequency of the ADuC7061 for both the inputs is 4000Hz (the band replica is centered here, see figure 3.8), this value is used to determine a useful band band up to a maximum of2kHz, from the sampling theory.

From previous studies, see the preceding chapters and also [1], there are, at least, three output components caused by the EMI disturances are present (see figure 3.5):

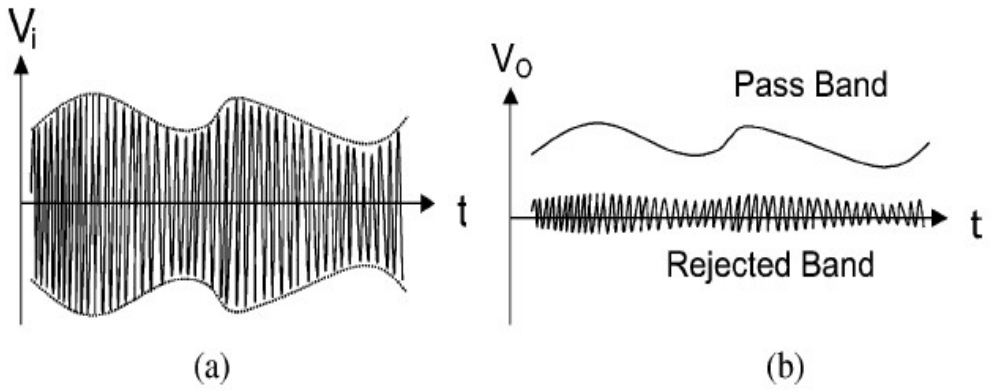


Figure 3.5: Opamp response to EMI from [1]

a DC component, in the form of an offset shift, related to the average of the signal after the non-linear distortion, an RF component due to the original signal that is highly attenuated due to the amplifier band limit band, and a low frequency signal that depends on the envelope of the radio frequency signal at input that is AM modulated are present. In this last case, when the modulating signal is not sinusoidal, other harmonics are present and are demodulated in band (see, for example, a square wave). The latter appear as distortion peaks in the spectrum of the output (see the square wave RF modulation in [13] and figures 3.6 for the intrinsic distortion caused by the amplifier with only an LF applied as the input signal and, see figure 3.7, for the output corrupted by the demodulated EMI square wave harmonics).

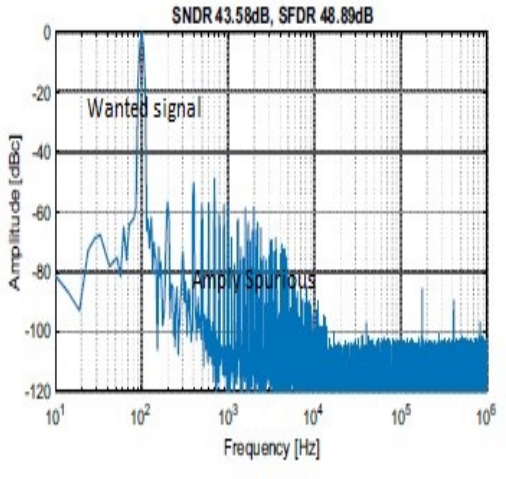


Figure 3.6: Amplier spectrum w/o EMI from [13]

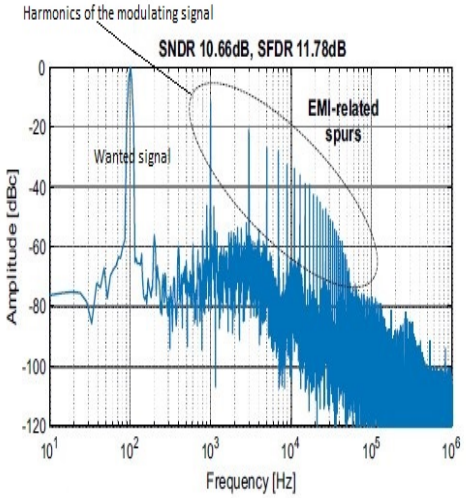


Figure 3.7: Amplier spectrum with EMI from [13]

For this reasons and with the aim of keeping, as much as possible, the EMI-demodulated, induced harmonics within the band of the filter and attenuate everything at an higher frequency (i.e., the residual RF signal and spurious), a first order RC filter with a pole at almost 1kHz has been designed. To avoid an impact on the desired output harmonics, the cutoff frequency has been maintained at least a decade above the first relevant component of the injected EMI (one of the injected RF signals is modulated with a 100Hz square wave whose spectral components are present on the opamps output spectrum). Because it is of interest to show that at the output the original input signal is present (in our case a low frequency sine wave at 10Hz) and some other added frequencies that the technique must reduce and that depend on the demodulation of the RF superimposed at the input, the band of the filter can theoretically be reduced. The reduction, however, must at least be such that it includes the desired signal and the RF demodulated one. To be remembered, a pole starts its effect one decade before and ends one decade after the nominal pole frequency value. This choice, therefore, must keep this situation into account to avoid distortion of the results introduced by a pole frequency of the filters which is too low.

Moreover, as will be shown, the sampling frequency used here is exactly 4kHz, and it cannot be increased thing that would permit to increase the band of the filter. Although the micro-controller Sigma-Delta can reach a maximum sampling frequency of 8KHz, which, in this case, has limited the choice, was available on the

chip UART controller used to exchange data with the PC. This, indeed, cannot transmit as much of the sampled data in the desired time if the ADC works at 8Khz. The ADuC7061, in fact, communicates directly with a FTDI FT232QN integrated circuit, taking advantage of the UART already present on the microcontroller. The FT232QN, then, implements the USB protocol to the personal computer. Increasing the speed of the transmitted data would have required not using the integrated UART (too slow) and, instead, would have required the use of the SPI bus. This, in turn, would have required changing the UART to USB chip with a SPI to USB integrated circuit. The drawback of this solution would have been the inability to use the demo board already available and the need to write a dedicated driver (not compatible with the time and the project target).

The above described filter has been installed on the double amplifier board that has been prepared in advance to host two SMD 0805 components. For this reason, the filter chosen is a first order RC filter made with a resistor and a ceramic capacitor. Because both the amplifiers have a maximum output current of almost 30mA (see [32, 33]) and to avoid working with the current limitations, especially regarding step response, the resistor employed is a $3.4K\Omega$ 1/8W, and the capacitor is a ceramic version of $47nF$ that gives a pole at a frequency of about 996Hz. The value has been chosen to respect the E96 resistor sequence. With the above components, the maximum output currents are some orders below the limit that can be delivered by the opamp, so no issues with current are present. The attenuation obtained, (see figure 3.8), is about 80dB at 10MHz and about 94dB at 50MHz (the lower of the RF signals applied to the input), frequency this that for replica drops in band at 0.3Hz and 0.075Hz. The frequency replicas that are present at 1kHz are, instead, attenuated only by 9.5dB and can create an aliasing effect. However, the cutoff frequency choice is a trade-off between maintaining an unaltered 10Hz base band signal and leaving, at least, the first fundamental harmonics of the modulating RF signal at input intact (one of the test measurements consists of injecting a 10Hz sine-wave and an RF modulated with a 100KHz square wave, see measurement chapter for details). To relax the analysis with respect to the previous amplifiers tests (see [13]), the value of the base-band signal injected at the input has been reduced from 100Hz, adopted in the cited study, to 10Hz and, in turn, has reduced the applied AM modulating signal frequencies. As highlighted by measurement, the need for the filter in this application has little effect on the sample spectrum obtained, also due to the digital filtering effect performed internally by the ADC decimator filter. On the other hand, it makes a relevant difference for the ADuC7061. Indeed, without the filter, during changes in the power or frequency of the injected signal, the microcontroller has shown erratic behaviour or has even stopped its operation, requiring a manual reset. The presence of the filter has almost totally eliminated this effect.

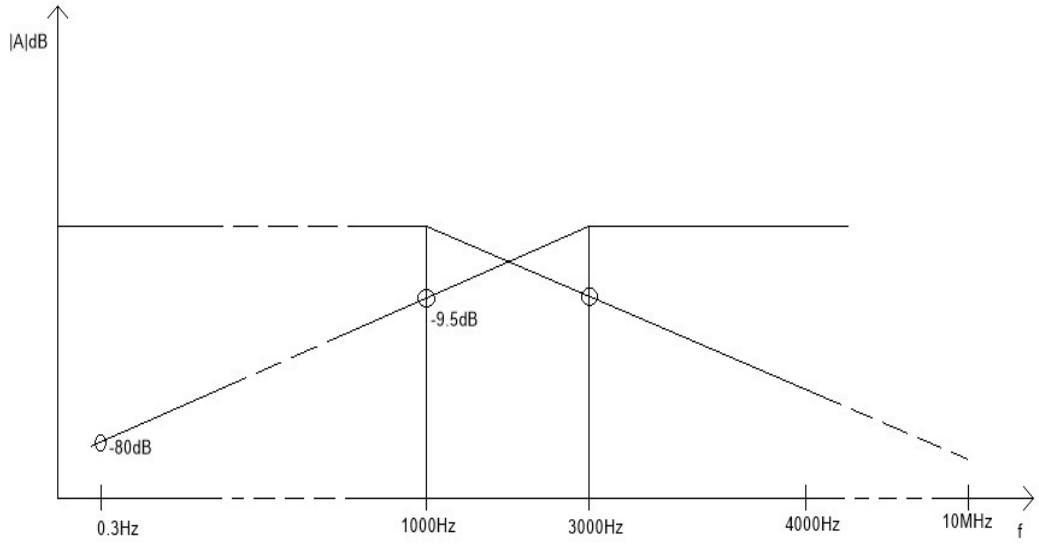


Figure 3.8: Filter attenuation tradeoff

Looking at the input of Sigma-Delta, its impedance is quite high (but not declared in the technical report) because a programmable gain amplifier or a buffer, for the secondary ADC, is placed between the input and the converter. Problems related to the presence of switched capacitors that operate at a high frequency can be neglected, considering the high band of the amplifiers used with respect to the sample frequency (see [37], where they highlight the problem of the output impedance of the opamps seen by the ADC that must be considered at the switching frequency and not at DC). So, thanks to this configuration, no particular issues must be taken into consideration.

The technical report does not provide the input capacitance of the ADC stage. So, it has been assumed that a value comparable with the one declared for the digital inputs is present (the only value provided in the data sheet is equal to 10pF). This value do not affect the behaviour of the filter (if time permits,a different test will be conducted which varies the filter pole value to better evaluate its effectiveness in the acquisition process and on aliasing effects).

With reference to the opamps stability, the presence of a quite large capacitor at the output is not an issue here. Generally, if is applied a $47nF$ capacitor directly at the output of an amplifier, this could dramatically reduce the phase margin (this is valid for both the opamps used). In this situation, an input step signal can create an oscillating behaviour at the output with excessive overshoot,

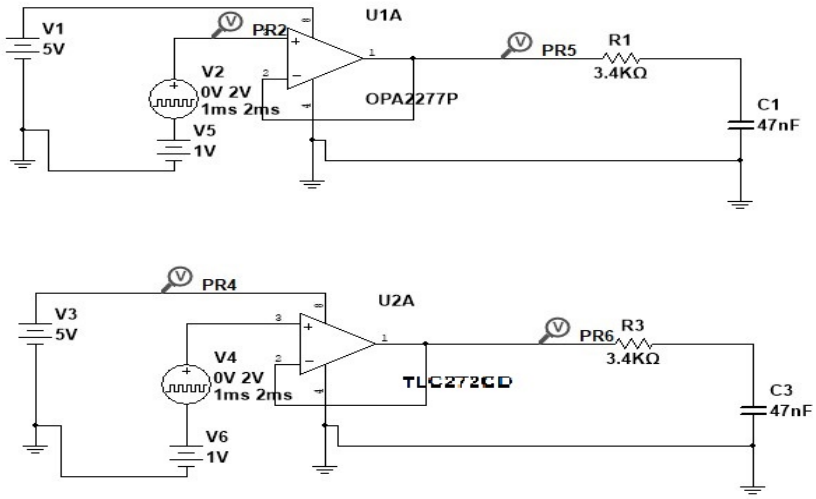


Figure 3.9: Test circuits for the step response

ringing, and excessive settle time. Some simulations have been performed with the available SPICE models, see figure 3.9 for the configuration, confirming this PM reduction that brings a relevant oscillating behaviour at the output (see figure 3.10 for TLC272, and figure 3.12 for OPA2277). The OPA2277 still shows a tolerable response that confirms what was declared in the data sheet [32]. Notice, instead, the limited output swing, which is clearly visible with 0V DC input bias and for a unipolar voltage supply.

These effects are, in this project, dumped due to the presence of the high valued resistor ($3.4K\Omega$) of the filter that nullifies the drawbacks of the high capacitance. Indeed, the presence of the resistor can stabilize the amplifiers since a couple of zeros and poles are introduced to the feedback networks (see [38, 39, 37]) whose frequencies only differ in the value of the output impedances of the amplifiers (almost 30Ω for the OPA). The pole and zero presence cancels the effects of the phase reduction, maintaining a safe PM (also for this case, a simulation with a SPICE model has been performed to verify the behaviour, see figure 3.11 for TLC272 and 3.13 for the OPA2277). It is worth being noted, that the effect obtained with the filter resistance is exactly comparable to the technique used to stabilize opamps in the presence of a pure capacitive load, where, to obtain a sufficient PM, an R_{iso} is placed in series with to the opamp output (see the articles just cited).

With reference to figure 3.2, the connection between the double amplifier board

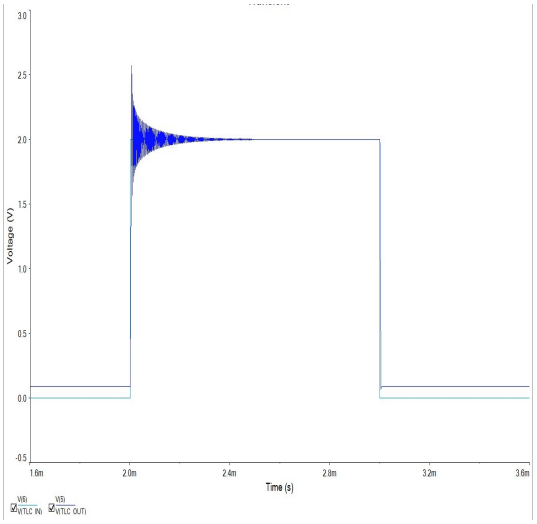


Figure 3.10: TLC272 47nF capacitive load behaviour, 0V input bias

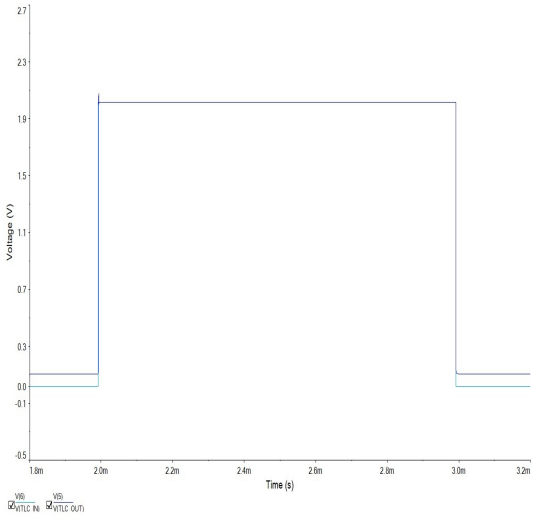


Figure 3.11: TLC272 loaded with a $3.4K\Omega$ series resistance 0V input bias

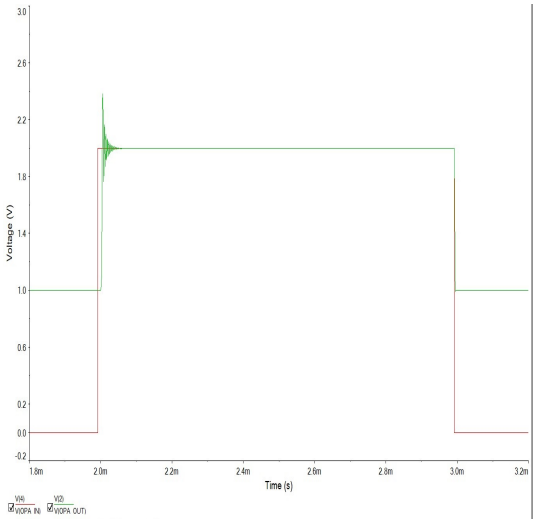


Figure 3.12: OPA2277 47nF capacitive load behaviour, 0V input bias

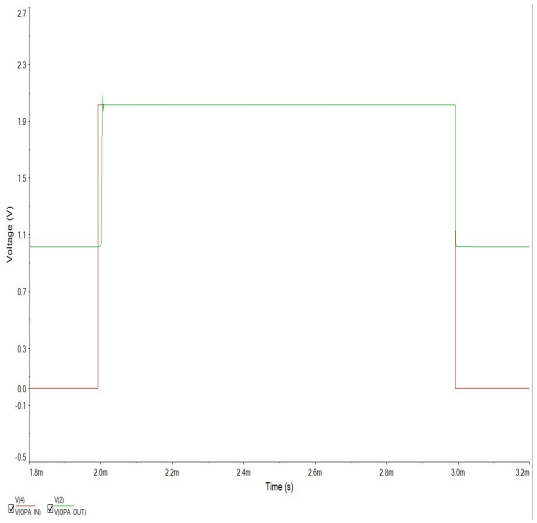


Figure 3.13: OPA2277 loaded with $3.4K\Omega$ series resistance, 0V input bias

and the ADuC has been done through the output pins strip connector located at the bottom left of the board, following 3.4. The connections have been maintained as short as possible and twisting the conductors to avoid an RF coupling to these cables that can cause them to act like antennas.

The GND of the double amplifier board, as illustrated above, is connected to the analog GND of the PCB that hosts the ADuC7061 that will be described in detail in the next section. On the demo board, the analog ground and the digital ground are separate but tied together, see figure 3.14.

3.3 The micro-controller board

In this section the board used to interface the double amplifier board to the PC through the USB will be described. In figure 3.14 the EVAL ADuC7061 MKX evaluation kit can be seen, provided by the same manufacturer that produces the micro-controller, ANALOG DEVICES. Starting from the left, the USB connector can be seen (a mini USB type B) followed by the U3 device, the low dropout, LD, and the ADP3333ARM-2.5Z voltage regulator, that from the 5V provided by the USB gives the correct 2.5V to the ADuC7061 micro-controller. Continuing to the right, there is the programming switch, the FTDI UART-to-USB controller, the ADuC micro-controller, and the reset button. Other components are installed on the PCB for testing purposes in addition to some evaluation software provided by the manufacturer (for example the RTD resistor). As previously explained, the USB gives a 5V power supply referenced to the ground. That ground, in particular, is the same as the personal computer that, in turn, is connected to the ground of the power lines shared with the laboratory equipment. In this way the ground is fixed and cannot be changed. The same 5V from the USB are provided to the LD voltage regulator that feeds the micro-controller. The 2.5V coming out of the regulator is separated into digital and analog ground through an inductance (L2) and a resistor (see figure 6.1 in the appendix). Instead, the digital and analog ground are tied together (see figure 3.14 circled in red).

Also present on the board is the FTDI integrated circuit that permits data transfer from the micro-controller to the USB through the integrated UART controller of the ADuC. As already described, this has been the bottleneck for the choice of the sampling rate. The maximum baud rate was, indeed, limited by the UART speed and not by the FTDI chip or by the USB protocol. Although, as will be seen, the data sheet shows only some standard configurations for the choice of the desired baud rate, the configuration registers can be adjusted to obtain a different and wider set of speeds. Theoretically, setting the ADuC configuration registers as described in the data sheet (see [36]), it is possible to obtain a baud rate of 320,000baud/s that, however, in practice cannot be used. After some com-

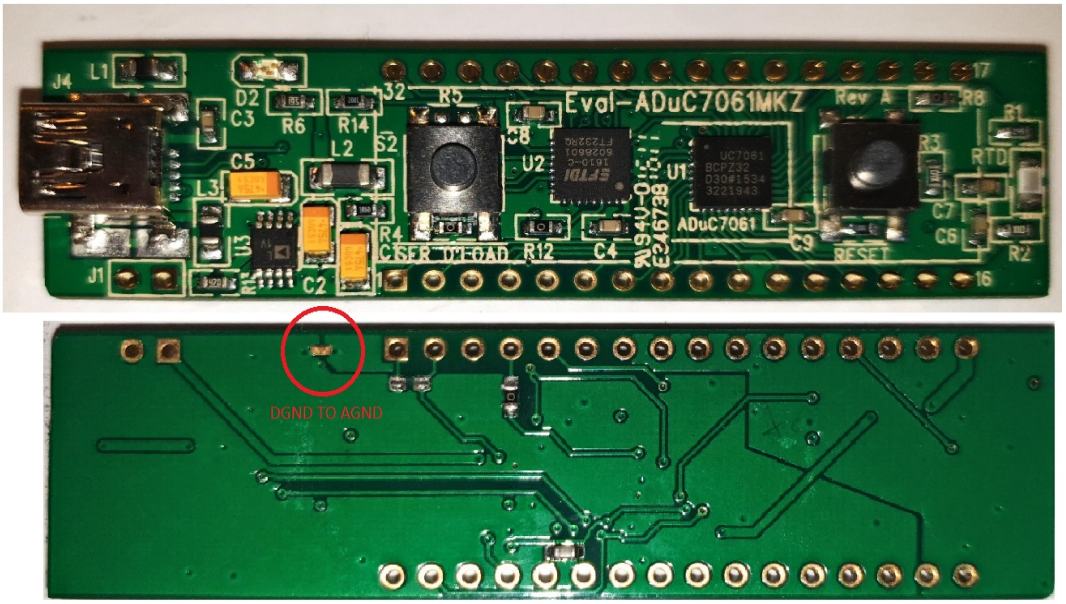


Figure 3.14: The ADuC7061 evaluation board

munication tests, this speed has been discovered to be imprecise and to have a tendency to go out of synchronization. Although not permitted this configuration would have been useful. In fact, this speed could have exploited the complete potential of the ADC, allowing a sampling rate of 8KHz that would have had enough performance to send all the acquired samples to the PC at 8kHz sampling frequency. For what has been discussed, the final speed chosen was 230,400baud/sec. This, in turn, has forced the reduction of the sampling frequency to 4,000Hz (see figure 3.15).

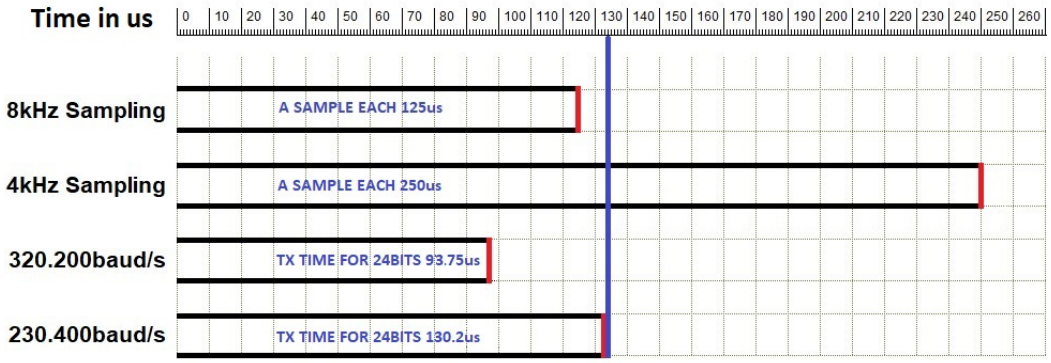


Figure 3.15: The UART timing

The choice of the board described above has been made starting eith the evaluation of the ADC required for the experiment. This had to have separate and synchronous inputs and, at the same time, had to have a sufficiently high resolution (here 24bits) and a reasonable sampling frequency, compatible with the amplitude of the induced signals and with the frequencies employed in the measurements. In particular, as will be seen in the ADC section, the ADuC7061 has two separate and integrated Sigma-Delta converters. These enable, through a specific firmware sequence, two contemporary readings, one for each channel. There was no need to interpe a multiplexer that would not have permited simultaneous reading of the inputs and, so, would have not permitted testing the efficiency of the proposed theoretical EMI cancellation solution. In fact, an interposed multiplexer would have required, at the very least, a faster ADC.

On the other hand, the EVAL-Board is an easy to use system because all the components required to dialogue with the PC and to provide the adequate power supply are already present. The UART to USB FTDI chip and the USB connector itself, were already configured, reducing many of the problems related to the design of an ad hoc solution. As explained, the drawback of this configuration is the incomplete exploitation of the ADC speed, limited by the integrated UART device inside the ADuC7061. A possible increase of the speed has been investigated since the first analysis. One of the hypothesized solutions was to use an SPI-to-USB converter. This, in fact, would have satisfied the required speed but, at the same time, would have required a specific driver and dealing with the complex architecture of the USB, that would have involved considerable time to correctly implement that kind of a communication system. Moreover,this last solution were adopted, a specifically-designed PCB would have been required, bringing the overall design time to an excessive level (the only easy-to-use, pre-packaged drivers are the human interface device drivers, HID, for other solutions necessitating the writing of specific code and calling for a deeper knowledge of the operating system behaviour). The ADuC board, moreover, does not require particular hardware to be programmed and to test the firmware. The drawbacks are the limited debugging possibilities, indeed, only simulated hardware not dedicate actually to this micro-controller can be used in the provided integrated development environment, IDE. There are no possibilities to use a direct hardware reading from the USB without a JTAG interface (see the Keil μvision technical documentations program for further information). Another benefit of this integrated system is that all the supply voltages are already present and all the required devices are pre-designed without the need to redraw schematics and to make a dedicated PCB with similar characteristics. However, some limited connection modifications, in terms of GND and voltages, are required to adapt the 5V GND referenced coming out of the USB connector, to the double amplifier board designed to work with $\pm 2.5V$.

3.4 The ADuC7061

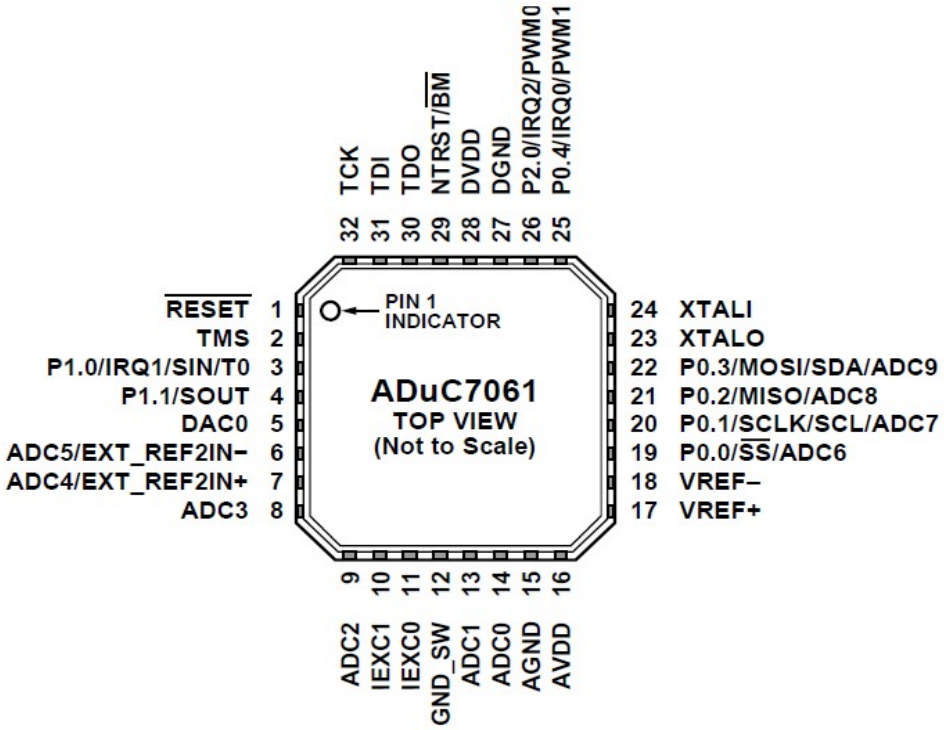


Figure 3.16: The ADuC7061 output pins from [36]

The micro-controller used for the implementation of the technique in [13] is the ADuC7061 from the manufacturer, Analog Devices (see figure 3.17 for all of the devices available on the chip). The principal characteristics of this controller are the supply voltage of 2.5V for both the digital and analog parts (with an absorption of 10mA when both the converters are active) and the presence of two separate ADCs that work at a maximum frequency of 8kHz and with a nominal resolution of 24 bits (the internal Sigma-Delta oversampling frequency is 512kHz).

The micro-controller is an ARM7 type, operating with both 16 or 32 bit instructions at a maximum frequency of 10.24MHz. The clock is provided by an internal programmable oscillator (an external one can be used). The architecture is RISC. An IDE is readily available that permits programming, at least for basic algorithms, with C++ (The Keil μvision IDE can be used in the free version with limited program dimensions and in the full version for a fee). For dedicated programs and for hardware-specific applications, it supports the ASM for ARM7

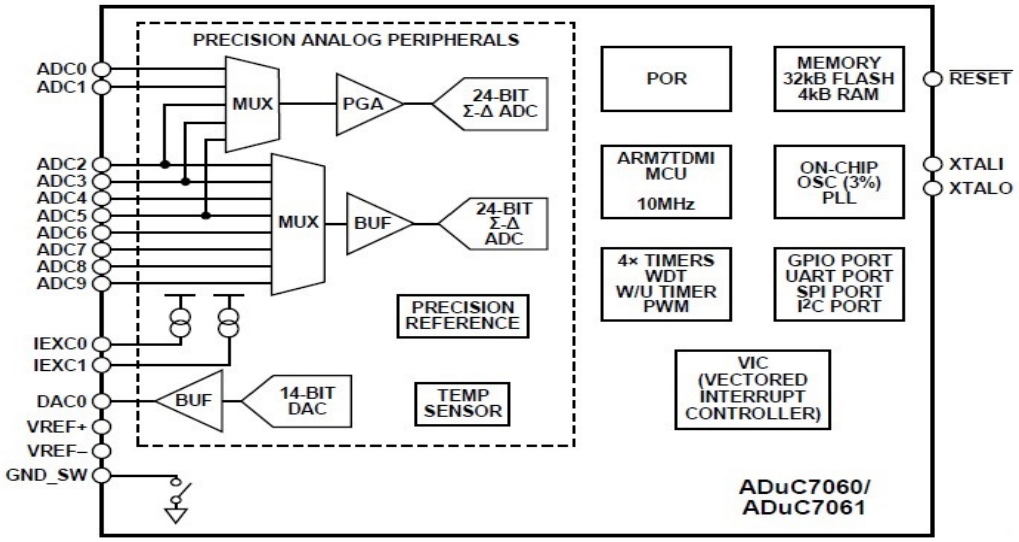


Figure 3.17: The ADuC7061 functional block from [36]

language (for details, see [40]). The latter is used in the initialization routine to set the various memory stack positions and dimensions and to map the available memory (see [36]). The installed memories are 32kB FLASH, with an endurance of almost 10,000 cycles, data retention of 20 years, and 4kB of SRAM.

The micro-controller hosts a 16450-compatible UART (one interrupt for any character sent, FIFO buffer of 1 byte) serial I/O, an I^2C and an SPI communication interface. The first is the only one used in this application. A vectored interrupt controller is available, triggered by both ADC and UART controller events. Other devices present, like digital to analog converters, timers, PWM generators, current references and temperature sensors were not used in this project. The micro-controller can operate in various low power modes with the possibility of choosing which functional blocks must be on and off. This functionality was also not exploited since the micro-controller must operate at maximum performance.

Connections to the outside can be seen in figure 3.16 for the chip itself and in figure 6.1 of the appendix for the connection to the demo board.

3.4.1 The analog to digital converter

The current project requires an ADC that permits synchronous sampling of both channels, as shown in the figure 3.1, and sufficient resolution to capture the small, for high frequency EMI injection, induced offset variation, and the demodulated RF amplitude at the output of the amplifiers.

In general, to convert two input signals at the same time, two sample and hold, S/H, circuits must be driven at the same frequency so that the information is captured at the same $t = t_k$ instant. When only one converter is available, the subsequent S/H stage would require a multiplexer and an analog-to-digital convert that works at a double frequency respecting expectation for each channel. Another problem to take into account in case of Sigma-Delta converters, is the fact that the input of the ADC must switch between two different signals with every sample and that the decimator filter (see figure 3.18), first, requires previous data be cleared and, then, must wait until the new averaged value is provided at the output. Indeed, for a change of the read input, the decimator takes a number of periods

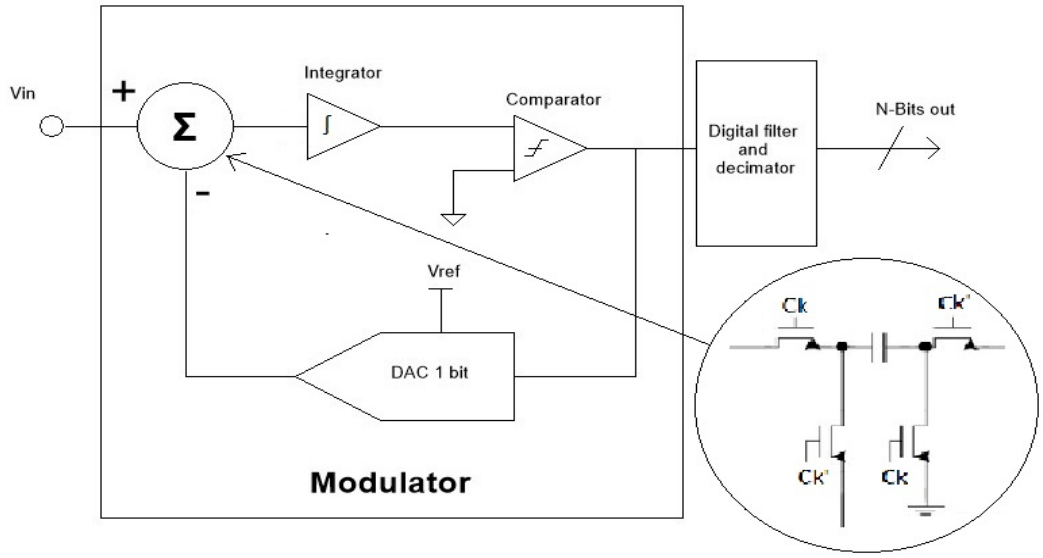


Figure 3.18: Sigma-Delta converter

to stabilize and give a correct value. This implies that it is not always possible to jump from one channel to another for every single sample. To overcome this issue, an even higher frequency, greater than double the minimum needed for the desired application, is required because some of the first samples are corrupt and must be discharged. For example, the ADC of the ADuC7061 used without the chopping technique and without internal averages, has a settle time of $\frac{3}{f_sampling}$,i.e., 3 sampling periods (see the ADC Conversion Rates and Settling Times table in [36]).

In this project, the above-mentioned problem was overcome since there are two ADCs available in parallel with similar performance. In this way, the same schema of figure 3.1 has been exploited without requiring the use of a multiplexer or a high

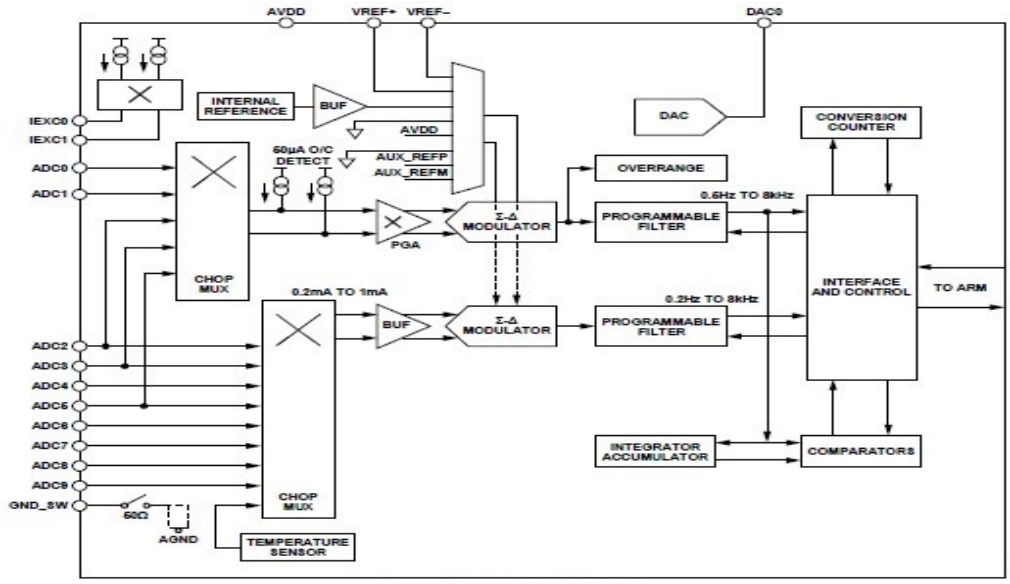


Figure 3.19: The ADuC7061 ADCs from [36]

frequency converter (that can have reduced resolution). The final schema can be seen in figure 3.3.

The equipped ADCs are two Sigma-Delta converters (the primary named, ADC0, and the secondary, ADC1) that intrinsically have the sample and hold (generally, a switched capacitor is present at the input though not indicated in the data sheet, see the figure 3.17 and 3.18). Both of these have a nominal resolution of 24-bits and the possibility of sampling different channels each. The primary Sigma-Delta can operate with 4 channels and the secondary with up to 7 channels (see figure 3.19). In our case, one channel is used for each converter. It is to be noted that some inputs are shared between the two converters. This is because the ADCs inherently work in a differential configuration and can be set in a single-ended configuration. As such, one of the channels is a common reference between the two ADCs (see figure 3.19). The programmable gain amplifier (PGA) for the primary ADC and the buffer, for the secondary, amplify both the corresponding plus and minus inputs.

The presence of the PGA and the buffer between the input pins and the ADCs is useful to adapt the input level at the required situation through configurable registers, and not charge the signal source. Here, the PGA is also configured as a buffer, with gain equal to 1, and in single-ended mode since the outout of the two amplifiers that come from the double amplifier board, are sampled separately and are ground referenced. The maximum properly-working voltage that can be

applied is 1.2V (see [36] table 1). Worthy of note, the presence of the PGA and the buffer can cause some minor voltage problems to arise. Indeed, if it is not directly configured, the programmable amplifier always works for both the plus and minus channels, also the case if the ADC is configured in a single-ended mode. This, in turn, requires that the reference pin not be at exactly 0V, yet it must be at least at 100mV above this value to ensure the correct operation of the buffer. Although not well detailed on the data sheet, the configuration registers can be set in a way to bypass the PGA, or the buffer, of the reference input so that 0V can be applied. This configuration has been adopted for both the gain amplifier and for the buffer.

The converters can be set to give, as output, a unipolar or a complement two data. The same converters can be configured in a differential or in a single-ended configuration. The latter has been adopted.

The ADCs, moreover, can work in normal mode at the maximum sampling frequency of 8kHz and with some noise and offset reduction techniques enabled like the chopping mode and the averaging mode. Automatic ADC offset and PGA gain calibration routines are present. None of these improvements were exploited in this project, to maintain non-manipulated samples as much as possible.

The ADC voltage reference can be internal or external. In this case, the internal one was adopted which is 1.2V $\pm 0.1\%$.

Both the converters are declared to have no missing code and limited input leakage current of 1nA. The primary ADC shows an offset error of $\pm 8\mu V$ with chop mode off. The maximum and minimum allowed input voltage, applied to both the Vin_+ and the Vin_- channels, are from 0.1V to VDD-0.7V. The 0.1V limit on the reference pin has been removed, as previously discussed, with the PGA (buffer) bypass. The standard input voltage with a gain of 1 is 0-1.2V. The secondary Sigma-Delta has an offset error of $\pm 30\mu V$ with chop mode disabled, similar input voltages are required and the 0.1V limit, as for the primary ADC. The latter was overcome exactly as for the primary ADC.

Resolution and RMS noise

From the measurements performed for the OPA2277 opamp and the MCP6V02 in [13], with an injected CW RF from 50MHz to 500MHz, a maximum EMI rejection ratio, EMIRR, (see eq. 3.6), of almost 85dB has been obtained for both the amplifiers.

$$EMIRR = 20log_{10}\frac{V_{EMI,pk}^2}{\Delta V_{off}100mV} \tag{3.6}$$

This corresponds to a $\Delta V_{off} = 56.23\mu V$ for a frequency of 500MHz. At the same time, the output spectrum of the opamps for an injected RF power of 0dBm at 100MHz, modulated by a 1kHz square wave, shows a peak at 1kHz of about -28dBc with respect to the base-band input signal amplitude of 0.5V (almost 4dBm $P_{dB} =$

$10log_{10}(\frac{(0.5V)^2}{2 \cdot 50\Omega})$). This power corresponds, thus, to a -24dBm that, in turn, gives a peak voltage for the demodulated EMI of about 20mV ($V_P = \sqrt{P_{lin} \cdot 2 \cdot 50\Omega}$, that corresponds to the RMS value being an alternate square wave). The critical value to be sampled is, therefore, the offset voltage.

From the data sheet of the ADuC ([36]), the output RMS noise for the primary ADC is declared to be, in case chop mode disabled, $8.54\mu V$ for a 1kHz sampling frequency and $54.97\mu V$ for a sampling frequency of 8kHz. Deriving the value for the 4kHz sampling frequency, with a linear approximation, the root mean square, RMS, noise at output is about $28.44\mu V_{rms}$ (equal to $6.6 \cdot 28.44\mu V = 187.70\mu V_{pp}$ with a confidence of 99.9%). This last RMS value is comparable to the value of the induced offset obtained from the previous study ($\Delta V_{off} = 56.23\mu V = V_{off,rms}$ see [13]) and some orders of magnitude below the demodulated EMI signal. Similar values are declared for the secondary ADC.

Following the data sheet one can compute the effective number of bits as $n_{eff} = log_2(\frac{Full\_Scale\_Range}{RMS\_noise}) = log_2(\frac{1.2V}{28.44\mu V}) = 15.36bits$.

3.4.2 The UART interface

The UART interface present inside the ADuC7061 is a 16450-compatible, therefore, for each character sent (received), it enables an interrupt to the microcontroller. The buffer is limited to 1 byte. In this project, no particular information is received from the PC except for a start and stop character to enable and suspend the sampling. The transmission has the limits already discussed.

The UART can work with two different baud rate generation methods: a normal divider and a fractional divider (see figure 3.20).

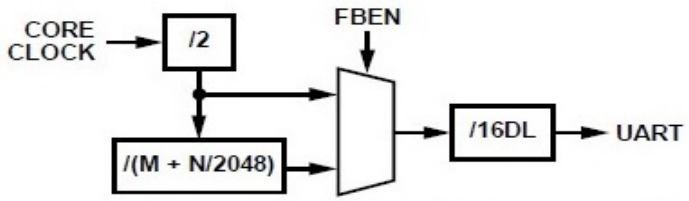


Table 88. Baud Rate Using the Standard Baud Rate Generator

Baud Rate	DL	Actual Baud Rate	% Error
9600	0x21	9696	1.01%
19,200	0x11	18,824	1.96%
115,200	0x3	106,667	7.41%

Table 89. Baud Rate Using the Fractional Baud Rate Generator

Baud Rate	DL	M	N	Actual Baud Rate	% Error
9600	0x21	1	21	9598.55	0.015%
19,200	0x10	1	85	19,203	0.015%
115,200	0x2	1	796	115,218	0.015%

Figure 3.20: The ADuC7061 UART block from [36]

The fractional divider allows the attainment of more precise speed at the cost of configuring more registers. The tables in figure 3.20, taken from [36], show some common configurations for the standard and fractional methods. The formulas to compute the wanted speed values are the following (eq. 3.7), where DL, M, and N are the configuration registers to be set:

$$\begin{aligned} \text{Standard Baud Rate} &= \frac{10.24MHz}{16 \cdot 2 \cdot DL} \\ \text{Fractional Baud Rate} &= \frac{10.24MHz}{16 \cdot DL \cdot 2 \cdot (M + \frac{N}{2048})} \end{aligned} \tag{3.7}$$

Looking at these equations, a vast set of baud rates can theoretically be obtained, however, limitations have been encountered (see the configuration chapter for details).

3.4.3 The memory and the control registers

The ADuC memory is divided into 32kB FLASH memory and 4kB SRAM memory. The read and write access to the RAM memory is done in words of 16bits and erasing is done in pages of 512bytes. All of the operations can be synchronized by interrupts. The time required for writing is about $50\mu s$ and to erase a page is about $24ms$.

The FLASH has a reserve of 2kB, used internally for the kernel of the micro-controller that deals with the post-reset operations like the configuration routine that downloads the factory calibration values to the device registers and that makes the initial program routine starts at the 0x00000000h address. This part of the memory cannot be exploited and is hidden from the user.

The SRAM is divided into 32bit arrays and can be accessed in 8 to 32 bit segments.

All of the available memory and configuration registers are accessible as seen in figure 3.21. The memory-mapped registers, MMRs, are the configuration registers that provide a connection to the integrated ADuC peripherals, like UART, ADCs, interrupt controller, and that permit setting of the required parameters which regulate the operations to be performed.

At start-up, the micro-controller executes the internal firmware (kernels) and jumps to the starting location at the 0x00000000h address where the FLASH (at address 0x00080000) is mirrored. This make it possible to access the non-volatile memory through these lower addresses (0x00000000 to 0x00007FFFF). In this location, after the reset, the program code can execute other remapping operations. Indeed, it is possible to remap the position on the SRAM for any purpose, data, or program. The ADuC can, in fact, execute instructions of 16 or 32bits dimension. The 16bits instruction improves performance but increases the number of machine operations to do. At the same time, not all the operations can be executed in 16bit mode.

In figure 3.23, the detail of a properly configured MMR can be seen with the appropriate addresses used to obtain the desired function. In this project, the

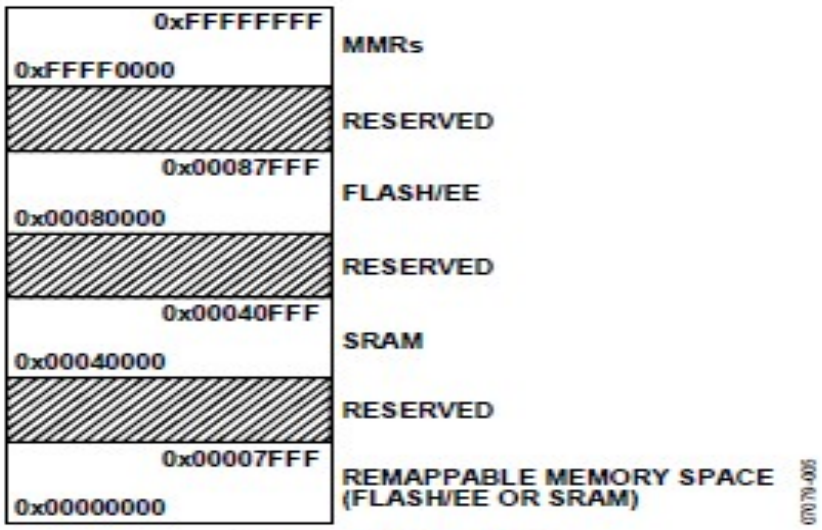


Figure 3.21: The ADuC7061 memory allocation from [36]

following registers will be used. Starting from the top, the GPIO MMR will activate the correct input and output pins for the UART and the ADCs. The UART register is to configure the speed of the protocol, the number of bits transmitted, the number of stop bits, the parity of the transmission, and how the interrupts are generated for data transmission and reception. Then will be used the ADC MMR to configure the active converters, the sampling frequency, the decimator filter (SINC3), the chopped and average modes, and to enable the interrupts when the samples are ready. The oscillator register is to set the maximum speed and internal clock generation. The remap register, at the beginning, and the interrupt register configures how must to handle the different peripheral generated interrupts. A detailed explanation of the configurations can be seen in the next section.

3.5 The ADuC7061 configurations and the overall algorithm

This section will describe, in detail, all the configurations adopted for the current project, showing the parameters stored in the MMR registers and the relevant associated functions (see figure 3.23 for the available registers). In the subsequent sections, instead, the core of the implemented algorithm, in charge of acquiring and processing the sampled data, and how the micro-controller communicates with the PC to transfer the computed values will be elaborated.

The overall flowchart of the program can be seen in figure 3.22 (see also the program code in the appendix 6.3.1). It starts with the basic configurations that will be discussed next, follows with the execution of the core program, and ends with the transmission of the processed data.

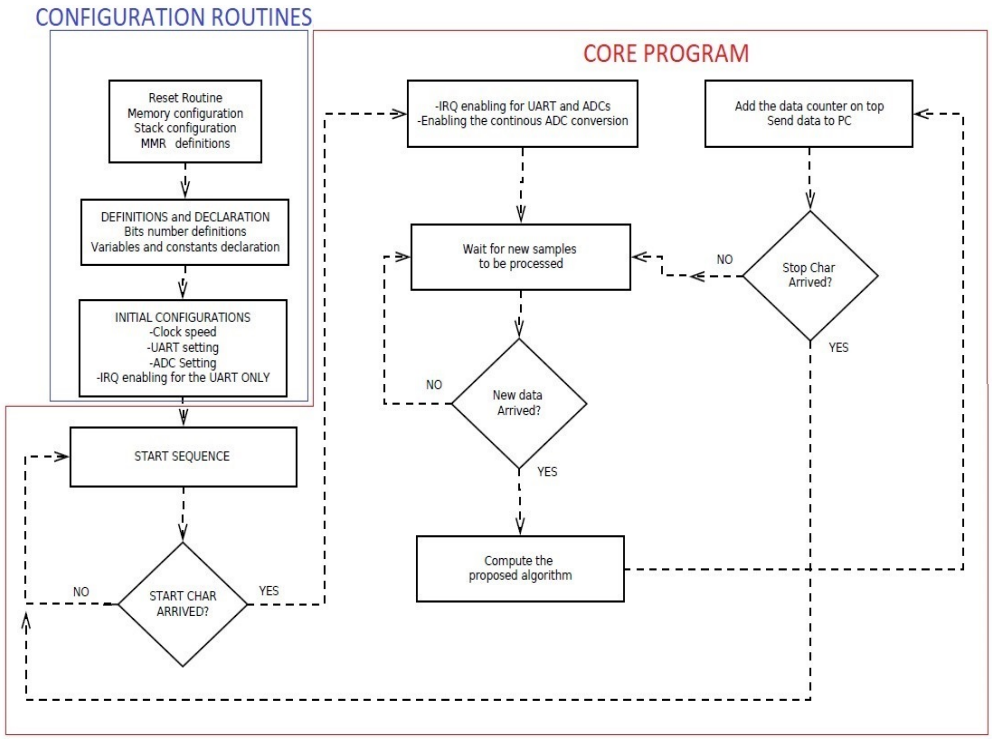


Figure 3.22: The program flow chart

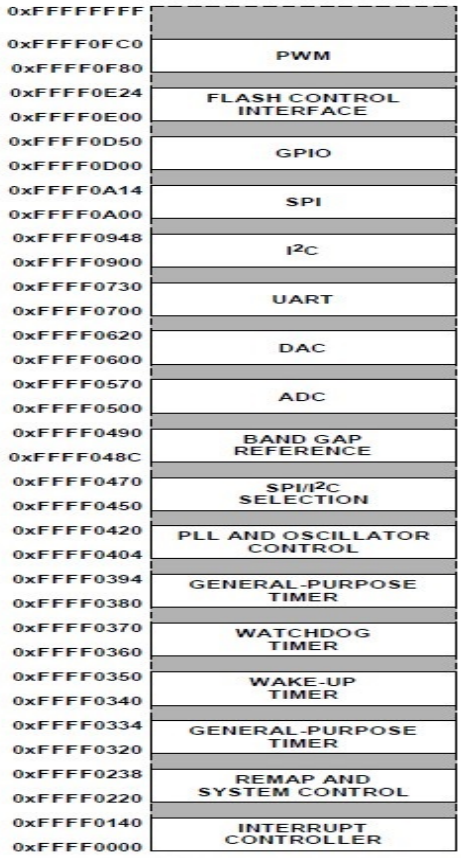


Figure 3.23: The ADuC7061 control registers from [36]

The startup routine

After a reset or after power-on, the integrated kernel of the ADuC7061 executes the initialization routine by charging the factory parameters to all the peripheral MMRs. After completing this task, it jumps to the starting user routine. In this routine, which is already almost written, the memory position of the interrupt codes are set and the stack dimensions for the user program, for the interrupt handlers codes and for the various supported operating modes (user mode, IRQ mode, FIQ mode, abort mode; here only the user mode and the IRQ mode are required) are allocated. Moreover, some constants like the base address of the MMRs (see figures 3.21 and 3.23 for reference to the addresses) are defined, and the space for the after-reset program entry point is reserved. Finally, a jump is made to the main routine to run the actual job.

3.5.1 Configuration of the oscillator and the active devices

Upon entering to the main program, the first configurations required are those concerning the phase locked loop, PLL, and the active devices. The program sets the internal oscillator and the maximum speed of 10.24MHz. At the same time, the UART controller is enabled and all the others peripherals (I^2C, SPI, PWM) are disabled, to avoid energy wasting, since they are not always in use.

The above configurations are enabled through the PLL and Oscillator MMR, along with other sub-registers (i.e., the POWCONx registers) complying with the data sheet ([36]). Of note is that these types of configurations are critical, thus, in order to prevent errors, there is a compulsory writing procedure. The initial writing of a specific determined register with specific data must be executed. Then the required configuration register must be written and, finally, a specific write operation of another specific register must be executed again. If the procedure is not correctly followed, no effects are obtained on the desired configuration register. This is imposed to preserve micro-controller integrity.

3.5.2 The UART

A specific routine is dedicated to the configuration of the UART parameters. Following the program flow, the first to be set are the output pins that communicate with the FTDI UART-to-USB chip through writing of the MMR UART sub-register called GP1CON. The selected input-output pins are the P1.0 and P1.1 that correspond to pin numbers 3 and 4 of the integrated circuit. The transmission speed is set to 230,400baud/s, for the reasons already explained, adopting the fractional divider. Notice that this rate can only be obtained with the fractional mode configuration. Also, in this case, the required writing scheme is done in three steps, to avoid mistakes as explained in the startup routine subsection. The involved registers are now COMDIV1 and COMDIV2.

The COMCON0 MMR sub-register is used to impose an 8bit transmission packet with only one bit of stop. This choice has been made to maximize the transmitted byte per unit of time, also taking into account that communication with the external FTDI chip is not so critical and does not require particular precautions, considering also the limited distance between the two integrated circuits.

Finally the UART interrupts are configured to acitvate through the COMIEN0 MMR in case of receipt, RX, register full, and those of transmission, TX, buffer empty. As will be explained in the algorithm section, these interrupts are handled from the beginning to permit the transfer of the online sequence to the PC, which informs that the ADuC is ready to work and receive the start byte that enables the ADC sampling. This activation is done through the interrupt MMR sub-register called IRQEN (the same register is also used to enable the handling of the ADC

IRQ after the start signal).

As discussed in the ADuC description, UART subsection, a conspicuous set of baud rates can be obtained by exploiting equation 3.7. In testing the functionalities of the UART, it has, however, emerged that the simple divider is not as precise as expected. Indeed, also for standard baud rates i.e., 115,200baud/s, the synchronism between the ADuC and the PC was obtained with a reduced value with respect to the estimate configured (instead of 115,200 a lower value had to be used to correctly communicate). An acceptable result with the fractional divider at the same speed has, instead, been obtained .

It is worth noting that, from the fractional divider formula, a maximum baud rate of 320,200baud/s can, theoretically, be obtained. This value, after some tests, has turned out to be unstable. After continuous transmission, the data were lost by the PC, obtaining only incorrect data. For this reason, the value used for this project is 230,400baud/s that has shown good results. The drawback, as mentioned, was the need to reduce the sampling frequency to 4kHz. The protocol used for the transmission is 8bits of data, 1 stop bit and no parity bit. Considering the start bit, as well, the transmission has an efficacy of 80% (10bits must be transmitted to have 8bits of data). By computing the number of bits required related to the sampling speed, it can be obtained (eq. 3.8):

$$\begin{aligned}
&\#\text{bits @ 8KHz} &&= 24bits \cdot 8000Hz = 192,000 \quad bits/s \\
&\#\text{bits @ 4KHz} &&= 24bits \cdot 4000Hz = 96,000 \quad bits/s \\
&\#\text{bits @ 4KHz+ctlr} &&= 32bits \cdot 4000Hz = 128,000 \quad bits/s \\
&\#\text{Tx data bits @ 230,400} &&= 230,400baud/s \cdot 0.8 = 184,320 \quad bits/s \\
&\#\text{Tx data bits @ 320,200} &&= 320,200baud/s \cdot 0.8 = 256,160 \quad bits/s
\end{aligned} \tag{3.8}$$

From these equations, it can be seen that for a 320,200 baud rate a sampling speed of 8kHz would have been supported. On the other hand, a sampling frequency of 4kHz generates 96,000 bits per second and as such leaves room to transmit an additional byte that can be used as control (see the fourth equation of 3.8). The final solution is adopted in this project. The control byte has been used, in fact, to send a sample counter to check if the correct sequence of samples was transmitted correctly or if some were lost (see the algorithm section for details).

3.5.3 The ADC setting parameters

The ADCs, as discussed, work intrinsically in a differential way, therefore, the PGA and the buffer are in this configuration too. If the latter two are configured in single ended mode, a pin of the input must be dedicated to the reference voltage

that, in this case is GND connected to the double amplifier board, and it must be shared with both the converters. In our case, this pin is ADC5 that is in common with the primary and secondary ADCs. The chosen input channels are, instead, ADC2 for the primary and ADC6 for the secondary. This choice has also been made with reference to the available pins on EVAL board that correspond to the free pins called J2-9, J2-19 and J2-6, (see figure 6.1 in the appendix).

The presence of the PGA and the buffer between the input pins and the ADCs has required some extra configurations. Indeed, worthly of note is that, due to the presence of these components, some minimal voltage problems can arise. If not directly configured, the standard settings of the programmable amplifier and buffer make them always enabled for both the plus and minus channels. This is also true if the ADCs are configured in a single-ended mode with the minus at ground, because the system is intrinsically differential (see figure 3.19). In turn, it requires that the reference pin not be at exactly 0V but must be at least 100mV above this value to ensure the correct operation of the buffer (PGA). Although this situation is not well-detailed on the data sheet, the configuration registers can be set in a way that bypasses the PGA, or the buffer, of the reference input so that 0V can be applied correctly without requiring a DC bias. The bypassed configuration has been adopted for both the gain amplifier, configured with a gain of +1, and the buffer in this project

As discussed in the previous section, the chosen operating sampling frequency is 4000Hz.

The ADC setting is done through the dedicated MMR sub-registers and the interrupt MMR. The required amplifier (buffer) configuration is done with the ADC0CON and ADC1CON MMRs. For the primary ADC, the input pins are ADC2 and ADC5, the latter as a reference. Moreover, the data is set to be provided as a unipolar value (0-2^{24} and not in two complement), and an internal ADC reference is imposed. For the primary converter, the PGA present at the reference input (ADC5) is bypassed by standard configuration when it works as a voltage follower. The secondary converter has similar configurations, except for the buffer bypass that must be set directly. To be noticed, for this last ADC it is also possible to bypass the buffer at the positive input, if required. This option has not been used here because a similar configuration is not available for the primary ADC. This allows the same topology to be adopted for both the positive inputs.

The ADCs are configured to not exploit averaging, used to obtain a noise free sampling, and the chopping techniques, used, instead, to reduce considerably the offset by inverting constantly the input reading (see the technical report for details [36] about both methods).

ADC1 (the secondary) shares some of its input pins with other devices (see [36]). This project, for PCB reasons, requires the use of these common inputs be-

cause they are readily available on the ADUC-EVAL board. Thus, the GP0CON1 register has been configured to assign the pins named P0.0 and P0.3 as ADC1 inputs. This particular configuration is not well-described in the data sheet and has been highly investigated (the data sheet is really not clear about this configuration). Without the correct parameters written on the above mentioned MMR, ADC1 does not work properly. The data returned is corrupted but, in some way, is correlated to the ADC0 input. This is a problem that, in the initial stage, suggested that the converter was damaged when it was not.

Finally, the oversampling frequency of 512KHz and the continuous conversion mode are enabled. The latter permits exploitation of the interrupts without requiring a manual start for the next sampling phase i.e., after an acquired sample reading, done in the data register, it automatically starts the next sample conversion. This activation of the ADC is only done after the start character is received from the PC. Prior to this, the ADCs are in IDLE mode for the standard configuration at start up.

It needs to be highlighted that the ADuC can exploit a gain and offset calibration of the internal routine. This procedure must be repeated from time to time. In our case, no calibration was done to maintain the possible source of error unaltered.

3.6 The core algorithm

This section describes the operation executed by the core program after the initial configurations described in the previous section (see figure 3.22 for the flow chart).

As described above, the first operations done after the reset are the memory allocation for stacks, the code allocation, and the MMR definitions. Then, the execution jumps to the main program where, for convenience, a set of constants and variables are defined that will be used during the overall program, once the require configurations have been completed the core program starts.

Of note, since the beginning, is that the ADCs provide 24bit data that, however, is stored in a 32bit register. Therefore, when the acquired sample is read, the first 24 least significant bits, LSB, are valid and the others contain only zeros. Moreover, the output of the ADCs has been set to provide only positive integer numbers (unipolar data), so the returned data can be saved in a long int variable (32bits wide).

Next, the code that is employed in a typical application will be described. This code, indeed, provides the PC with the processed data only, and no information about the single sampled channels is present. During the laboratory test, the code was modified to permit sending the data to the PC in sequence: the first channel

read, the second channel and, finally, the elaborated data. This modification will be further illustrated in a successive section.

In a first draft of the code, the computation of equation 3.3, reported here for ease, (eq. 3.9), has been done through floating point variables.

$$y_k(\alpha) = \alpha y_{1,k} + (1 - \alpha) y_{2,k} \tag{3.9}$$

This solution is simple to be implemented and does not require particular calculations to express the constant values α and $1-\alpha$ that could be declared directly as "Constant_Name=Decimal_Value". These definitions were, indeed, easily done as $ALPHA = 0.3846154$ and $NALPHA = 1 - ALPHA$. The Keil IDE uses for the floating point variables was the IEEE-754 standard in single precision of 32bits (see figure 3.24 where the most significant bit, MSB, corresponds to the sign, the next 8 bits to the biased exponent, and the last 23bits to the mantissa, where the first integer value equal to 1 is implicit and not stored).

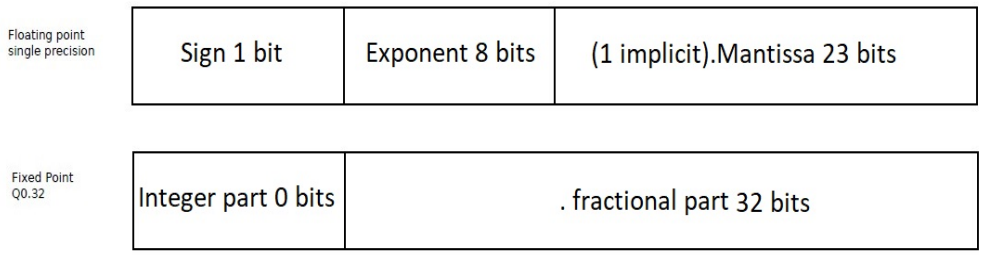


Figure 3.24: Floating point and fixed point variable structure used for alpha

The ALPHA constant represented in this way, corresponds to an approximated decimal value of almost $ALPHA = 0.3846153914....$ that, expressed in hexadecimal, is 0x3ec4ec4f, because this value cannot be exactly represented in floating point. The error due to the conversion is of the order of -8.5^{-9}. In this case,in fact, the mantissa is formed of 23 bits and has a precision of 2^{-23}.

The multiplication required by the algorithm is done first and the results are always rounded to the number of bits of the mantissa (23bits), keeping a result that is still a floating point of 32bits. So, for example, the multiplication of a 24bit ADC sample $y_{1,k}$, (that can be stored in exact form in a floating point variable since the maximum value of the sample is expressed as 24bits at 1), with the alpha

constant is approximated at 23bits. The next sum operation requires the same approximation. At the end of the overall computation, three rounding operations have been done. This behaviour is, actually, not a limitation in our case, because all the fractional parts of the results are neglected but, generally worth being considered.

Also needing consideration is that after the computation required for the exploitation of the technique, the floating point value obtained requires a casting operation to convert the 32bit floating data to 32bit integer data (an long int variable). This conversion implies another approximation that, however, has been proven to not cause issues because the fractional part is always removed with limited effect on the result. Indeed, with reference to the ADC section, the relevant bits are not exactly 24 but less i.e., the equivalent number of bits, ENOB (almost 15).

Of particular note, the ADuC7061 does not have a dedicated floating point unit, so the C++ code must "simulate" the floating point operations using the available machine instructions. This was investigated during the debugging phase, and it emerged that the floating-point calculations increase the program code dimensions, also observed during the compilation stage. Studying the problem in depth, the compiler, in this case, adds a set of standard program functions that carry out, with the available micro-controller instructions, the computation of the operations between floating point numbers (computation between the mantissa parts and the exponential parts). Due to this fact, the time required for the computations has increased with the increasing number of operations required.

For these reasons, an alternative solution has been adopted. In particular, the constants have been converted into a fixed point format Q0.32 of 32bits (see figure 3.24) and then stored in a 64bit format (on int long long type) ready to be used in the following computation. The $ALPHA = 0x627627A4$ and $NALPHA = 1 - ALPHA = 0x9D89D85B$ constants have been, thus, obtained with a precision of $2^{-}32$ that is higher than the one obtained with the floating point constants. The exploitation of the required technique, in this case, is straightforward, because standard operations are executed (integer sums and multiplications). This, in turn, has permitted a reduction of machine instructions and an increase in speed. Moreover, because both the 24bit data samples and 32bit constants(α and $1 - \alpha$) are stored in 64bit variables, the multiplications and the sums can be done without the need for intermediate approximations (they do not overflow with these dimensions). The only one required is at the end of the calculation when the fractional part is removed (rounding to the nearest schema, see the program) and the data is stored in a smaller 32bit long int variable, ready to be sent to the PC.

The drawback of the fixed point calculus is that the constants must be computed each time in the programming phase (or, at least, execute a conversion

routine at start-up) and not entered directly in the code. This has been time-consuming especially during the measurements done to evaluate the correct α to be used.

Altogether, the fixed point computation does not significantly increase the accuracy (from the tests performed, most of the results of the floating point and the fixed point calculus coincide), but a benefit is obtained in terms of computation time and energy reduction that should be considered, especially if this technique must be adopted in an integrated or in a battery-supplied system view. The disadvantage is the requirement of a more complicated computation of the constants and the need to check and ensure, from the beginning, that the operations will be executed successfully.

The execution of the main code of the program starts by setting the desired clock speed, (maximum of 10.24MHz with internal clocking), and making the required configurations for the UART and the ADCs (see the previous section for the parameters used). Then, it enables the handling of the IRQ for the UART only. The ADCs, at this time, are still in IDLE mode and do not sample.

After this, a starting sequence is executed (see figure 3.22). In this routine, the micro-controller sends a string to the PC to inform that the system is online, ready to operate, and waiting for a start character. When the start byte is sent from the personal computer, the program enables the IRQ for the ADCs and keeps the UART active to send and receive interrupts. This is necessary to correctly detect the sampled data and to evaluate if a stop execution byte is sent from the PC. Afterwards, the ADCs pass from IDLE to continuous sampling mode.

At this point, the program enters into a loop waiting for the converted sample. When ADC0 (remember that ADC0 and ADC1 work in parallel and that the data arrive at the same time instant) has acquired the signal sample, it sends an interrupt request that is managed by the micro-controller. The main execution stops and jumps to the IRQ handler routine. This routine determines what kind of IRQ has stopped the normal execution and sets a specific variable for each case. The latter will be detected by the main program that will execute the required code.

For the analog-to-digital converters, the interrupt request recognized comes from ADC0 only when the sample is ready to be provided. At the same time, the sample of the secondary ADC is ready. After the interrupt request, the handler routine reads the data from the ADC data registers. To be noted, a particular sequence must be respected to not lose the samples of the secondary ADC. Indeed, after reading the ADC0 data, the register that contains the information is automatically cleared which triggers another sampling phase. So, to avoid to losing data, a reading of ADC2 is done first (this does not trigger the next phase) and

then, the primary one. This order is compulsory to obtain a synchronized reading of both channels, since the interaction with the ADC1 data register triggers the next phase of sampling for both the converters and resets the control registers.

The interrupt handler routine stores the data into two long int variables that are visible from the main program. For test purposes and to ensure that the working speed was sufficient, after this operation two 1 byte counters are incremented by 1 to keep track of the number of samples acquired. Taking into account the UART speed (see the UART section and configurations) of 230,400baud/s, the transmission of the 24bits to the PC leaves a byte free. Exactly in its space, one of these counters is placed (in the modified program used in the laboratory, it is put the number of the channel transmitted in this place).

After this phase, the program jumps to the computation function where the discussed theoretical technique is exploited (see eq. 3.9) with the value of $\alpha = 0.3846154$ and $1 - \alpha$ expressed, as seen above, in 32bit fixed point and stored in a 64bit variable to avoid overflow during multiplication and sums. Though it is not necessary to save the constants and the samples in a 64bit variable, this has been done to exactly control the casting operation that would otherwise be executed implicitly by the compiler. The result of the calculation is, at the same, stored in a 64bit variable. After the computation, the result obtained is rounded with a "to the nearest" scheme, with a simple addition of +1 to the first fractional bit and the truncation to the desired number of bits (32). The result is returned to the main program in the form of long int variable (32bits).

In the main program, the data is, thus, disposed in 32bits, 24bits are for the real data and 1 byte is for the counter (or for the channel tag), and sent through the UART (see the next section).

After the transmission, the program checks, with the help of interrupts, whether or not a stop character has been sent from the PC. If nothing has arrived, it continues waiting for the next couple of samples, otherwise, it stops the execution and returns to the initial start sequence to wait for a new start character from the USB.

During the laboratory measurement, to compare the values obtained by the technique discussed with the distorted outputs of the single amplifiers, the program has been modified as in figure 3.25.

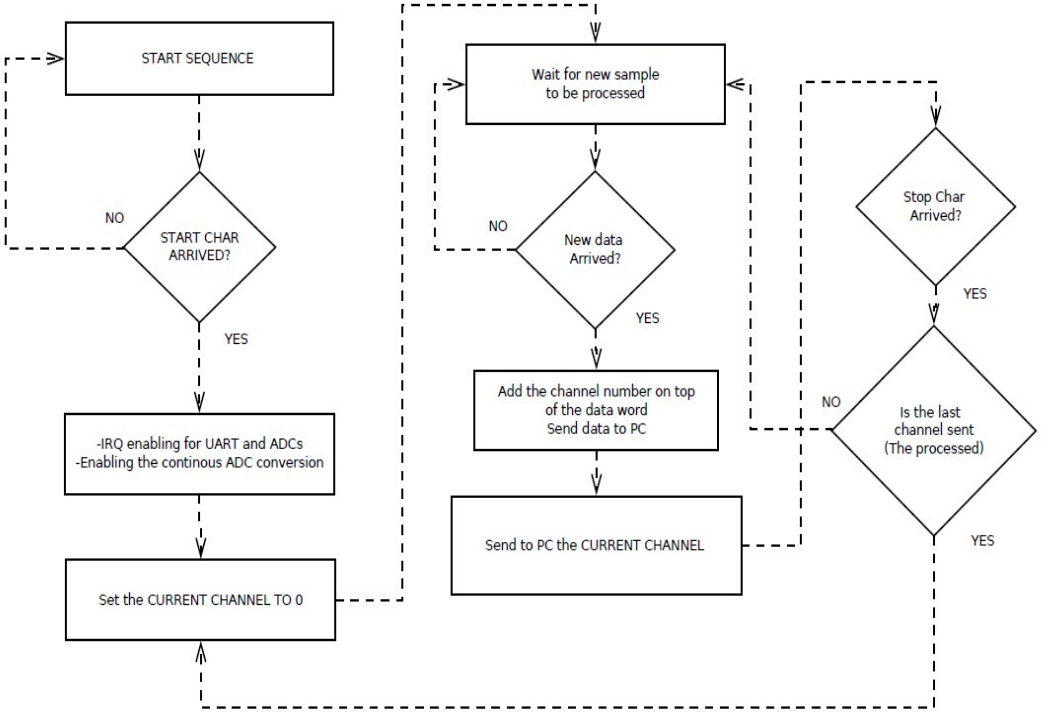


Figure 3.25: The modified program flow chart used in the laboratory

The execution is similar to the previous one but, for each start byte received from the PC, the micro-controller sends only one channel at a time. So, for the first start received, it sampled the OPA2277 channel and sent it to the PC (on top of the byte, the number 1 is present and used as tag to check for the correctness of the transmission). Then, the PC sent the stop byte and, after a while, a subsequent start byte. Now, the ADuC samples TLC272 and sends it to the PC. Finally, for the next start, the micro-controller samples both the channels, executes the calculus required for the technique, and sends the result to the PC.

This modification have been required to evaluate all the signals from the PC, avoiding the use of external equipment (see 6.3.2 in the appendix). Obviously, the measurements are not done at the same time instant, but this does not influence the overall result.

3.7 The serial transfer to PC

The data just computed by the dedicated routine is returned to the main program in the form of a 32Bit variable. The 24 least significant bits, LSB, are of interest information, and the 16 MSB are free spaces that have been used for control purposes. As already discussed, during the execution of the interrupt handler routine a counter is incremented for each sample acquired (and zeroed it reaches its maximum value). This counter is passed to the main program with the effective data. Here, with a bit mask operation, a 32bit word is composed in this way: the 8MSB is one counter and the 24LSB are the data. The entire word is then sent to the PC (see figure 3.26.

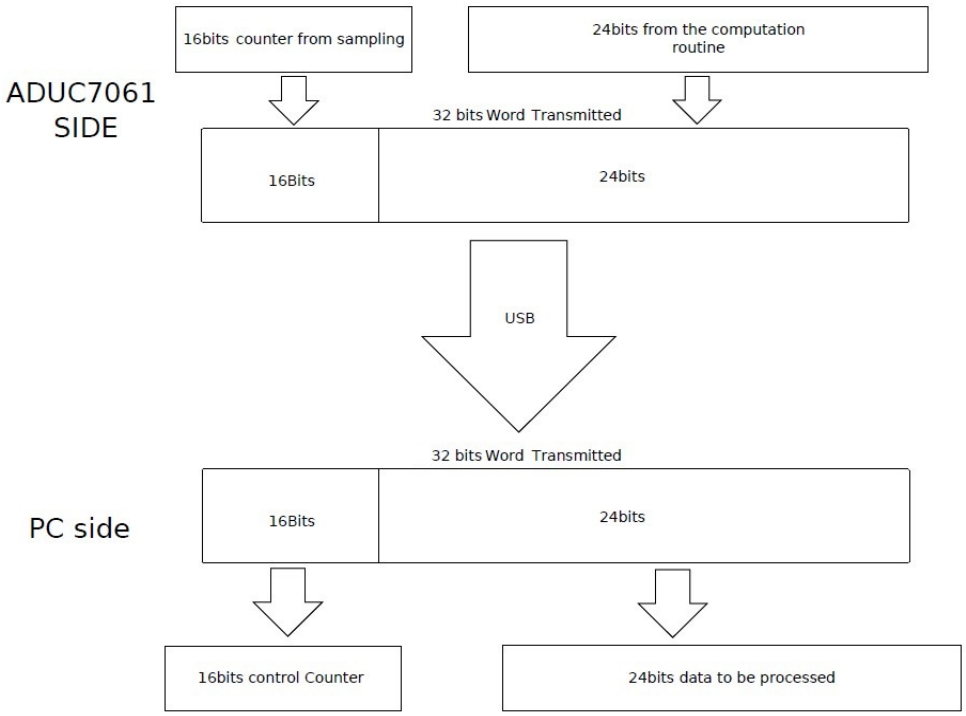


Figure 3.26: The data format transmitted to the PC

This procedure has been added because, using a transmitted data rate of 230,400 bauds/s, there was enough room for an additional byte to be exploited. This permits, on the PC side, evaluating if the board has transmitted all the acquired samples or not. For verification, indeed, with increasing sampling speed not all the samples are transmitted correctly and some are lost.

On the PC side, communication with the ADuC is implemented with the aid of

Matlab. Here, the program opens a serial connection port (COMx) and sends the correct start character. The reading procedure is simpler than the transmission, because, a reading dimensions of exactly 32bits at time can be used. It is not, indeed, required to read a byte per byte. From the words grabbed from the serial port the counter bytes (or the channel tag) and the data are, then, extracted and, both, are stored in separate vectors. The obtained data requires multiplication by a constant value of $M_{factor} = 1200/16777215$ that permits conversion of the binary number to the real voltage acquired from the ADCs. This number is obtained by dividing the maximum allowed input of the ADC (1.2V) by the corresponding value in terms of bits (24bits all at 1, in hexadecimal 0XFFFFFF) that is exaclty $2^{24} - 1 = 16,777,216$.

The next operations are those required to compute the spectrum of the treated signal and those to obtain the signal-to-noise distortion ratio, SNDR, and the spurious free dynamic range, SFDR, whose expressions are shown in equation 3.10

$$\begin{aligned} SNDR &= 10Log_{10}\frac{P_{sin}}{Ptotal - Psin} \\ SFDR &= 10Log_{10}\frac{P_{sin}}{Pmax, spur} \end{aligned} \tag{3.10}$$

3.8 Functionality tests

After the debugging of the program in a simulated way, the functionalities of the ADuC board have been tested in the laboratory. A wave generator RIGOL DG4162 has been connected to the ADuC board at the ADC inputs. Different signals have been injected, respecting the input range of the converters with the aim of testing the correct sampling (correct number of samples per second), checking the correct functionality of the buffer bypass, and evaluating the correct reading values. A RIGOL DS4014 digital oscilloscope with high input impedance has been connected in parallel to the input pins of the ADCs to measure the incident signals and their amplitudes. These values have been compared with the values obtained on the PC side after processing and transmission to the PC. Each channel has been tested separately and together. The simultaneous application to both channels of an identical signal must, indeed, (see euation 3.9) give the same identical input signal as a result. The tested configuration can be seen in figure

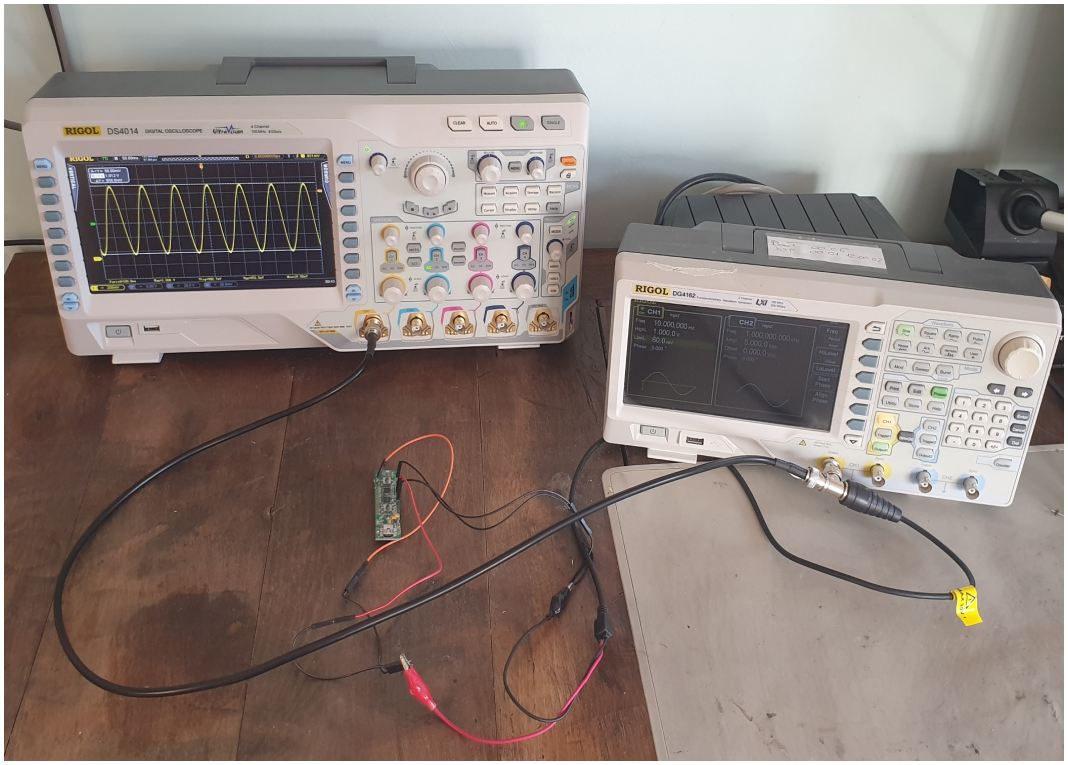


Figure 3.27: The configuration to test the program

As already explained in the ADC section, some problems have occurred with the secondary channel because the input-output pins named P0.0 were not configured properly. This requirement is not well-detailed in the ADuC7061 data sheet and has taken some time to be fixed. The solution was to correctly set the GP0CON MMR register.

CHAPTER 4

Measurement of the tested configuration

In this chapter we will show and discuss the measurements made with the designed acquisition system and evaluate the effectiveness of the presented technique. The results will be compared with those obtained by the direct measurements obtained in [13].

Compared to the latter, the double amplifier board (figure 3.2) has been modified by substituting the Microchip MCP6V02 amplifier with the Texas Instruments TLC272, as illustrated in the previous sections. This was necessary because the Microchip opamp is designed to exploit the chopping technique and this was evident in the spectral analysis of the sampled signal, where a spike appears at the chopping frequency. This fact did not emerge during the measurements done in [13], because the oscilloscope used to sample the data was set with high averaging factor (256). To avoid this spurious presence that would not have been cancelled with the method tested, the opamp has been replaced with another one that does not use this alternating technique.

Another modification with respect to the measurements done in [13] is that the bias shift test has required a non-zero voltage at the input, unlike in the cited article. This is due to the fact that the ADuC cannot accept negative voltages at the ADC inputs but only voltages within a range of 0V to 1.2V. Thus, to always obtain positive values, a DC bias (600mV) has been applied at the input of the board (value that corresponds after the divider at almost 500mV).

4.1 The configuration used for measurement, connections and the procedure

The configuration used during the laboratory tests is visible in figure 4.1. In particular, two Rigol DP832 DC programmable power supplies are employed. The

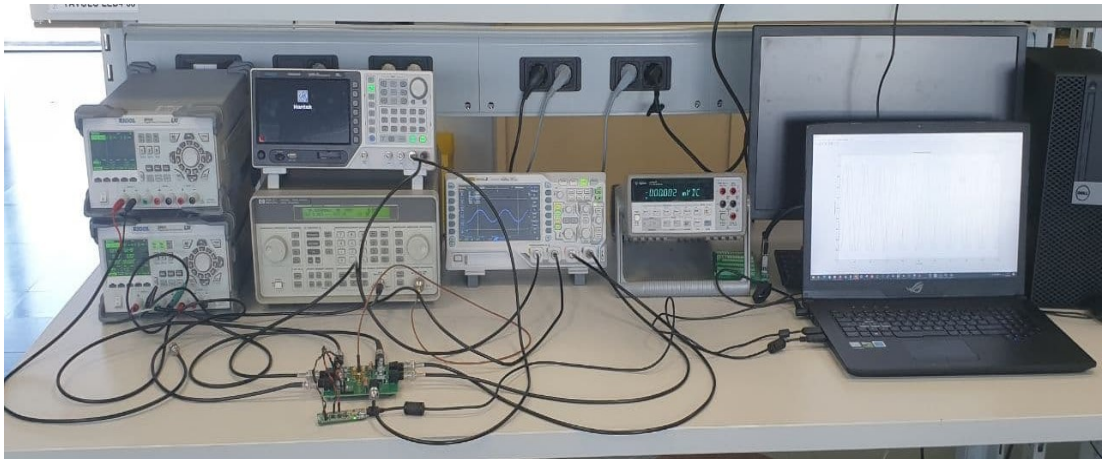


Figure 4.1: The laboratory test bench

first one has been used to provide the correct voltage of $\pm 2.5V$ to the double amplifier board, AB, and the GND reference. The second has been used to provide a fixed DC bias of 0.6V at the input of the AB during the offset measurements, as previously discussed. Their outputs are connected to the AB through banana-to-BNC cables.

An HP8648C 100kHz-3200MHz signal generator is adopted to inject the required RF at the AB EMI input (SMA connector). This device also permits internal modulation of the output signal with a 400Hz or a 1kHz sine wave (as will be seen, some distortions are present). This internal amplitude modulation is adopted for the sinusoidal tests. The high frequency output channel of the HP generator is connected to the double amplifier board through an SMA-to-SMA cable.

A Hantek HDG2032 low frequency waveform generator is used to generate the low frequency signal that is amplified by the opamps and also to generate, on the second channel, the square and exponential waveforms that modulate the RF signal. Therefore, CH2 has been connected with a BNC-BNC cable to the input of the HP RF generator. A similar connection has been used from CH1 to the low frequency input of the double amplifier board. The instrument manual [41] declares a -60dBc harmonic distortion that is well-respected, see figure 4.10 where the signal-to-noise-and-distortion ratio, SNDR, is 72.36dB (see below for the definition, remembering that $SNDR = \frac{1}{THD+N}$). No effects were due to the amplifiers, although information about the distortion is only present in the OPA2277 data sheet where they declare a total harmonic distortion, THD, of 0.002% that corresponds to -114dB. Some distortion has been noticed during the sinusoidal modulation test, where the harmonics of the modulation signal are well

visible.

The RIGOL DS1054 oscilloscope has been employed to calibrate the output of the AB in such a way as to obtain a correct input voltage for the ADuC and to check if every RF signal injected would determine an output signal within the input range of the ADCs. At the same time, the oscilloscope has been used to show the distorted and clean output signals of both the OPA2277 and the TLC272. CH1 and CH3 show, respectively, the distorted and the undistorted output signals of OPA2277, while CH2 and CH4 show the TLC272 corrupted signal and the clean amplified output signal. The connections between the AB and the oscilloscope were made with BNC-BNC cables.

The DS1054 has been, moreover, used to evaluate the correct α value of the amplifiers by saving the sampled channels, due to the impossibility of reading more than two signals at the same time from the ADuC, as will be discussed later.

An HP Agilent 34401A desk multimeter has been used to calibrate the double amplifier board potentiometers and to check if the system would correctly reveal the output bias that has been set. In addition, it has been utilized to compare the bias reading obtained by the ADuC with the direct measured voltages.

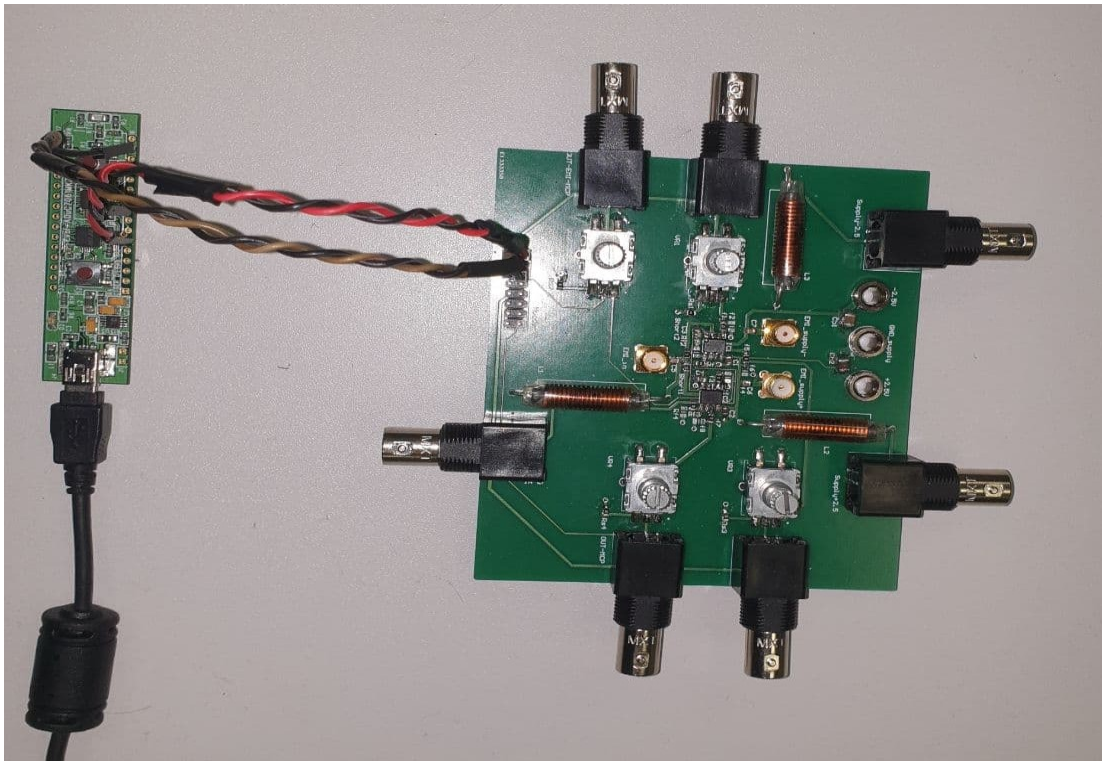


Figure 4.2: The double amplifier board and the ADuC connections

In figure 4.2, the double amplifier board and the ADuC7061 board are visible. The outputs from the first have been connected to the ADCs through twisted DUPONT-terminated wires to avoid coupling with external disturbances. The length of the cables used was approximately 10cm. This value was determined by the pre-assembled cables and the lack of available crimping equipment to reduce their lenght. The analog ground (already tied together with the digital ground on the ADuC board) is connected with the common reference pin (ADC5) on the amplifier board where the inner pin strip is at GND. In figure 4.3, instead, the overall connection schematics are visible.

The EVAL-ADuC7061MKX board is connected to the PC with USB cable. All the following measurements, has been conducted for a period of 1s to reduce possible windowing effects to the 10Hz LF input signal and for the 100Hz and 400Hz modulating signals.

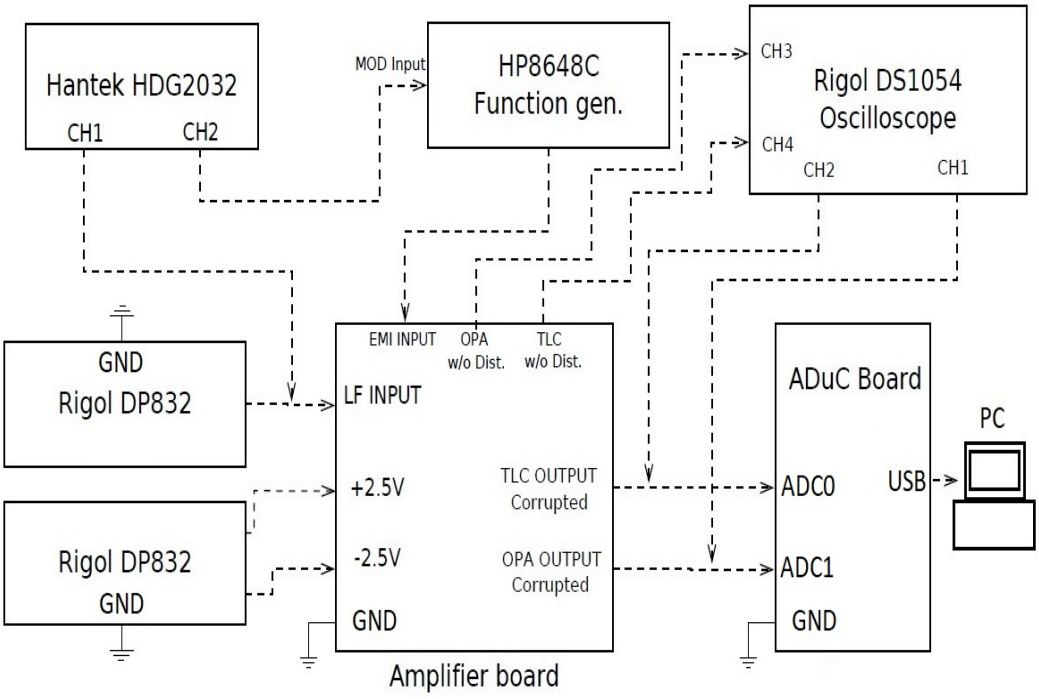


Figure 4.3: The connection schematics

In figure 4.4, the flow chart can be seen that represents the procedures followed and required for the calibrations of the bias and voltage divider, as well as determining the α value. In the next sections, this will be described in detail.

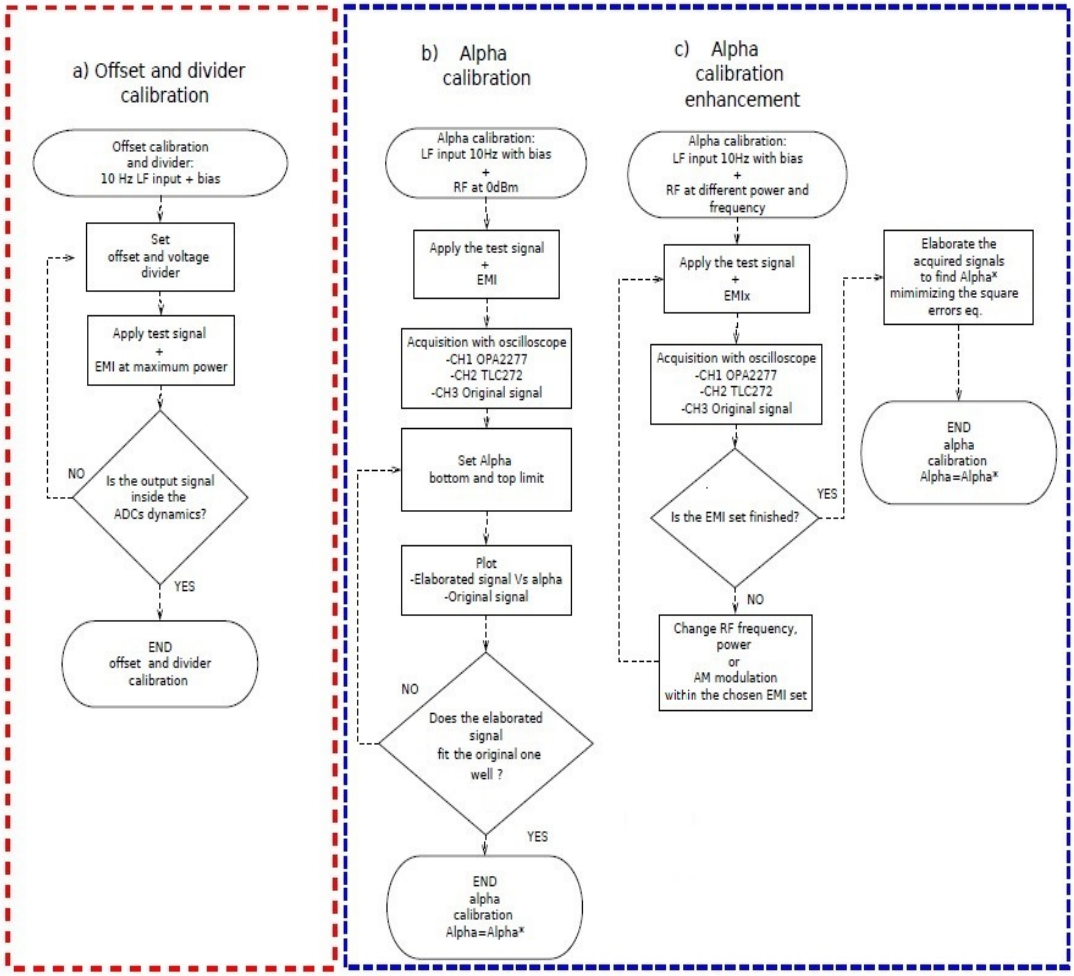


Figure 4.4: The calibration flow chart. a) the offset and voltage divider calibration procedure. b) the α calibration in an experimental way. c)the α calibration with multiple measurements

In figure 4.5, one can, instead, observe the procedure adopted during the measurement of the distorted signals and the data processed and obtained from the ADuC.

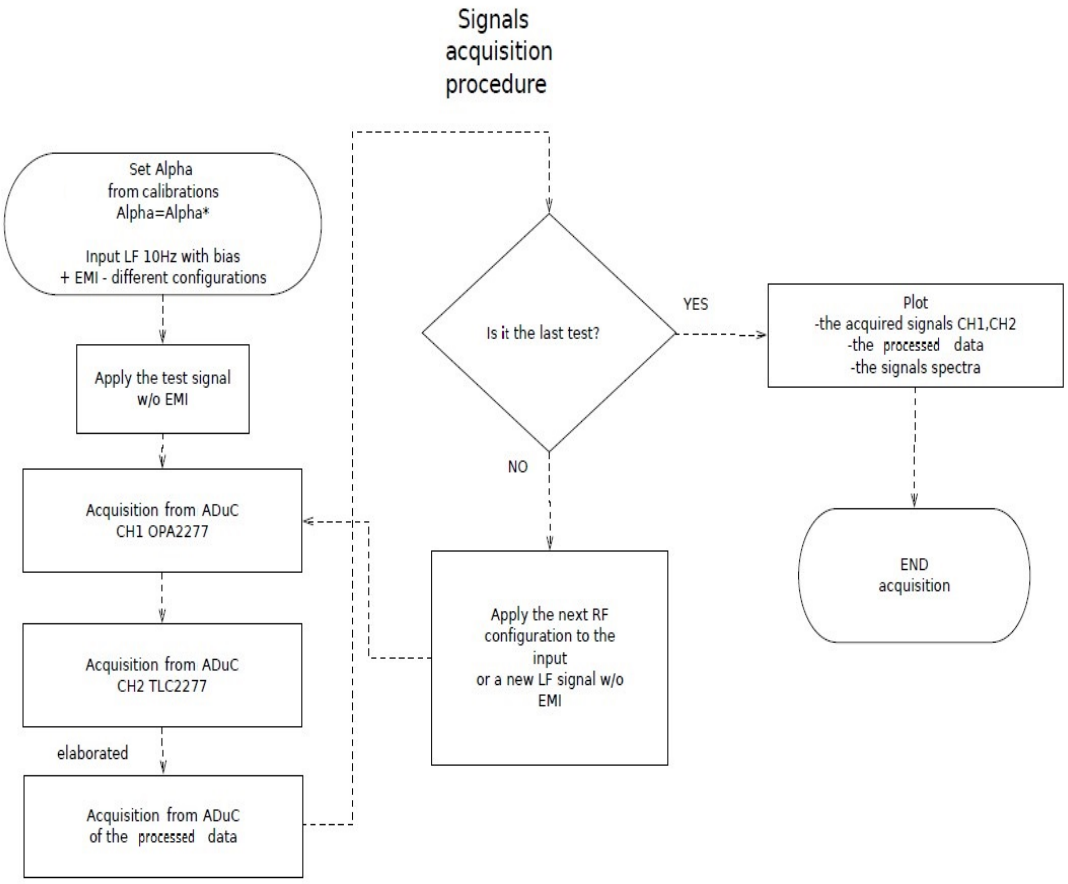


Figure 4.5: The acquisition procedure flow chart

In the following sections, the ADuC program is the modified version. As previously described, it permits to sent separately to the PC the OPA2277 channel, the TLC272 and the data obtained by the computed technique (for reference, see the program chapter and 6.3.2 in the appendix). This, in turn, gives outputs that are not time-related, so, a phase shift between the channels is present. This has not been an issue for the evaluation of the results obtained, since the input signal and the EMI do not vary between one acquisition and the next. The only drawback was that the signals were out of phase, and this difference required adjusting for a coherent plot of the signals in time.

4.2 Parameters definition

During the next section, some definitions will be used to evaluate the effectiveness of the processed data. In particular, the spurious free dynamic range,

SFDR, will be used that, in dB, expresses the difference between the power of the fundamental frequency and the highest spurious harmonics, in addition to the signal-to-noise-and-distortion ratio, SNDR or SINAD, that, in dB, provides the difference between the power of the signal and the power of the harmonics with the noise, DC excluded. For this parameter, the band must, generally, be expressed but, in case of FFT, this is defined as half of the sampling frequency. The SNDR corresponds, in modulus, to the total harmonic distortion plus noise, THD+N, definition ($SNDR = \frac{1}{THD+N}$) since it is its reciprocal.

The above introduced parameters are defined in equation 4.1,

$$\begin{aligned} SNDR = SINAD = \frac{1}{THD+N} = 10log_{10}\frac{P_{sin}}{P_{tot}-P_{sin}} \\ SFDR = 10log_{10}\frac{P_{sin}}{P_{max,spur}} \end{aligned} \tag{4.1}$$

where $P_{sin} = V_{sin}^2/2$ with V_{sin} the peak value of the fundamental signal, $P_{tot} = \overline{v}^2$ is the total power of the signal, noise included, and $P_{max,spur}$ is the power of the highest harmonic.

Another parameter that permits measuring the quality of the amplifier with respect to the EMI-induced offset is the EMI rejection ratio or EMIRR, defined in equation 3.6 or, equivalently, in 4.2

$$EMIRR = 20Log_{10}\left(\frac{V_{RF_Peak}}{\Delta V_{OS}}\right) + 20Log_{10}\left(\frac{V_{RF_Peak}}{100mV}\right) \tag{4.2}$$

with $V_{RF_{Peak}}$ the peak amplitude of the applied RF voltage and ΔV_{OS} the induced DC offset voltage shift. The second term of the equation refers the EMIRR to an input signal of 100mV (for 0dB EMIRR, an offset value of 1 is obtained at the output).

4.3 Offset calibration

Before starting the measurements, the potentiometers of the AB have been set to have compatible input voltages coming from the amplifier outputs. The LF injected into the opamps requires, indeed, an offset to avoid the output voltages going below 0V. At the same time, there must be a guarantee that the voltages do not reach a value greater than 1.2V. These conditions must be respected for all the RF amplitudes and injecting conditions and, also, in case of modulated signals.

As will be clarified in the next section, the OPA2277 and the TLC272 react to the input RF in different ways. The OPA has a positive behaviour with respect to the RF at input, whereas the TLC has a negative reaction. Moreover, the OPA has a more evident response to the applied RF than the TLC (see figure 4.6). For

these reasons, the LF input offset cannot be exactly at the center of the ADCs input dynamic (0-1.2V) but must be maintained slightly lower. For high injected RF power, this avoids the output of the amplifiers going beyond the maximum allowed input of the ADCs or below the minimum.

Following these considerations, the DC voltage of the input 10Hz LF signal coming from the Hantek HDG2032 is set to a value of 600mV. In turn, the voltage dividers present at the output of the OPA and TLC are set to have an average voltage of 500mV. This configuration has made the obtained voltages compatible with the ADCs input dynamic for almost all the measurement conditions (some limitations that have required an RF power reduction, have been encountered in case of an injected sine wave modulated RF of 9dBm that have exceeded the maximum allowed voltage, relative to the OPA output).

Altogether, the opamp output voltages require translation to avoid going below 0V and attenuation to avoid going beyond 1.2V. The voltage divider ratio at the opamp output has been set to about 0.83. This for the low frequency input signal of 10Hz correspond to a bias value of 500mV (as will be seen, the signal coming out of the generator has a peak to peak voltage of 500mV and a bias voltage of 600mV).

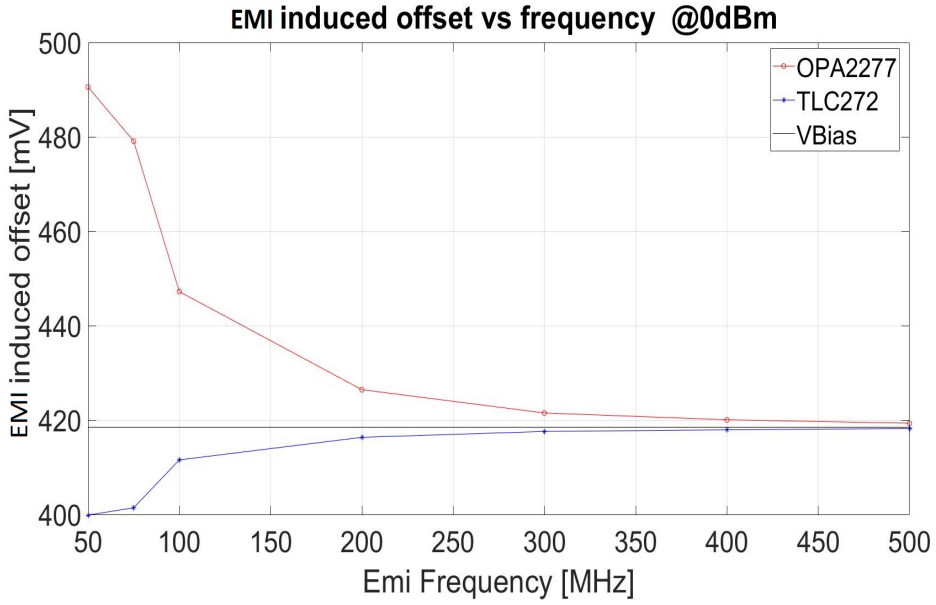


Figure 4.6: The different responses of the opamps

4.4 Alpha estimation

Before starting the acquisition of the signals, the required value of alpha has been evaluated. The alpha obtained in [13] could no longer be considered correct, since one amplifier has been changed.

As previously discussed (see chapter 3, theoretical base section) two possibilities are available. The first is to evaluate $\alpha = \alpha^*$ through direct measurement, exploiting equation 3.4, reported here for ease, eq. 4.3.

$$\alpha^* = -\frac{n_{EMI,2,k}}{n_{EMI,1,k} - n_{EMI,2,k}} \tag{4.3}$$

In this case, the samples of the first and second channels are acquired, in addition to the original unperturbed channels. The errors are obtained by subtraction, eq. 4.4. This method can be utilized for any k sample and for any RF injected (since alpha is almost independent of these conditions, see alpha redefinition section).

$$\begin{aligned} n_{EMI,1,k} &= y_{1,k} - x_{1,k} \\ n_{EMI,2,k} &= y_{2,k} - x_{2,k} \end{aligned} \tag{4.4}$$

Here, however, a slightly different approach has been used. At the input of the double amplifier board, 10Hz LF signal ($V_{pp} = 500mV$ and $V_{off} = 600mV$) and an RF signal of 100MHz, 0dBm (that corresponds to an open circuit voltage of 632mV) have been applied. All of the output channels (OPA and TLC distorted and clean) have been sampled through the RIGOL DS1054 oscilloscope. The estimation of α has been done on the PC side where both the results obtained by the technique ($y_k(\alpha) = \alpha y_{1,k} + (1-\alpha)y_{2,k}$) for a variable α and the clean one have been plotted (see figure 4.4 for the procedure flow chart and the appendix chapter for the Matlab script). It is to be noticed that this procedure is not feasible through the ADuC since only one channel at time can be sent to the PC (a firmware modification could have been possible but, in any case, would not have permitted simultaneous sampling of all four required channels at the same time). Since the original signal has not been available due to the fact that both the OPA and TLC outputs were attenuated by the voltage divider, the voltage of the OPA has been considered as a reference due to its high precision and its reduced input offset. In this case, however, the TLC and OPA differed from one another by almost $300\mu V$ (as will be seen, this value was comparable with the induced offset and, for this reason, the choice of the reference voltage influences the results for low power injection or for high frequency).

On the PC side, the acquired signal and the reference signal have been plotted, refining the α value to best fit the two graphs. The value obtained from this

procedure was $\alpha = 0.3846154$ (see figure 4.7 for the best fit). This easily estimated value, however, turned out to not be very accurate, especially for lower injected power. For this reason, a better evaluation of alpha has been performed.

Following what was explained in the technique chapter, this new, more precise evaluation has been done through a bigger set of measurements (A set) that includes 15 different power, frequency, and RF modulation conditions (here, the procedure has been done by varying the RF from 10MHz to 400MHz for various power injections (-15dBm to 9dBm) and for various types of RF modulating signal). The estimation of alpha has, therefore, been obtained by minimizing the sum of the square errors of expression 2.10, (eq. 4.5):

$$\begin{aligned} \alpha^* &= \arg\min_{\alpha} \sum_{k \in A} [\alpha n_{EMI,1,k} + (1-\alpha) n_{EMI,2,k}] = \\ &= \frac{\sum_{k \in A} n_{EMI,2,k}(n_{EMI,2,k} - n_{EMI,1,k})}{\sum_{k \in A} (n_{EMI,1,k} + n_{EMI,2,k})^2} \end{aligned} \tag{4.5}$$

After measurement, this value has also turned out to not be the best choice, especially in low power conditions or where a reduced induced offset is present. The modifications have been investigated (see the Alpha redefinition section)

All signals have been acquired with the Hantek oscilloscope and then processed on the PC side (see figure 4.4 for the procedure flow chart and the appendix chapter for the script used).

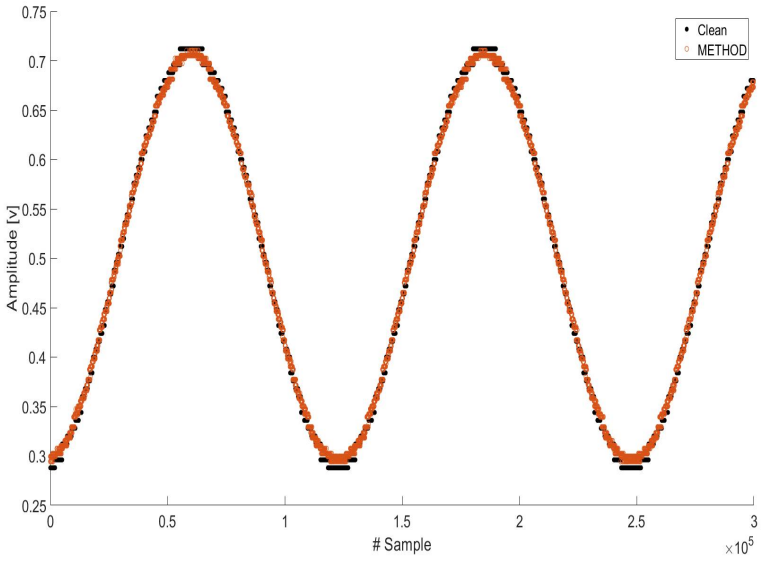
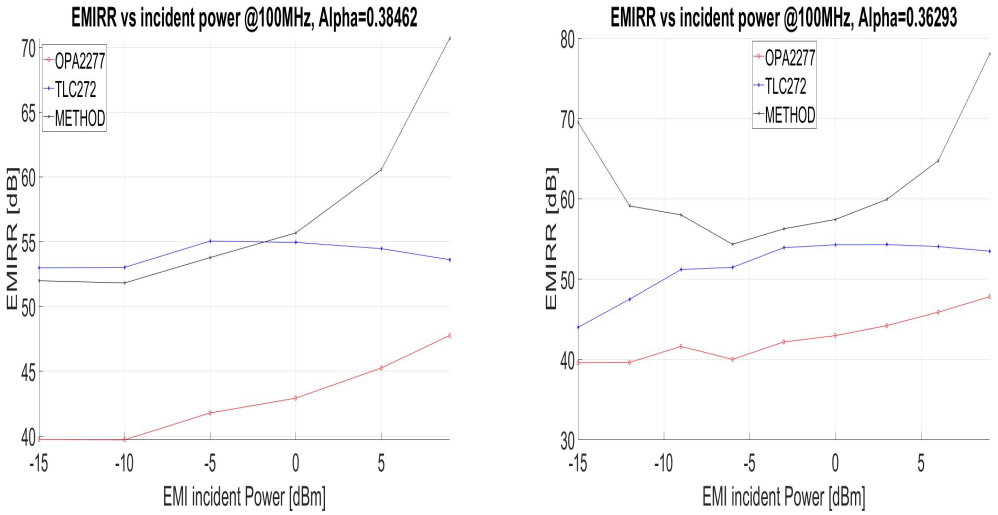


Figure 4.7: Alpha best fit



Alpha = 0.3846154 Alpha = 0.3629318

Figure 4.8: Alpha effect on EMIRR. a)Alpha = 0.3846154, b)Alpha = 0.3629318

For each processed signal, various samples have been taken for the computation (for example, one sample out of every 500 acquired points). The result obtained from this computation was $\alpha^* = 0.3629318$, a value that has been used for all the measurement procedures. Of note is that this value is the best solution only for the A set of RF inputs. Within the same set, it can, however, give better results in some cases and worse in others. See figure 4.8 to highlight the influence of the choice of the correct alpha, where EMIRR is compared, defined in eq. 3.6 or eq. 4.2, for the two alpha values discussed above.

It must be specified that the technique still works for the first alpha computed ($\alpha = 0.3846154$), yet better results can be obtained with the last alpha determined ($\alpha^* = 0.3629318$). To slightly increase the accuracy of this parameter, a bigger set can be evaluated but, in turn, this requires expending more effort in terms of time. Thus, the method to be employed for evaluating alpha also depends on the working conditions of the system.

In the next sections, the measurements are obtained with $\alpha = 0.3629318$.

4.5 System response without EMI

The first test done, after the input offset calibration, has been the evaluation of the signals without the EMI to verify the declared specifications of the generator and the amplifier. An LF signal of 10Hz 500mVpp and with 600mV of DC bias, without RF signal, has been applied at the input of the double amplifier board.

Then, the acquisition of the signals has been performed and the computation of the spectra has been done. In figures 4.9 and 4.10, the signal in time and the spectrum for the OPA output signal can be observed, likewise for the TLC opamp in figure 4.11 and 4.12. As can be seen, the generated waveform is clean and with an SFDR of almost 80dB and an SNDR of nearly 84dB (see equation 4.1 for the definitions). As explained, the obtained values are in line with the generator and opamp specifications (-60dBc of harmonic distortion for the Hantek, the critical one, and a total harmonic distortion, THD, of 0.002% for the OPA, no information about TLC distortions is present in the data sheet).

Comparing these results to the same done in [13], here, the signal has a frequency an order of magnitude less than in the article and this justifies the good performance of the generator, which is the main cause of distortions.

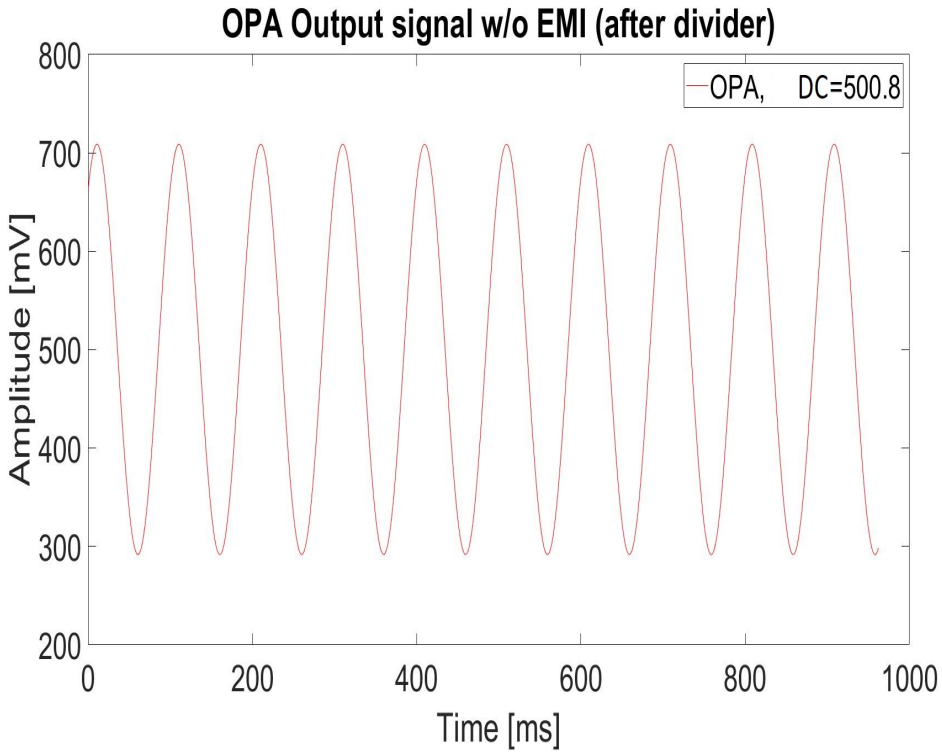


Figure 4.9: OPA output with no RF and 10Hz sin input

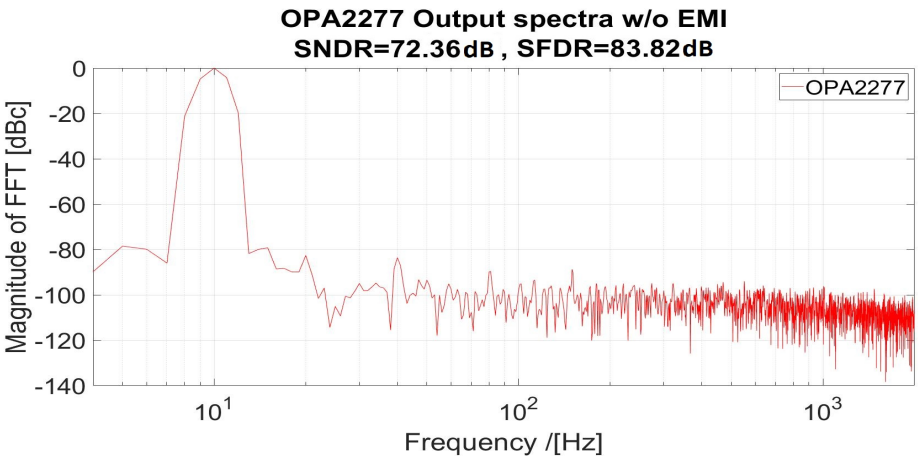


Figure 4.10: OPA output spectrum with no RF and 10Hz sin input

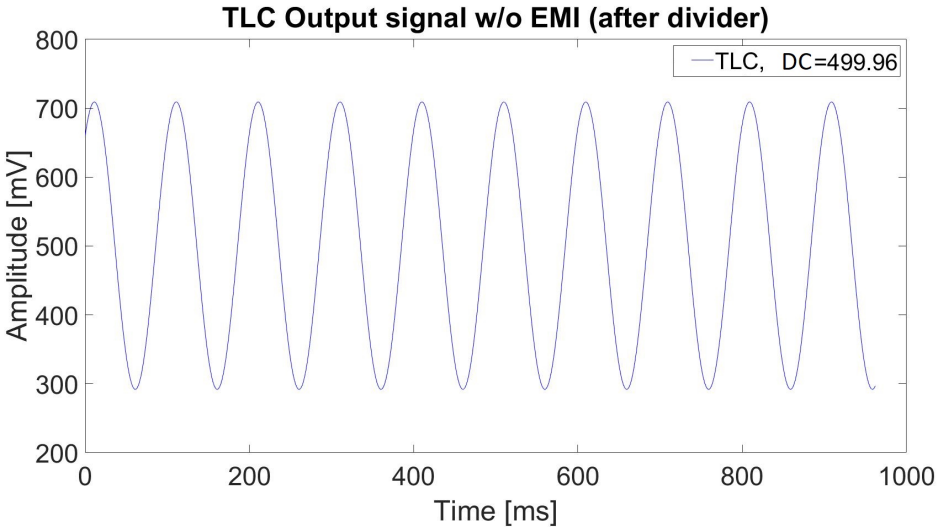


Figure 4.11: TLC output with no RF and 10Hz sin input

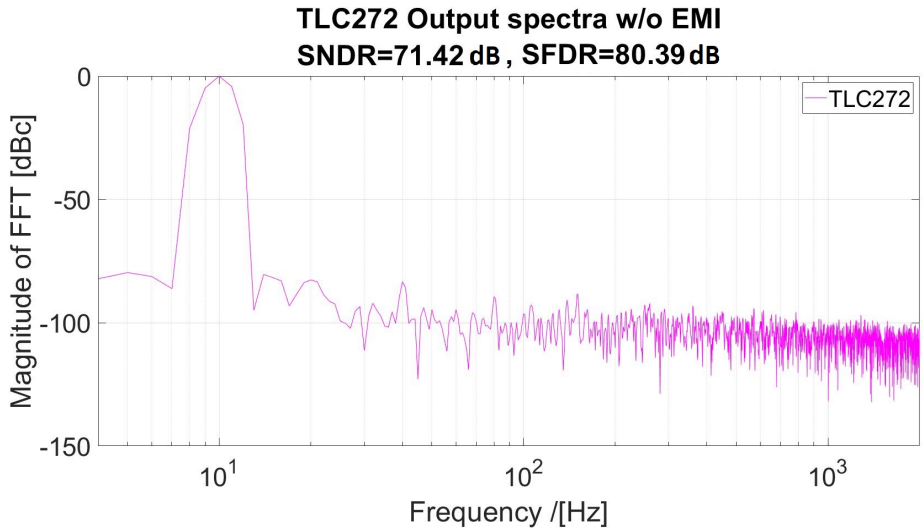


Figure 4.12: TLC output spectrum with no RF and 10Hz sin input

4.6 Induced offset measurements

In this section, the induced offset with respect to the injected EMI at the input has been tested. For this experiment, a DC voltage of 0.6V has been applied to the input of the amplifier board that, at the output, corresponds to 0.5V due to the voltage divider. As discussed, this bias has been required to respect the input range of the ADCs, differing from what was done in [13]. Two tests have been conducted. The first was a characterization of the offset versus power for a 100MHz EMI-injected CW. In this case, the power adopted was from -15dBm (that corresponds to an open circuit peak voltage of 112mV, $V_{EMI,pk} = 2\sqrt{2 * P_{inc} * 50\Omega}$) to 9dBm (1.78mV). The second test was conducted with an injected CW EMI at fixed power of 0dBm (that corresponds to a peak voltage of 632mV) and varying the frequency from 50MHz to 500MHz.

The output of the amplifiers has been sampled for a period of 1s, a value chosen to avoid windowing effects on the 10Hz signal, and compared with the computation returned from the ADuC that performs the techinque. For both the tests, the EMIRR has been calculated and the measured offset shift plotted, ΔV_{off}. As previously discussed, the reference voltage employed without EMI was the OPA voltage.

It is worth noting that, for low injected power, the EMIRR is significantly conditioned by the reference voltage used (OPA or TLC) and by the chosen Alpha. Indeed, for power of -9dBm and below, the induced error is of the order of a hundred μV, which is comparable with the quantification error and measurement uncertainty and, also, with the output difference of the two amplifiers under normal conditions.

In figures 4.13 and 4.14, the results of the test versus power can be seen. In this case, it can be observed that the induced offset shift for the method is limited to about 9mV with respect to the DC bias of 501.1mV for all the power values. Instead, for the amplifiers, it can reach considerable values i.e., 129.2mV (630.3mV-501.1mV) for the OPA and 67.3mV for the TLC. At -3dBm, the TLC induced offset is almost 2.7mV.

The EMIRR of the chosen method has a maximum of 78dB for a power of 9dBm and a value of 56.27dB at -3dBm. The value at -3dBm is only 2.34 dB above the TLC. This fact is due to the good behaviour, in terms of EMI, of the TLC which shows a good EMIRR value for almost the entire power range and for a slightly imprecise value of alpha in this same range (see below). The increase of the TLC EMIRR around -5dBm, in addition to the reduction for the OPA, gives this parabolic behaviour to the method and a minimum of 54.34dB at -6dBm.

In this case, it can be observed that the chosen Alpha value cannot cancel the EMI effects well. A reduced value can, in this situation, give better results.

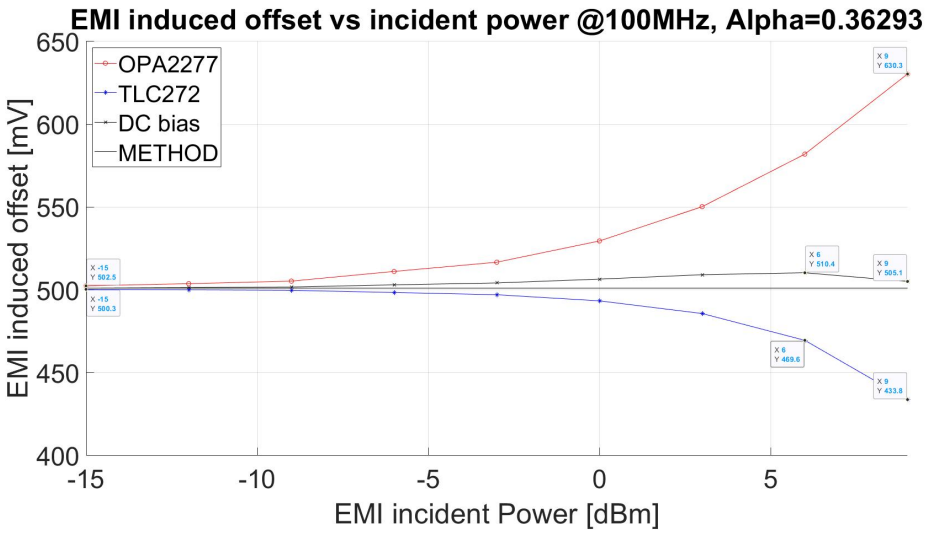


Figure 4.13: Induced offset for an injected 100MHz CW EMI versus power

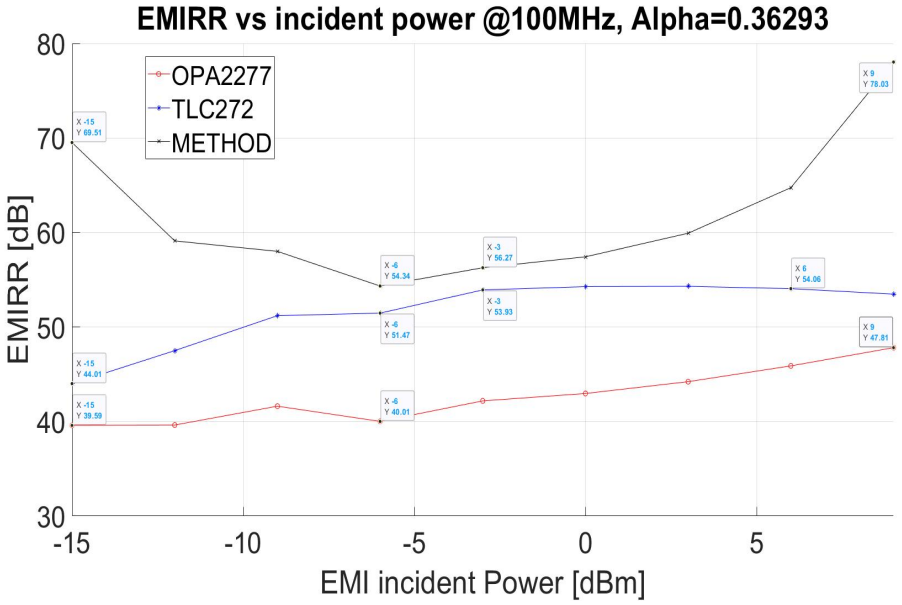


Figure 4.14: EMIRR for an injected 100MHz CW EMI versus power

Here, in fact, the behaviour of the method offset replicates the OPA offset (see the next subsection for an alpha redefinition). The same test with an alpha value of 0.3573428 (obtained reducing the set of measurements used to approximate this parameter) has given good results through -5dBm but, at the same time, has showed a reduction in the effectiveness under other EMI conditions. Altogether, good results were obtained for the method, especially for relatively high input injected powers. The technique, with this value of Alpha, has shown less effectiveness at low power.

The comparison with the previous study results [13] for the same type of measurements, shows a reduced efficacy for medium power injection. However, here, a different, less EMI-susceptible amplifier has been adopted that exhibits a better EMIRR (of about 10dB) with respect to the MCP opamp used and can influence the behaviour to reduce induced error. The method is, however, still valid because the results are always above the response of each single amplifier. An improvement of the results could be obtained by better refining the alpha value with a bigger set of measurements concentrated on the frequencies and powers that have shown reduced performance (see the dedicated subsection).

In figures 4.15 and 4.16 the results are shown for the frequency measurements.

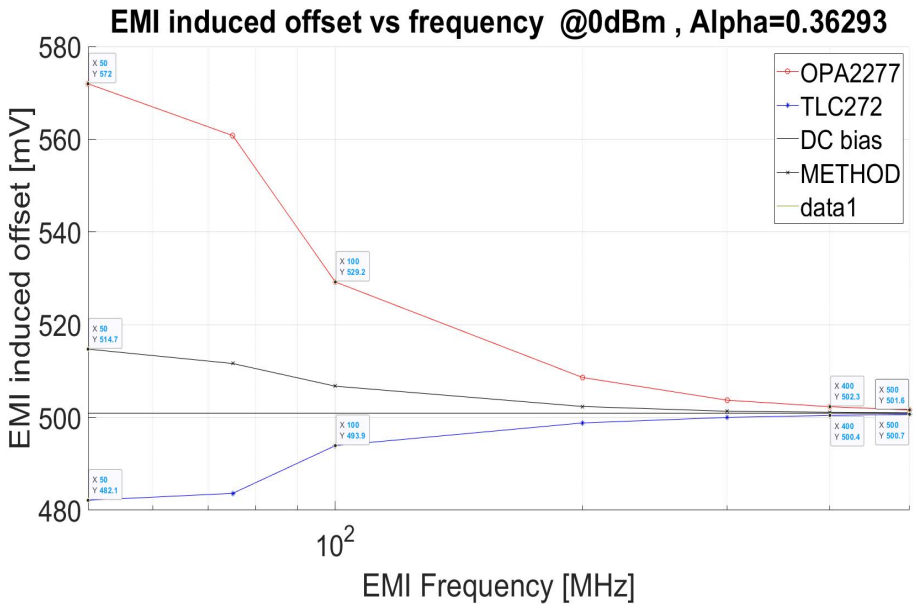


Figure 4.15: Induced offset for an injected CW EMI at 0dBm versus frequency

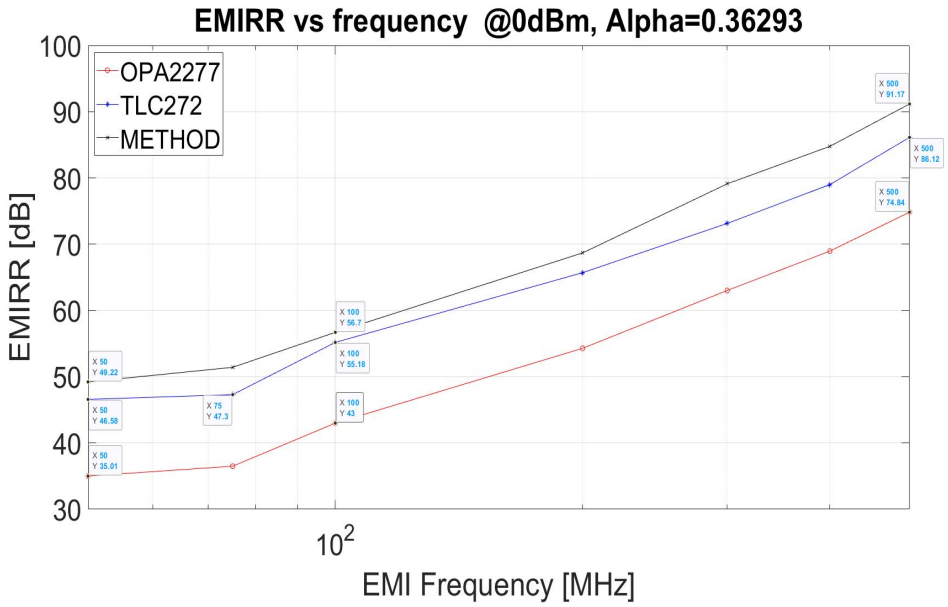


Figure 4.16: EMIRR for an injected CW EMI at 0dBm versus frequency

Here, the induced offsets at 50MHz are almost 15mV for the method, 19mV for the TLC, and 71mV for the OPA. For low frequency, the method is still effective but less with respect to the previous study analysis where a $\Delta V_{off} = 3.2mV$ was obtained. Also, in this case, the behaviour of the method copies the OPA amplifier, and this suggests that a new calibration of the alpha could be required. For higher frequencies both the amplifiers reduce the sensitivity to EMI and, so, the induced offset almost completely disappears at 500MHz.

The EMIRR at 100MHz are 56.7dB for the method, 55.18dB for the TLC, and 43dB for the OPA. There was only an improvement of approximately 1.5dB for the method at this frequency. For the other frequencies, the improvement is nearly 6dB. This behaviour can, also in this case, be explained by an imprecise value of Alpha that requires more evaluation at lower frequencies, as observed for the same level of power in the previous figure. The correct Alpha value is, therefore, lower than the one used in these measurements.

4.6.1 Alpha redefinition

Taking into account what is observed in the previous figure, a new evaluation of alpha has been performed. The best value of the parameter can be obtained using a bigger set of measurements and excluding the cases where the behaviour of the amplifiers is particularly subject to sudden changes. In figures 4.17 and 4.18 the effects of an alpha variation can be observed with respect to the power injected

on the offset and the EMIRR. Here, good results over all the measured conditions have been obtained with a value of alpha equal to $\alpha^* = 0.27$. This value, indeed, gives an offset of almost 4mV for an injected power of -6dBm. Of note is the fact that, in this case, the trend of the method does not copy that of the OPA but, instead, copies that of the TLC for high injected power. This value, hence, could be considered a good choice to use.

From the EMIRR figure, instead, it can be observed that a value of 0.27 can give uniform behaviour throughout the power sweep, yet a value of 0.32 gives the highest peak of immunity with an EMIRR of 85.21dB at 0dBm.

It should be noted that the two above-mentioned alpha parameters have a reduced EMIRR for low and high injected power. Although for low power the problem is irrelevant, for higher injected power the reduction can affect the responses of the method.

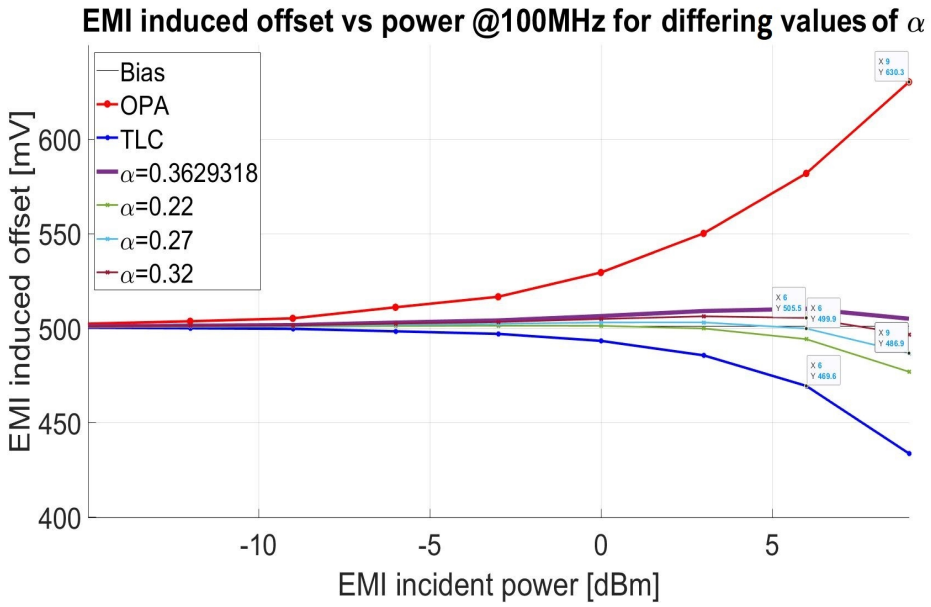


Figure 4.17: Induced offset for an injected 100MHz CW EMI versus power and vs α

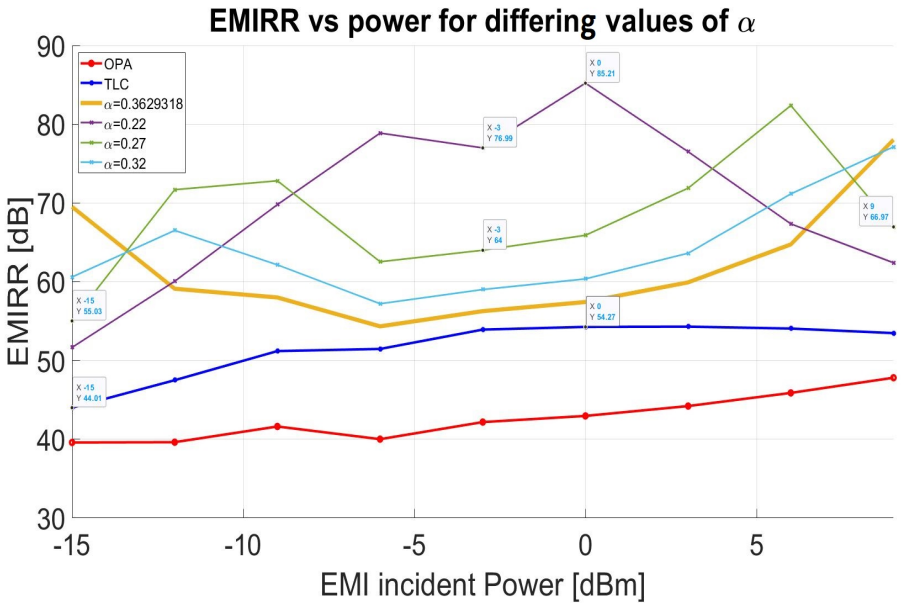


Figure 4.18: EMIRR for an injected 100MHz CW EMI versus power and vs α

In figures 4.19 and 4.20, a similar analysis is seen that evaluates frequency variation and constant 0dBm power. Also in this case, a better result is obtained for a lower alpha. In particular, a good choice could be 0.215. This value, indeed, results in low induced offset for all the frequencies involved, keeping the shift for the method below 1mV. Its trend seems to follow neither the OPA, nor the TLC (except in a limited way). Really good results have been obtained in terms of EMIRR where a relevant increase has been evidenced with respect to $\alpha = 0.3629318$, having a peak of 92.39dB at 50MHz that corresponds to an increase with respect to the TLC of almost 46dB.

As explained the choice of alpha in this technique is a critical step to carefully evaluate and a value valid for one condition may not be so for another. Next, the Alpha value computed with the minimization method has been maintained. The above evaluation of the Alpha value was possible after measurement. With simple modifications, this redefinition can be applied to the designed system.

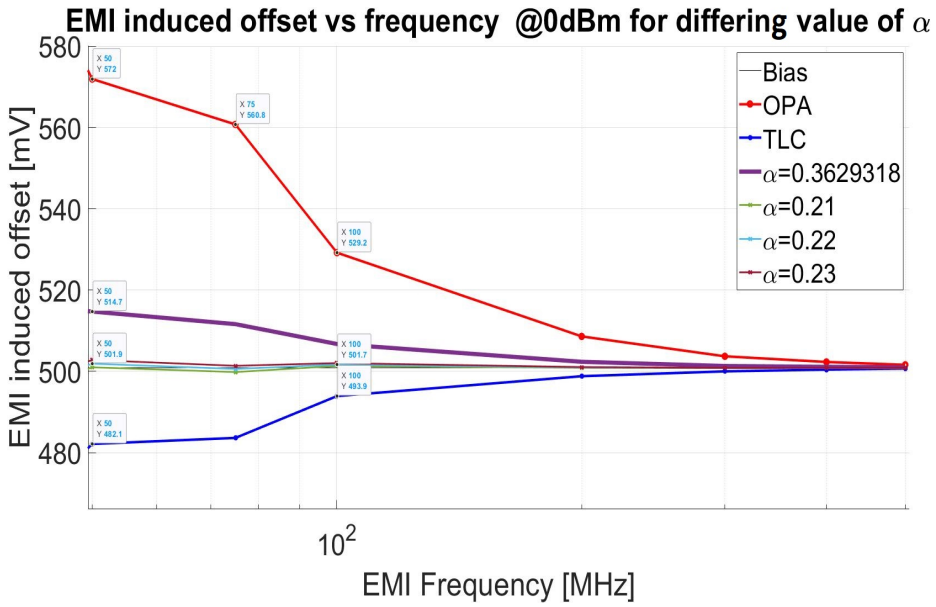


Figure 4.19: Induced offset for an injected CW EMI at 0dBm versus frequency and vs α

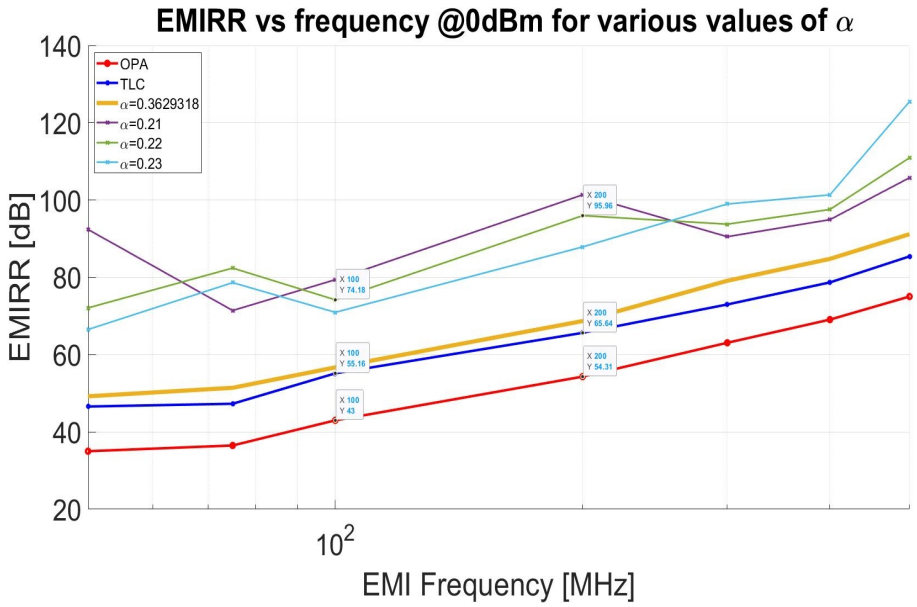


Figure 4.20: EMIRR for an injected CW EMI at 0dBm versus frequency and vs α

4.7 CW EMI measurement

In this section, the result of the measurements under different EMI conditions with modulated RF signal will be illustrated. Different modulations have been applied in order to prove the effectiveness of the method in different situations. For the square wave modulation, a comparison with a different alpha is conducted to show how it can positively or negatively influence the results.

The modulations employed are the same used in [13]. Having a limit imposed from the sampling frequency and the LPF, as explained above, the frequencies of the modulating signal have been scaled down to avoid possible attenuation. Indeed, for all the measurements, the LPF has been always present.

In equation 4.6, the formula is summarized that highlights the amplitude modulation applied to the input of the amplifier board.

$$V_{EMI,AM} = V_{EMI,pk}\left(1 + m\frac{v_{mod}(t)}{V_{mod,pk}}\right) sin(w_0 t) \tag{4.6}$$

with $V_{EMI,pk} = 2\sqrt{2 \cdot P_{inc} \cdot 50\Omega}$ the incident EMI voltage considered at open circuit (the reflection coefficient is almost 1), $0 < m \leq 1$ the modulation index and $V_{mod,pk}$ the peak amplitude of the modulating signal. In our case, the modulating signals used have been a sinusoidal wave, a square wave, and a decreasing exponential wave expressed as (eq. 4.7) with $\Pi(x)$ the unit pulse function and T the period of the applied wave:

$$V_{mod,exp} = V_{exp,pk}\sum_{-\infty}^{+\infty} \exp -\frac{t - hT}{\tau} \prod (1 - \frac{t}{hT}) \tag{4.7}$$

4.7.1 RF AM square modulated measurements

In this section, a 100MHz RF signal with different injected powers is tested. A plot of the obtained signals has been done, and a spectral analysis is obtained from the sampled signals. The spectra of the signals have been plotted by applying a Blackman window to reduce the frequency leakage due to the limited sampling period (other windows were applied with reduced effects compared with the chosen one). All the spectra plots are referred to the 10Hz carrier, so the y axis has to be considered as dBc.

It is important to remember that the figure plotted versus time has been previously manipulated to have the same phase for a better comparison. As explained, the acquisition from the ADuC has been done in three different successive steps.

The first step is for the OPA acquisitions, the second for the TLC, and the third returns the computed data with the applied technique.

The first signal applied has been a 100MHz RF at 0dB ($V_{EMI,pk} = 632mV$) modulated with a square wave of 100Hz with peak voltage of 1V unipolar (0-1V) with a duty cycle of 50%. The modulation index in this case has been m=100% (see eq. 4.6). In figure 4.21, all the signals that comes out out of the amplifiers, can be seen together. The output of the OPA and TLC are clearly modified by the demodulation of the injected RF. The signal related to the method is, instead, less perturbed by the distortion. Also worth noting is the presence of the offset in the amplifier output. In figure 4.22 for the OPA, 4.23 for the TLC, and 4.24 for the method, the details of each signal and its spectrum are visible.

The method permits, in this case, having a cleaned signal, passing from the SNDR=17.43dB of the OPA to a SNDR=33.9dB for the method. Similar results were found for the first harmonic that is present at almost 34.5dBc below the fundamental. An overall detail of the relative parameters can be seen in figure 4.25 where all the signals are considered together. Here, one can easily see the peak at 400Hz of the square wave that modulates the input and its harmonics. A reduced aliasing effect can be seen at high frequency, because the LPF and, even more, the internal decimator-filter have relevant effects on frequencies beyond the Nyquist frequency.

An important comparison has been made in figures 4.26 and 4.27 where the same test has been conducted for an alpha = 0.27. As can be seen, in time and in frequency, the improvement is evident. From the previous test, the SNDR has increased to 45.08dB. This fact can also be observed with a cleaner signal plotted over time, see below for a comparison with an injected power of 9dBm where the results are quite different.

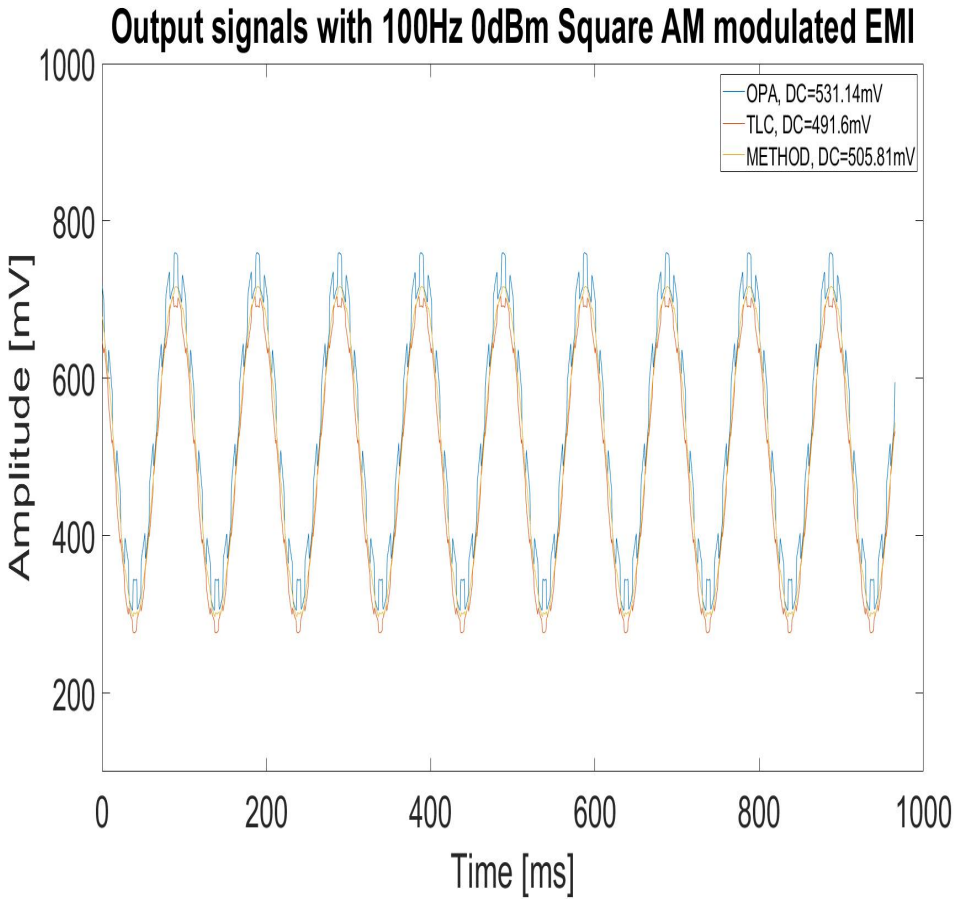


Figure 4.21: Outputs with 0dBm RF 100Hz square wave modulated input

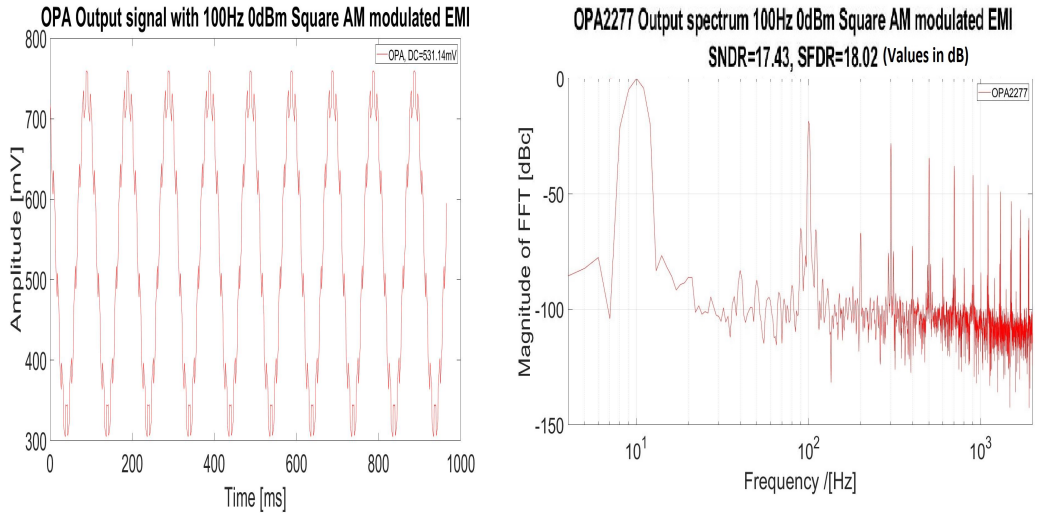


Figure 4.22: OPA output and spectrum with 0dBm RF 100Hz square wave modulated input

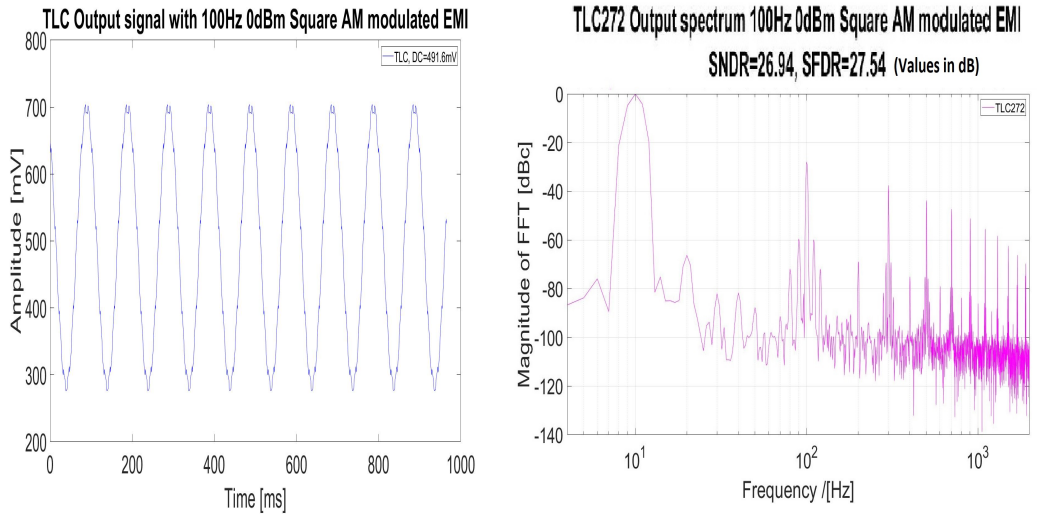


Figure 4.23: TLC output and spectrum with 0dBm RF 100Hz square wave modulated input

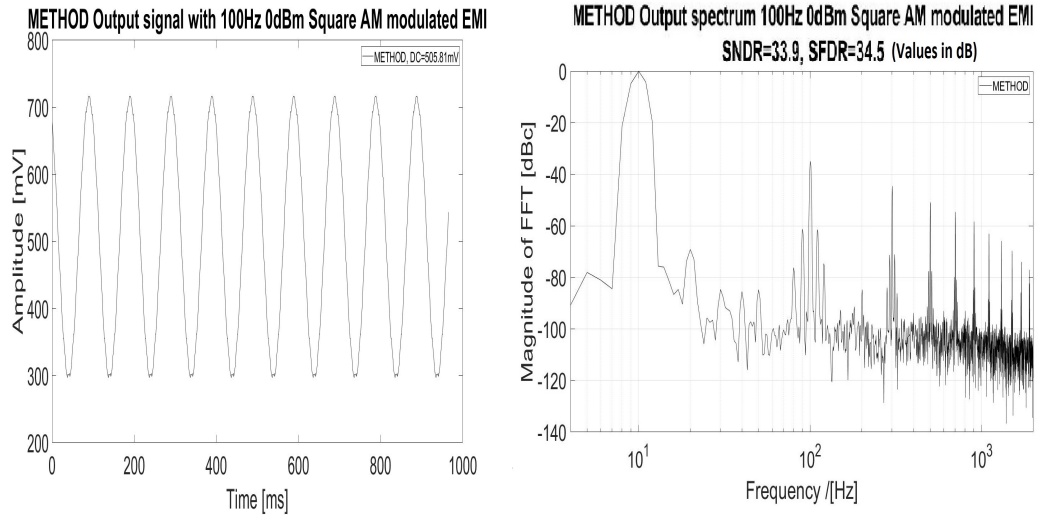


Figure 4.24: METHOD output and spectrum with 0dBm RF 100Hz square wave modulated input

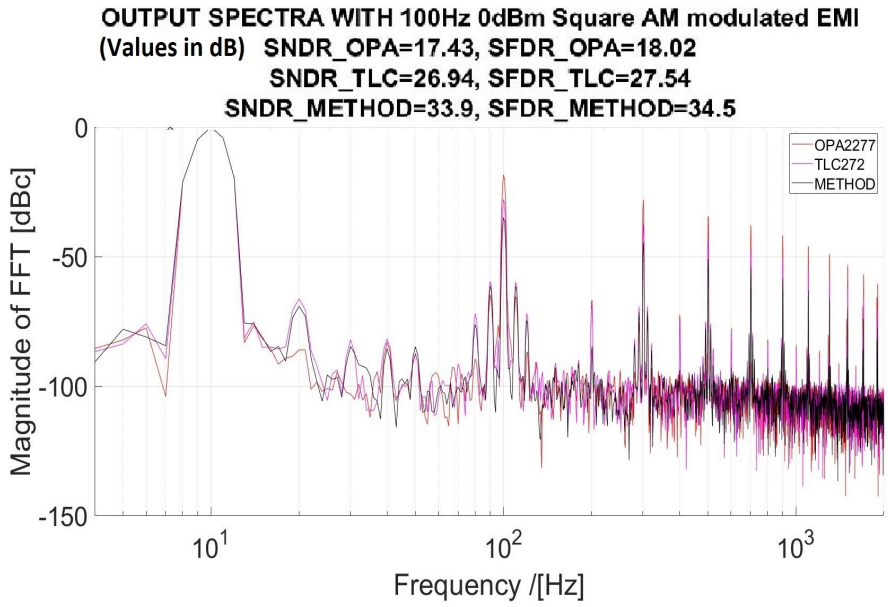


Figure 4.25: Output spectra with 0dBm RF 100Hz square wave modulated input

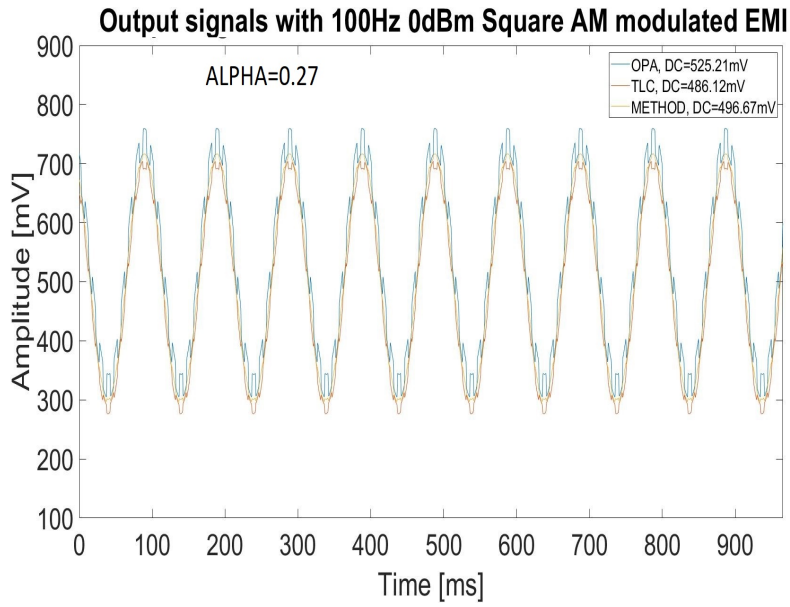


Figure 4.26: Outputs with 0dBm RF 100Hz square wave modulated input for $\alpha = 0.27$

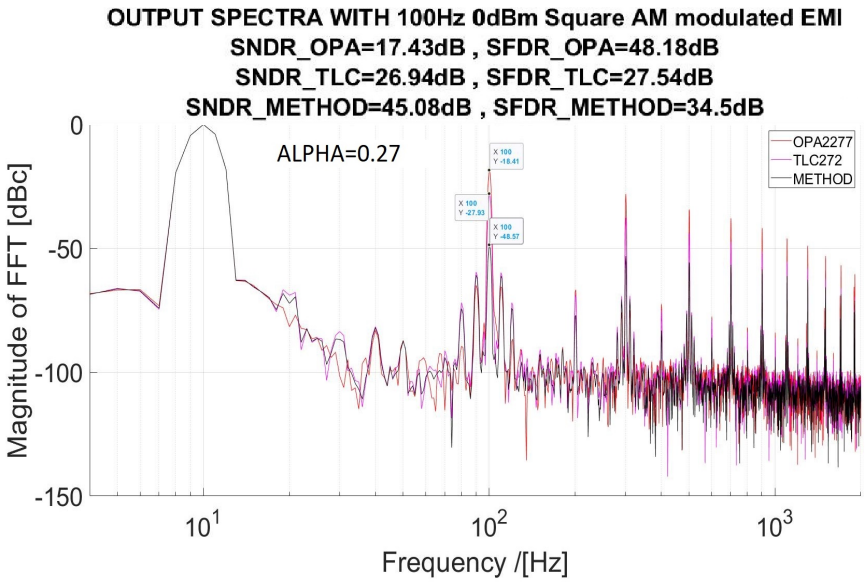


Figure 4.27: Output spectra with 0dBm RF 100Hz square wave modulated input for $\alpha = 0.27$

In figure 4.28, an injected power of 9dBm has been employed, keeping the same square modulated RF at 100MHz as before. In this case, the distortions are more evident. Already from the plot over time, the effectiveness of the method can be clearly seen. A detail of the single signals can, instead, be observed in figures 4.29, 4.30, and 4.31. Here, the SNDR and SFDR has drastically reduced. For the OPA an SNDR=3.88dB and an SFDR=4.48dB is obtained and for the TLC, respectively, 7.49dB and 8.09dB. These values are greatly reduced with respect to the SNDR=28.15dB and SFRD=28.82 of the method. Moreover, to be noticed is that the bias present in case of OPA and TLC is, also, mitigated by the technique.

In figure 4.32, the spectra of all the signals together can be observed. With respect to the test conducted under similar conditions in the previous study [13], here reduced improvements are obtained, an SNDR=28.15 compared to an SNDR=46.28dB and an SFDR=28.82 compared to an SFDR=53.09. At the same time, however, these parameters have also reduced for the other amplifiers: for the OPA the SNDR passes from 3.88dB, of the test conducted here, to the 23.04dB of the previous study. These effects are almost completely attributable to the imperfect tuning of the alpha parameter and working at a lower frequency for the modulating signal. At a higher frequency, indeed, similar has be seen in the offset section, while the effects of distortions are less evident.

As in the previous test,a comparison has been made under the same conditions but changing the alpha. In figures 4.33 and 4.34, a value of $\alpha = 0.27$, as before, is used. Needing to be highlighted is the fact that, although alpha has not changed with respect to the previous test, here the performance of the method with respect to that previously obtained has degraded, passing from an SDNR=28.15 of the previous test, to 17.15dB. This proves, even more, that the Alpha chosen can be greatly valid for some conditions while not for others, remembering also that, theoretically, this parameter should not be influenced by the EMI conditions: powers and modulations. This behaviour contrary to the theory must be researched in the effects not considered by the tested model.

In the appendix, other measurement made by changing the frequency of the modulating signal can be observed .

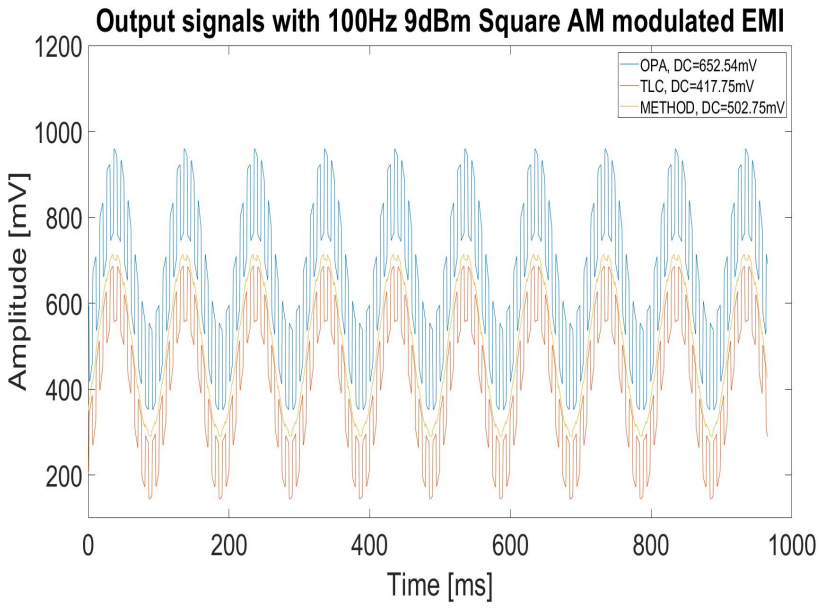


Figure 4.28: Outputs with 9dBm RF 100hz square wave modulated input

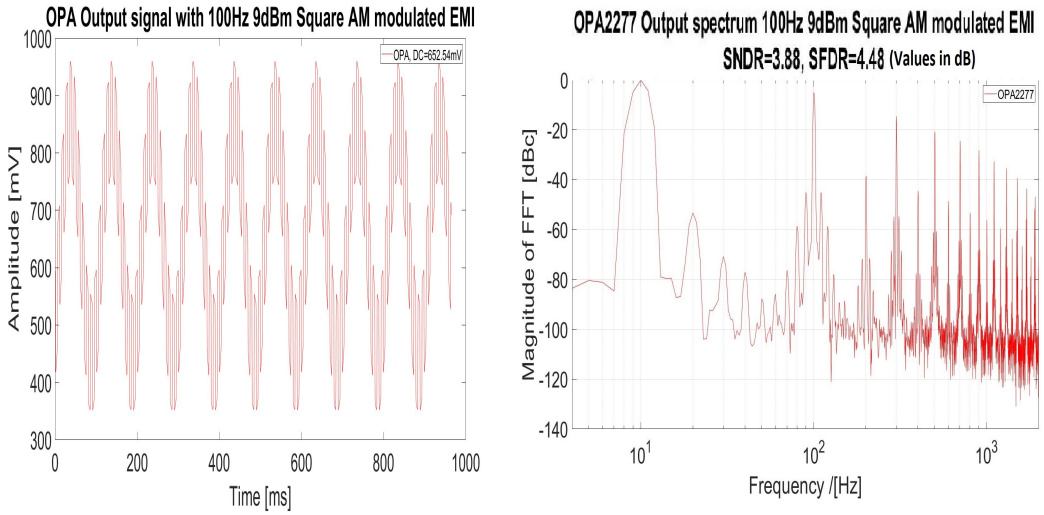


Figure 4.29: OPA output and spectrum with 9dBm RF 100Hz square wave modulated input

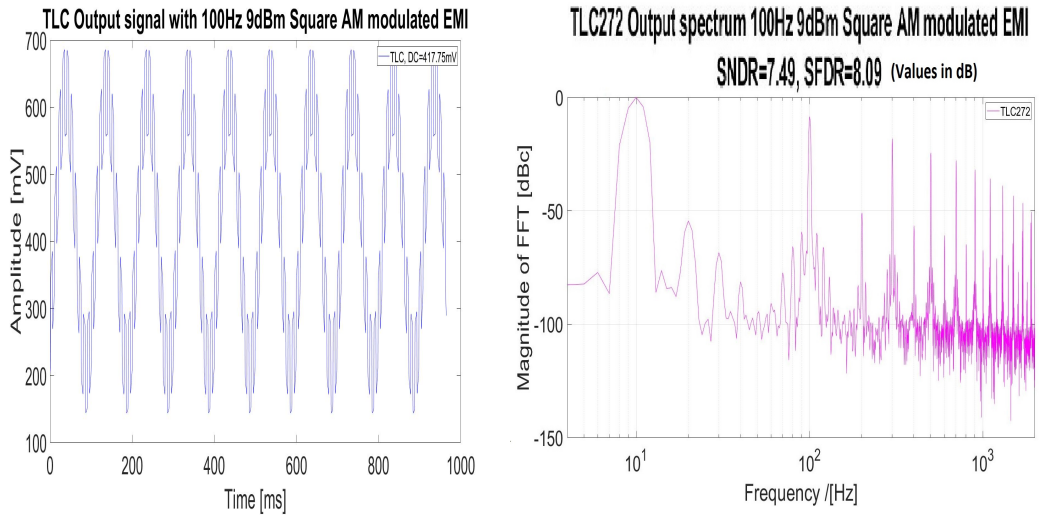


Figure 4.30: TLC output and spectrum with 9dBm RF 100Hz square wave modulated input

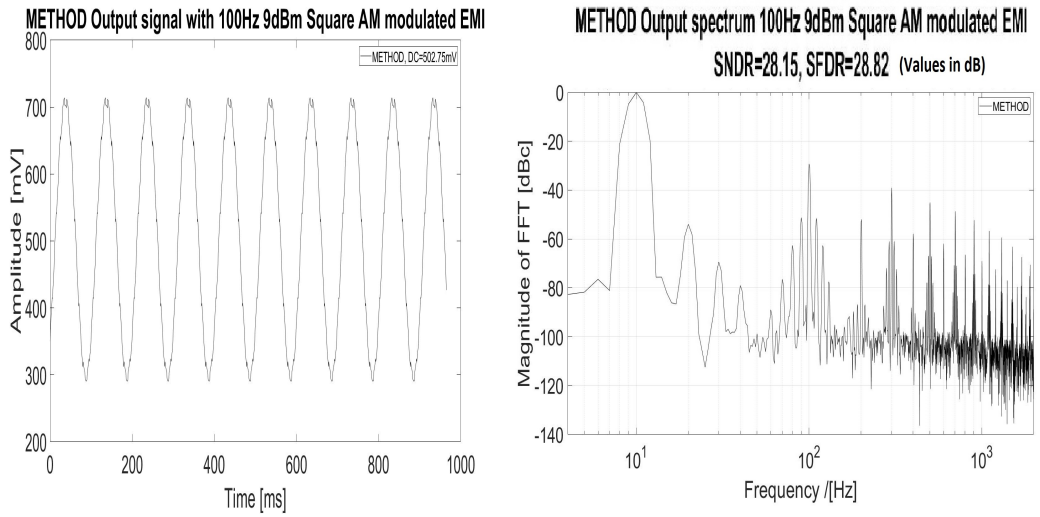


Figure 4.31: METHOD output and spectrum with 9dBm RF 100Hz square wave modulated input

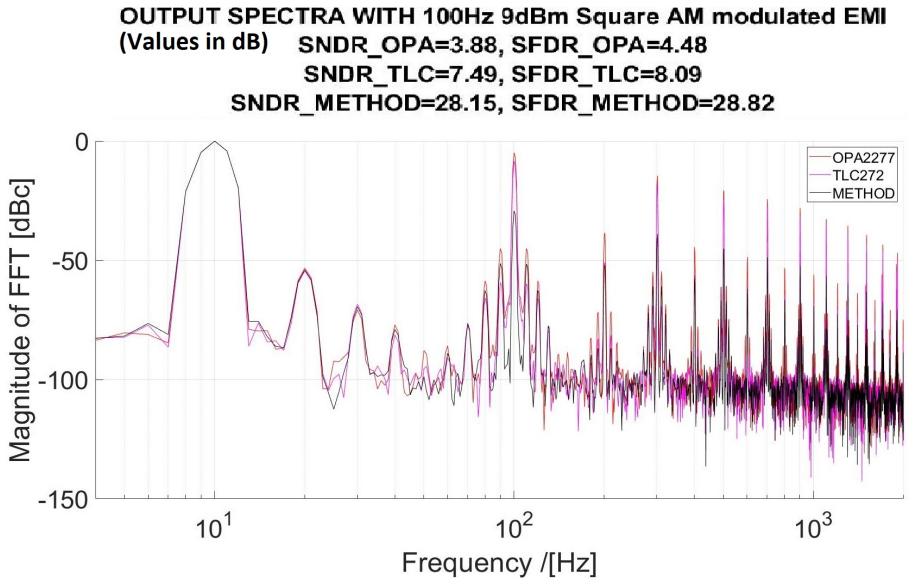


Figure 4.32: Output spectrum with 9dBm RF 100Hz square wave modulated input

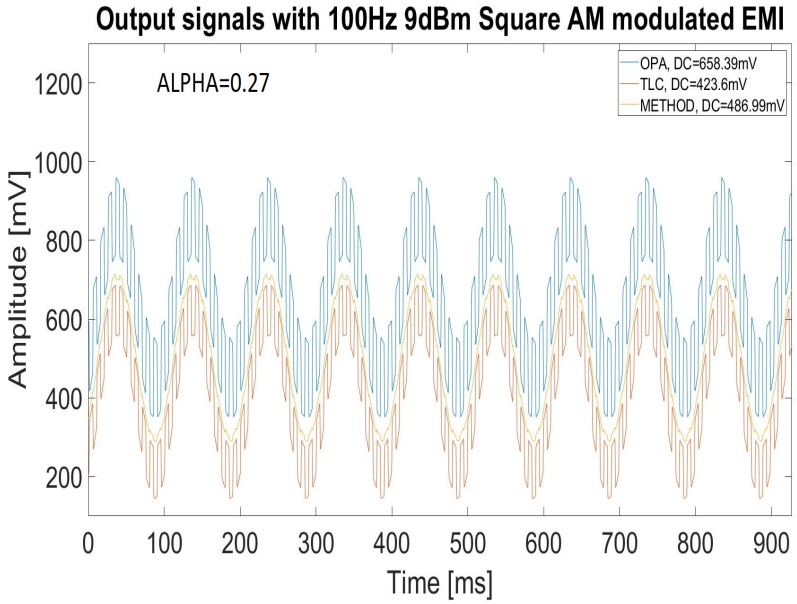


Figure 4.33: Outputs with 0dBm RF 100Hz square wave modulated input for $\alpha = 0.27$

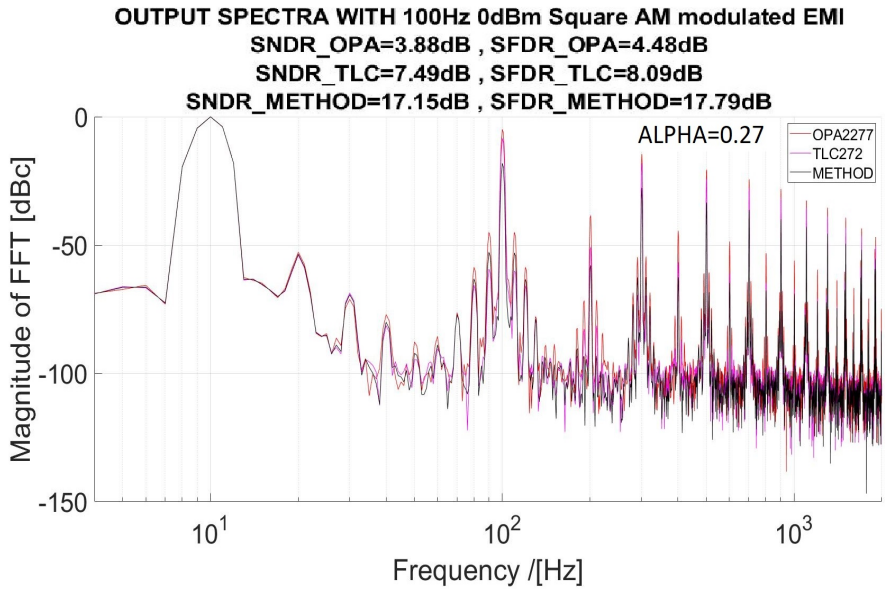


Figure 4.34: Output spectrum with 0dBm RF 100Hz square wave modulated input for $\alpha = 0.27$

4.8 RF AM sinusoidal modulated measurements

In this section, a set of measurements has been conducted for a 100MHz RF sinusoidal modulated input with a power of 0dBm. The 400Hz LF signal that modulates the EMI has been obtained internally by the HP generator. The modulating index m=100%. In figure 4.35, all the outputs coming from the amplifiers and the method can be seen and in 4.36, 4.37, and 4.38, the details of each output for time and for frequency domain is shown.

In figure 4.39, the spectra of all the signals can be observed. Also in this case, an improvement in terms of SNDR and SFDR is obtained. For the method, indeed, SNDR=32.79 and SFDR=32.86 exhibit an improvement of almost 10dB compared to the other amplifiers.

Worthy of note is the fact that, in the spectrum of the signals, more than one harmonic of the modulating signal can be seen. This, in fact, is not caused by the windowing effect on the 400Hz signal but depends on the harmonic distortion of the generator that, at that frequency, produces these impurities. The harmonics are of even order (800Hz, 1200Hz).

In the appendix, a similar test configuration for an injected power of 7dBm can be observed. In this case, the power has to be reduced below 9dBm because, at this power level, the voltages coming out from the amplifiers go beyond the allowed input range of the ADCs.

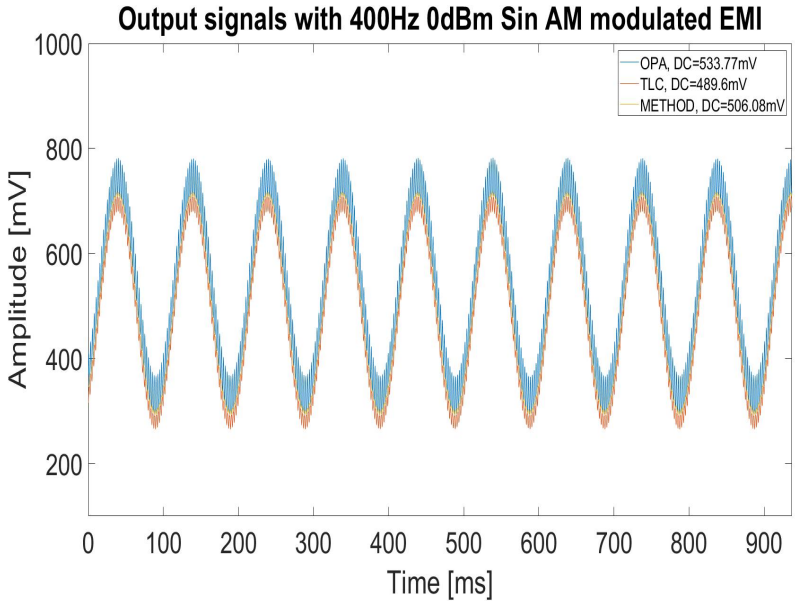


Figure 4.35: Outputs with 0dBm RF 400Hz sinusoidal wave modulated input

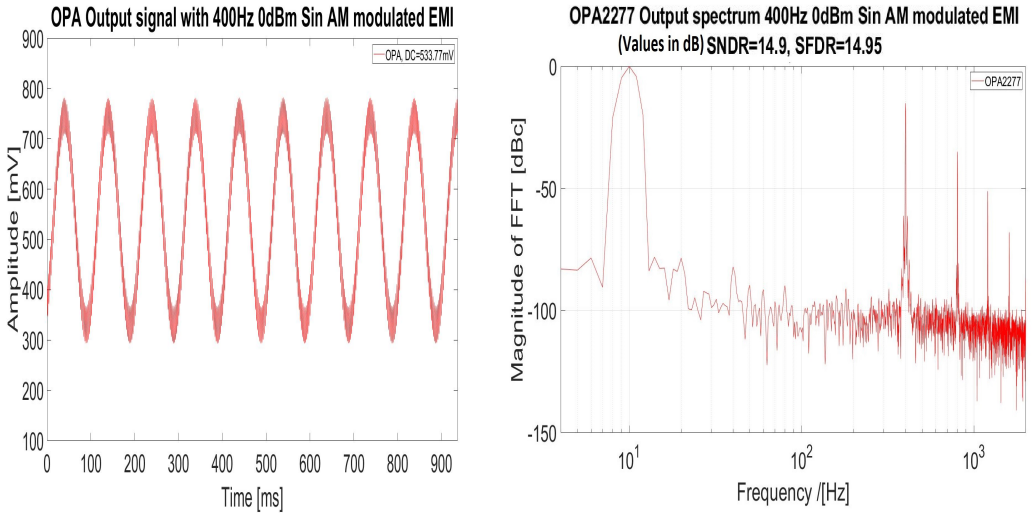


Figure 4.36: OPA output and spectrum with 0dBm RF 400Hz sinusoidal wave modulated input

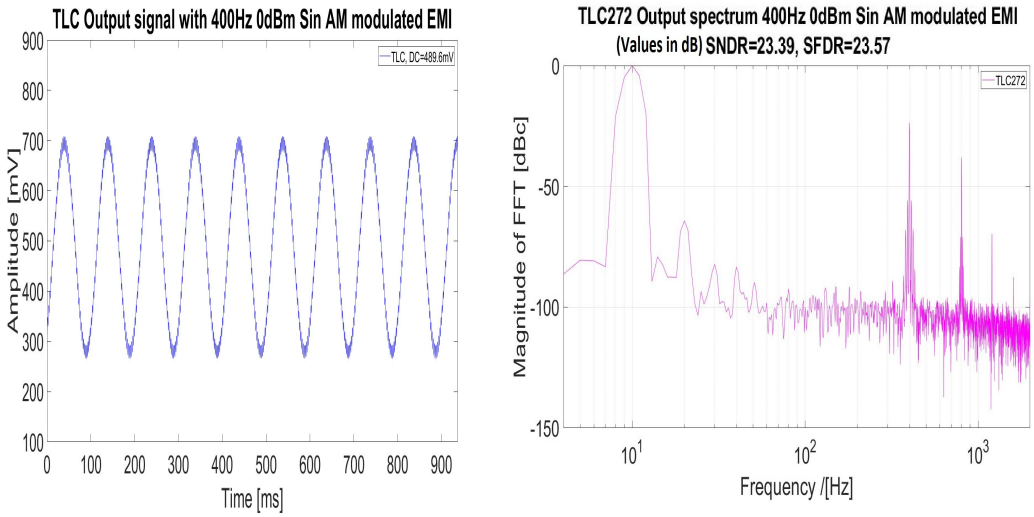


Figure 4.37: TLC output and spectrum with 0dBm RF 400Hz sinusoidal wave modulated input

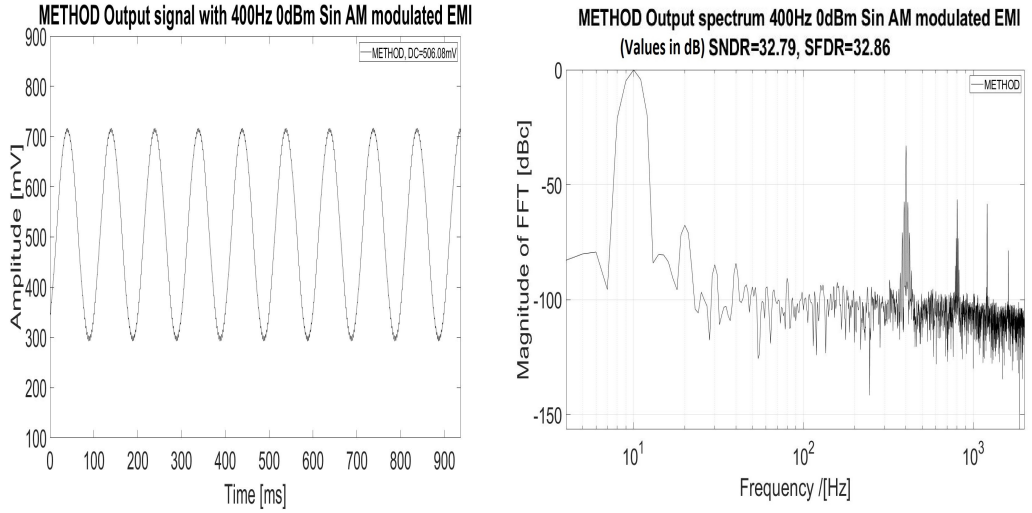


Figure 4.38: METHOD output and spectrum with 0dBm RF 400Hz sinusoidal wave modulated input

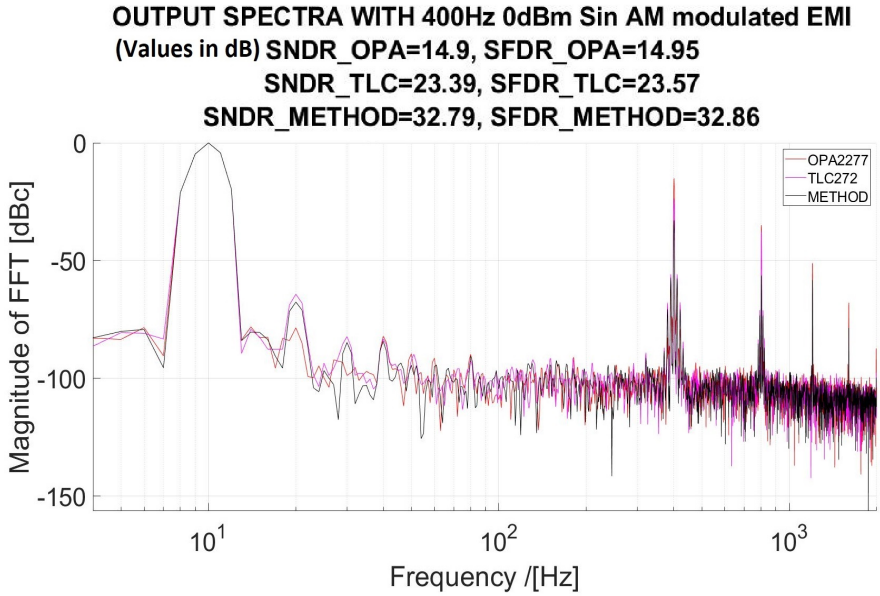


Figure 4.39: Output spectra with 0dBm RF 400Hz sinusoidal wave modulated input

4.9 RF AM exp modulated measurements

In this section, to thoroughly evaluate the effectiveness of the method in more practical situations, as well an exponential signal is modulating the 100MHz injected EMI. Indeed, generally, the RF is present for short periods of time and with an exponential damped behaviour. The case of a CW EMI is less common but, worth being evaluated as a worst case limit. The test has been conducted for 0dBm injected power and for 9dBm.

Here, it will be show a set of measurements made with a 100Hz exponential signal expressed as in equation 4.7 and a modulating index m=100%. In the appendix, one can find similar tests done with a more realistic 1kHz exponential wave, that could represent the activity of some switching supply devices.

In figure 4.40, the plot over time of all the signals together can be observed. In figures 4.41, 4.42, and 4.43, the details for each individual signal can be observed. In figure 4.44, the spectra of the all the signals has been plotted. For the method, both for the SNDR and SFDR, an improvement with respect to the TLC amplifier of 10dB can be observed, better results with respect to the OPA (SNDR= 41.62dB versus SNDR_OPA=23.81dB and SNDR_TLC=32.37dB and SFDR=43.84dB versus SFDR_OPA=26.48dB and SFDR_TCL= 35.38dB). Here, as well, the good effects of the proposed method can be appreciated, in particular with respect to the OPA.

Of note is the presence of some aliasing effects at the top border of the shown frequencies that depends on the presence of high order harmonics of the modulating signal. These spurious effects remain almost below -75dBc with respect to the fundamental signal and are attenuated by the method, as can be seen in figure 6.21 where they remain below -90dBc.

As for the previous measurement conditions, there is another slight difference worse with respect to the previous study [13] tests. As before, an improvement can be obtained by refining the employed alpha well. More comparable values can be observed, instead, for more similar test conditions (see the appendix for an injected 1kHz exponential modulated RF) where the difference with respect to the previous analysis is reduced (SNDR_METHOD=49.3dB versus 45.54dB of the previous study and SFDR_METHOD=49.44dB versus 55.23dB where the previous study still has better results, almost completely suppressing the 1kHz harmonic).

In the appendix, a test conducted with an injected 9dBm RF signal can be observed with the same modulating conditions.

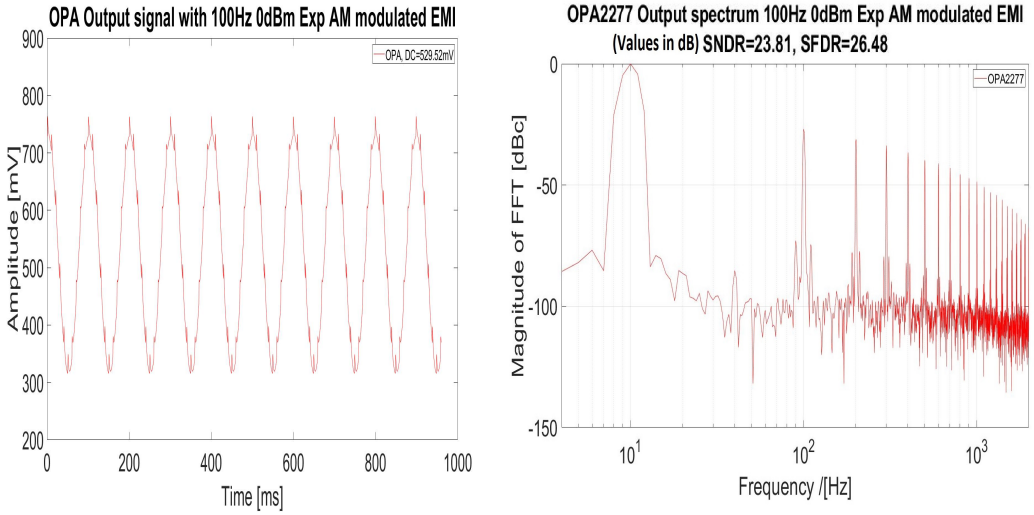


Figure 4.41: OPA output and spectrum with 0dBm RF 100Hz exponential wave modulated input

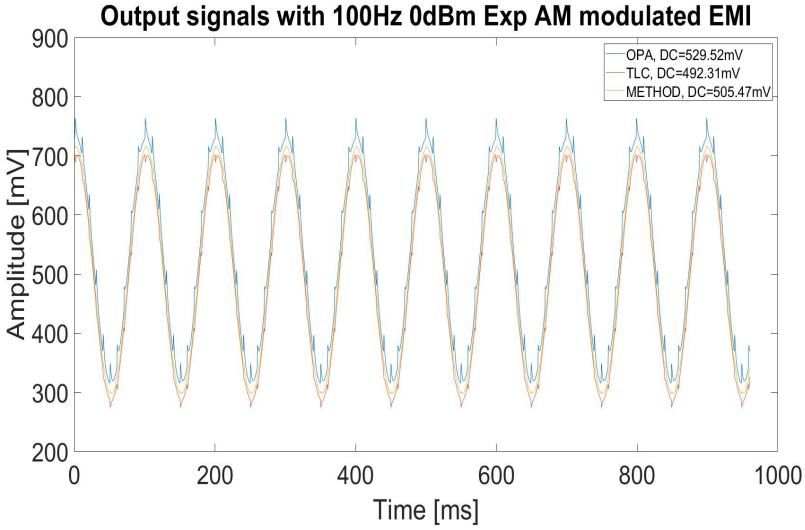


Figure 4.40: Outputs with 0dBm RF 100Hz exponential wave modulated input

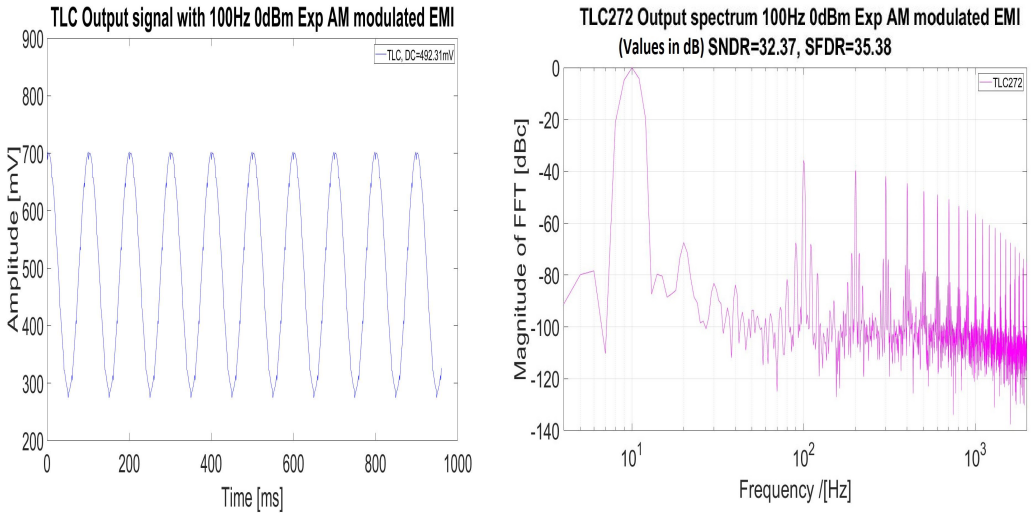


Figure 4.42: TLC output and spectrum with 0dBm RF 100Hz exponential wave modulated input

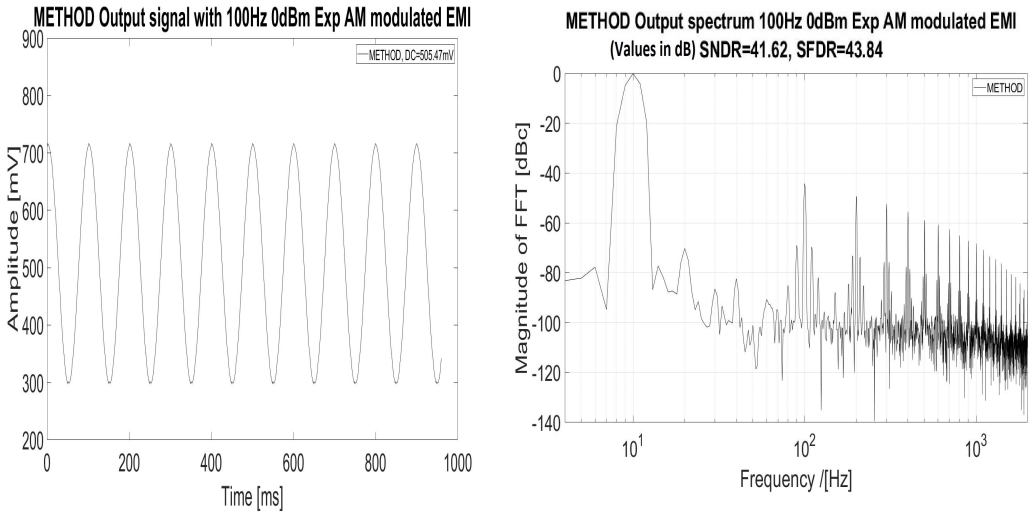


Figure 4.43: METHOD output and spectrum with 0dBm RF 100Hz exponential wave modulated input

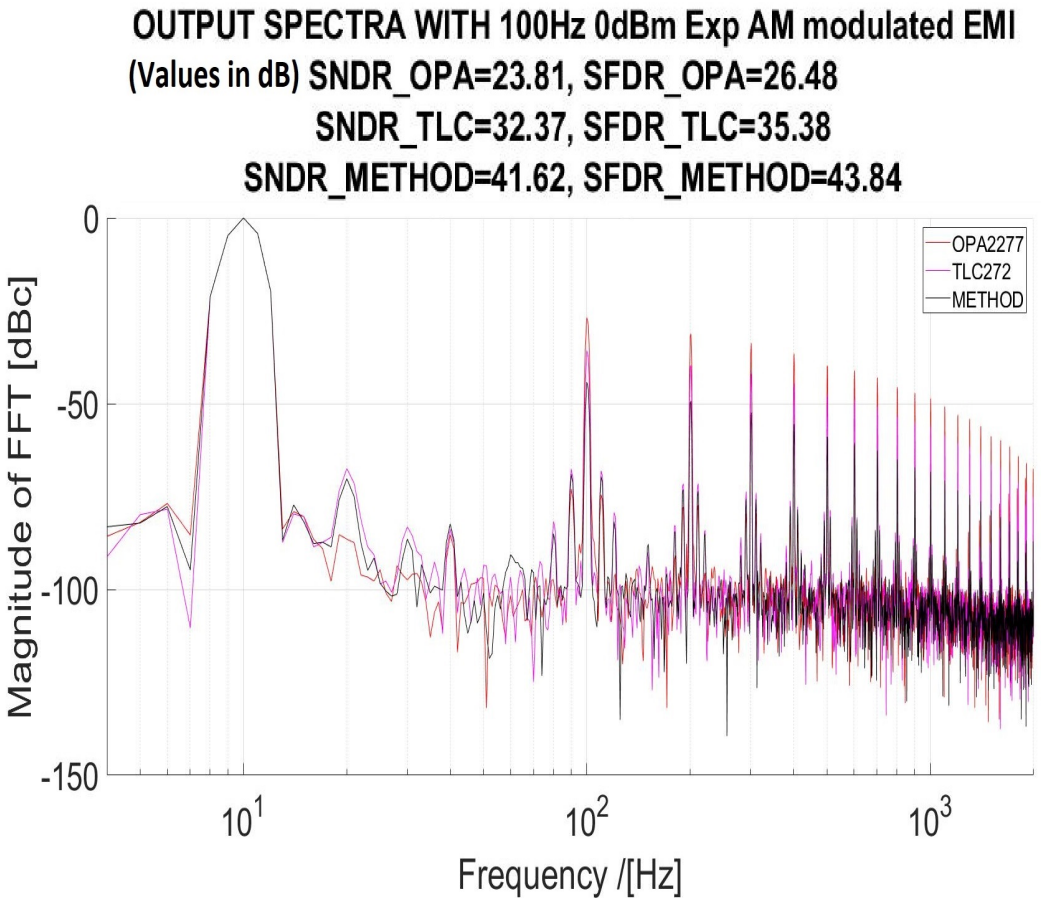


Figure 4.44: Output spectra with 0dBm RF 100Hz exponential wave modulated input

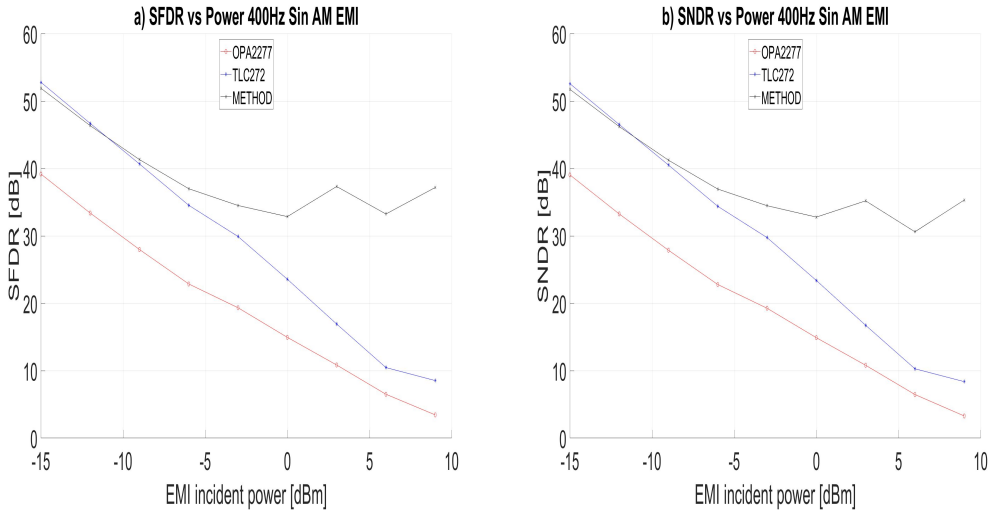


Figure 4.45: a)SFDR and b)SNDR vs power for injected 400Hz sinusoidal wave modulated 100MHZ EMI

4.10 SFDR and SNDR measurements

This section summarizes the measurements taken for various conditions of the EMI signal and the LF frequency modulating signal.

In figure 4.45, the results obtained for a sinusoidal modulating signal of 400Hz with respect to various injecting powers from -15dBm ($V_{EMI,pk} = 112mV$) to 9dBm ($V_{EMI,pk} = 1.78mV$) can be seen. In this case, for low injected EMI power, the method and the TLC behave similarly. This is because, for these values of power, the induced error is quite small (of the order of μV and comparable with the measurement uncertainly and quantization noise). This effect is also due to the fact that the TLC272 has high immunity with respect to the MCP6V02 employed in [13]. This, in turn, makes the method less effective for these levels of power.

An improvement of the results can be obtained by further reducing the α^* used, prioritizing TLC over the OPA, as already explained in the offset and alpha refining sections. It is worth remembering that the alpha employed is a compromise, obtained by minimization, and it is not truly independent of the EMI conditions. So, modifying it can increase effectiveness in some specific situations.

As can be seen in figure 4.45, superior results can be obtained with the method for powers higher than -10dBm where the SFDR and SNDR increase by almost 35dB at +3dBm. This confirms, as will be see again in the next figure, the effectiveness of the proposed technique.

Figures 4.46 and 4.47 show the results for the same 400Hz sine wave that modulates the EMI for a variation of the modulating index m (see equation 4.6),

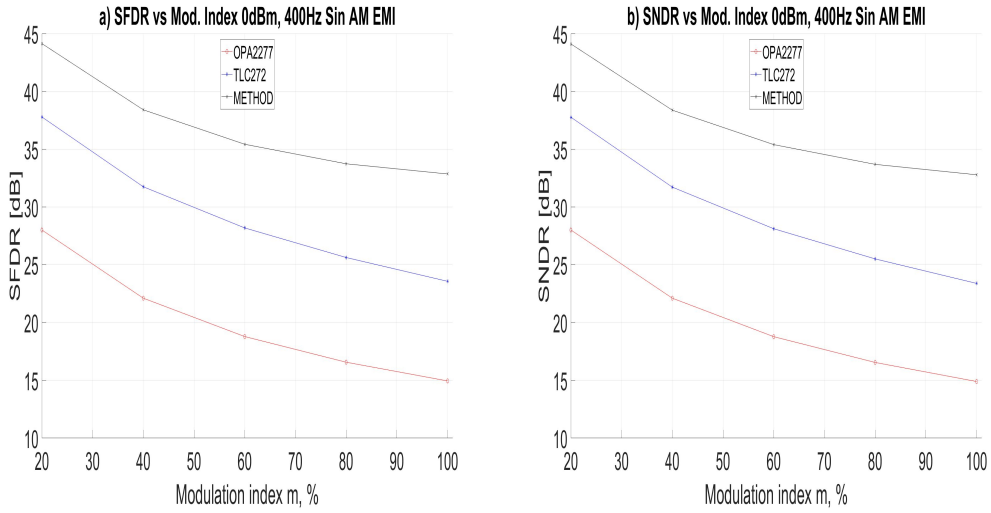


Figure 4.46: SFDR and SNDR vs modulation index for injected 400Hz sinusoidal wave modulated 100MHz EMI @ 0dBm

in case of 0dBm and 7dBm injected power. Here, the 9dBm power cannot be used, since the induced error was outside the input voltage limits of the ADCs. Since the RF power is above the -10dBm limit explained for the sinusoidal case, good results are obtained for all the tested conditions with a minimum SFDR of 32dB and a minimum SNDR of almost 30dB, all well above the results for both the amplifiers alone.

In figure 4.48 the measurement results can be observed in the case of 0dBm 400Hz-sinusoidal modulated RF injection for a frequency sweep from 50MHz to 500MHz. In this case one can observe that for a frequency of 100MHZ the TLC shows an increased immunity and this, in turn, reduces the effectiveness of the method since the same alpha has been used. These conditions can be evaluated during the system test and the alpha set according to what type of EMI is supposed to pollute the environment. So, for example, if it is known that the place where the system should work is contaminated with RF with a frequency below 100MHz, an Alpha must be chosen. On the other hand, if the frequency is above this limit, another value of alpha must be taken into consideration.

The same measurement conditions have been maintained for a sinusoidal modulating signal of 1KHz. As will be discussed later, the value of this frequency is partially influenced by the LPF. However, this effect is present for all the outputs in a similar way and, as such, does not influence globally the results. In figure 4.51, it must be noted that the increasing trend of the method is for a modulation index above 80%. This fact can be explained by the increasing induced error that is comparable to an increase in the injected power.

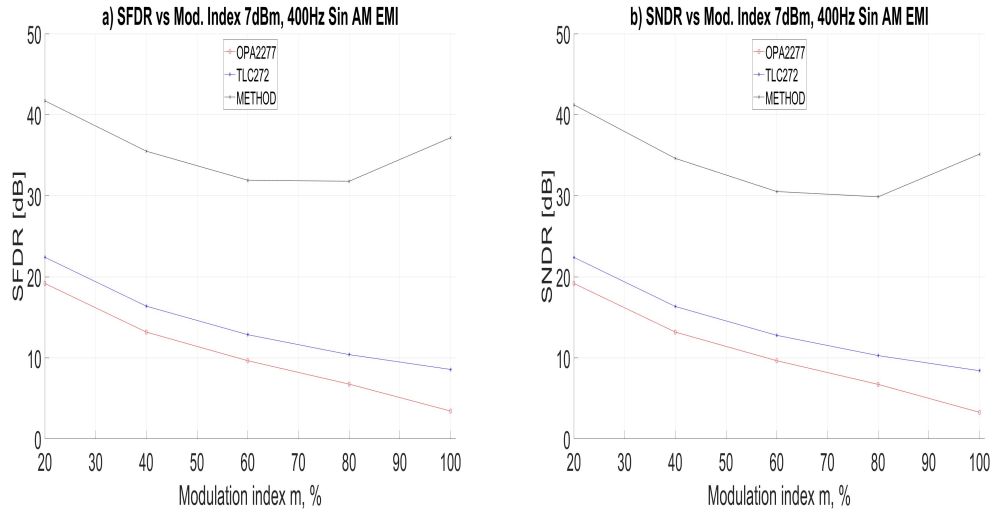


Figure 4.47: SFDR and SNDR vs modulation index for injected 400Hz sinusoidal wave modulated 100MHz EMI @ 7dBm

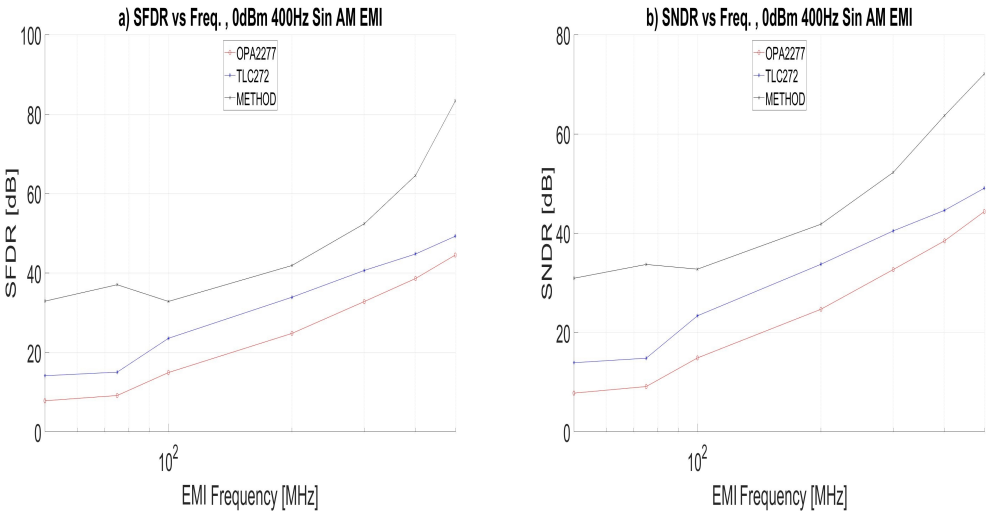


Figure 4.48: SFDR and SNDR vs frequency for injected 400Hz sinusoidal wave modulated EMI @ 0dBm

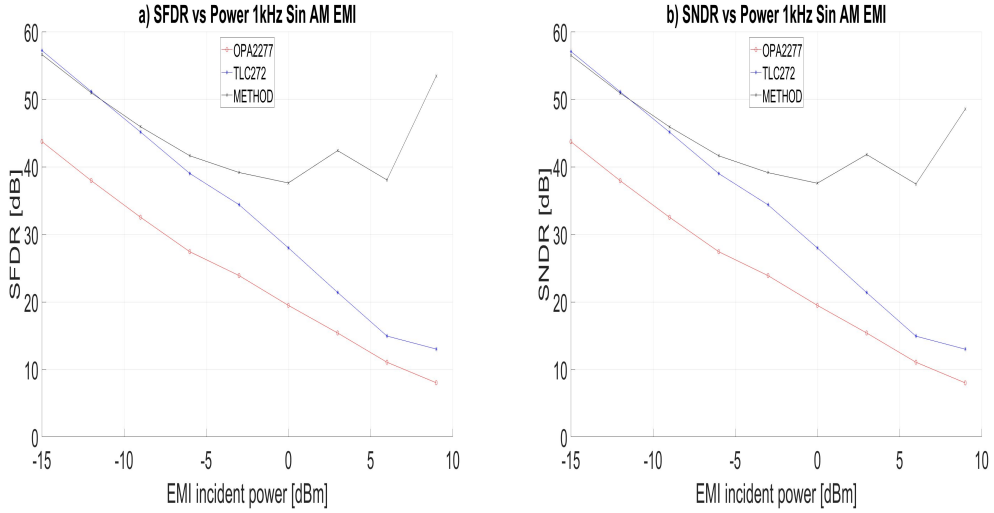


Figure 4.49: SFDR and SNDR vs power for injected 1kHz sinusoidal wave modulated 100MHz EMI

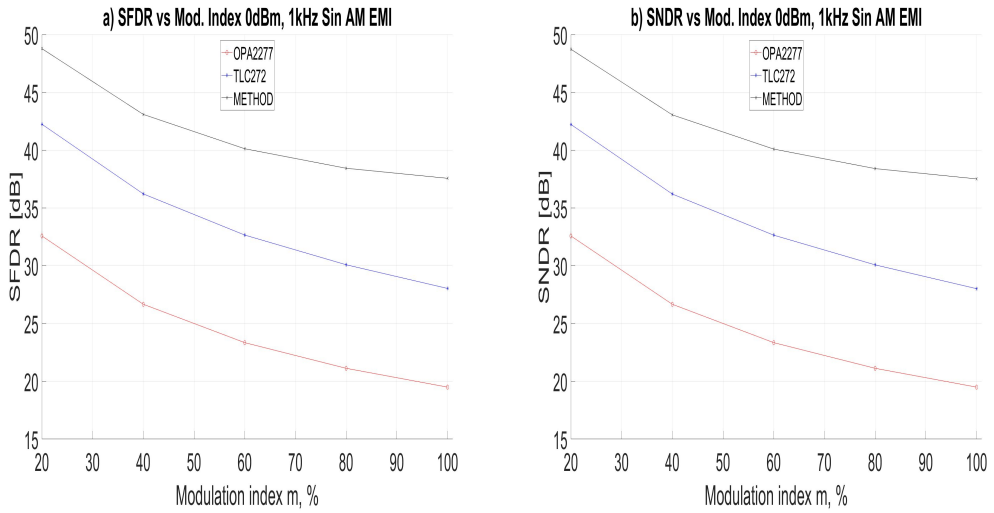


Figure 4.50: SFDR and SNDR vs modulation index for injected 1kHz sinusoidal wave modulated 100MHz EMI @ 0dBm

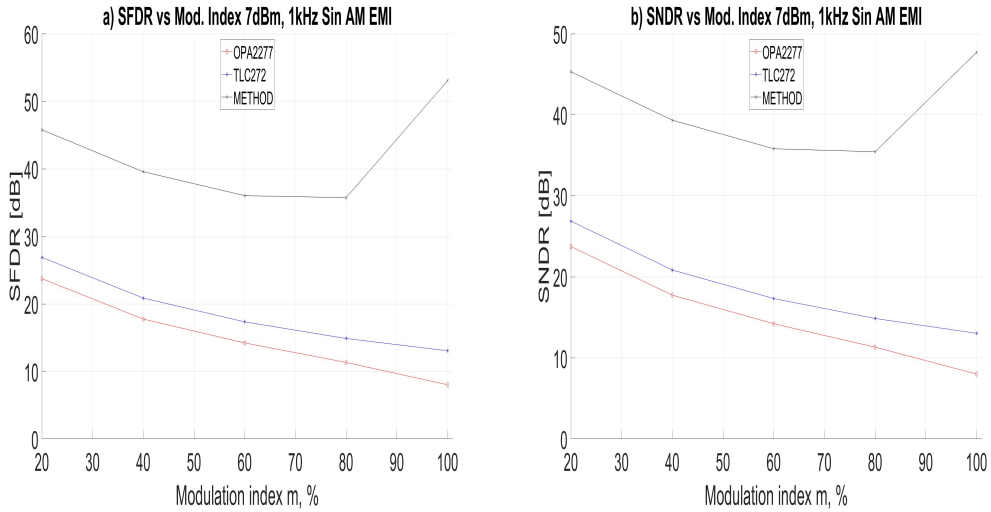


Figure 4.51: SFDR and SNDR vs modulation index for injected 1kHz sinusoidal wave modulated 100MHz EMI @ 7dBm

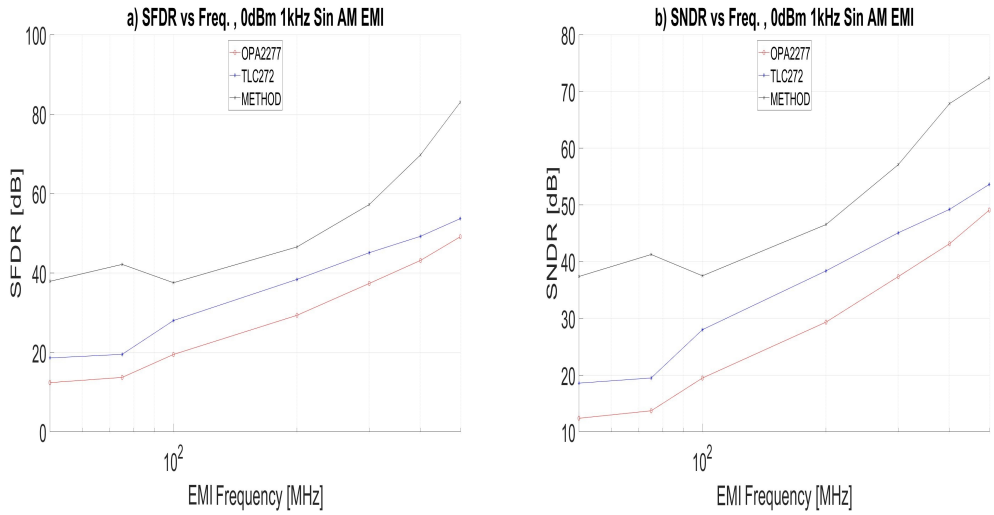


Figure 4.52: SFDR and SNDR vs frequency for injected 1kHz sinusoidal wave modulated EMI @0dBm

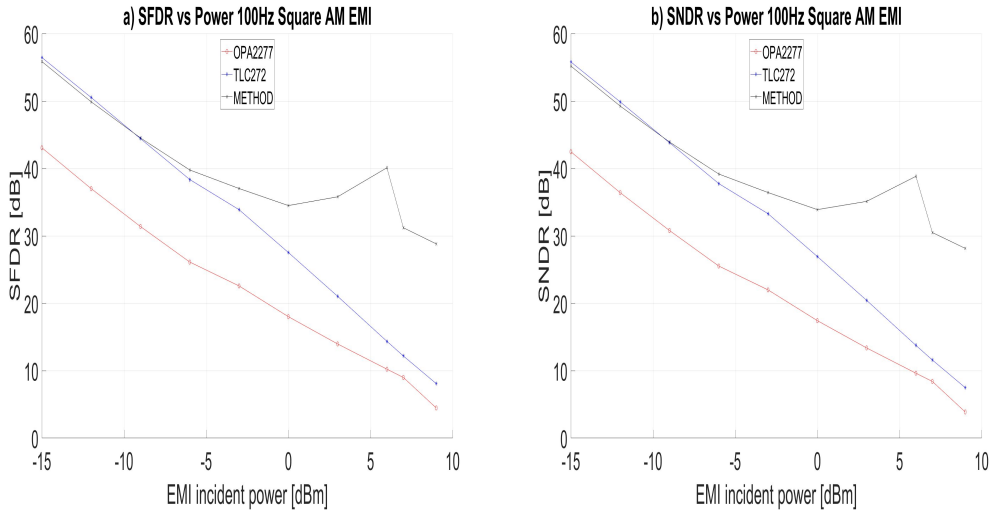


Figure 4.53: SFDR and SNDR vs power for injected 100Hz square wave modulated 100Mhz EMI

In figures 4.53, 4.54, 4.55 and 4.56, the same test has been performed for a 100MHz RF signal modulated by 100Hz 1V of peak voltage unipolar (0-1V) square wave. The same type of test has been conducted. Altogether, there was also good result also in these cases where an increase in both the SFDR and SNDR is obtained.

In the appendix, a similar test has been conducted for a 1KHz square wave that modulates the RF signal.

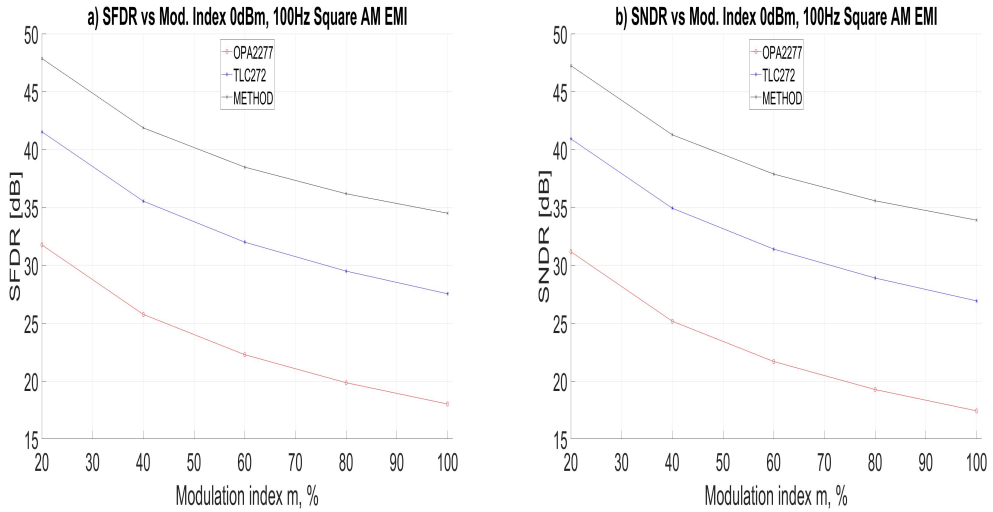


Figure 4.54: SFDR and SNDR vs modulation index for injected 100Hz square wave modulated 100Mhz EMI @0dBm

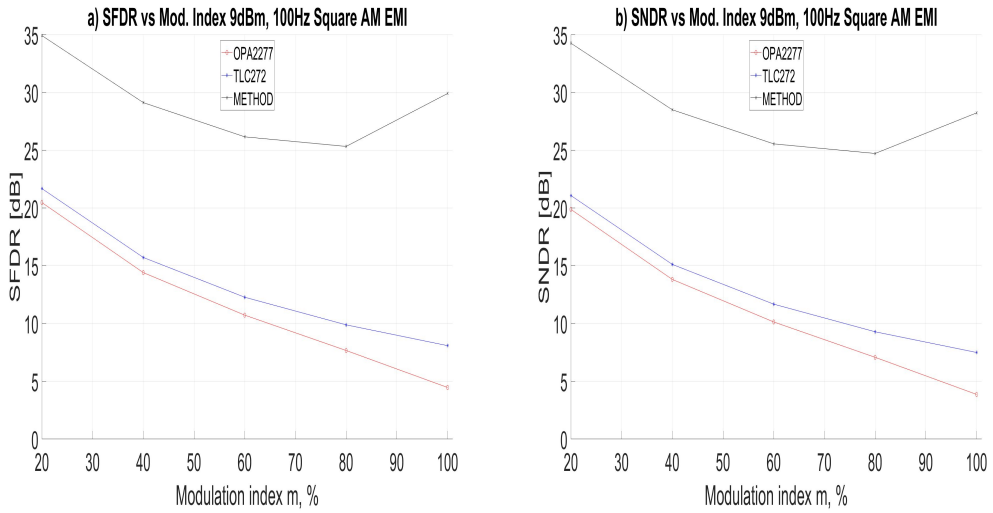


Figure 4.55: SFDR and SNDR vs modulation index for injected 100Hz square wave modulated 100Mhz EMI @9dBm

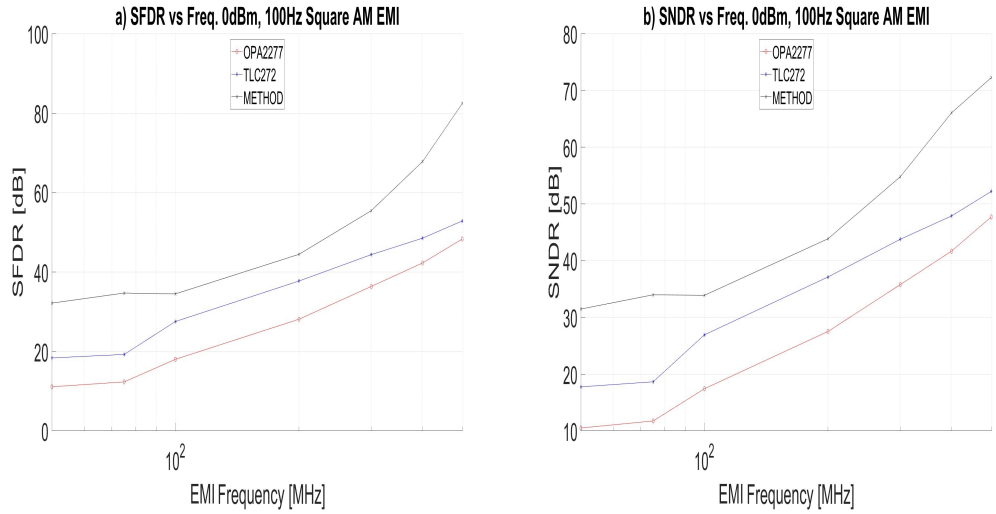


Figure 4.56: SFDR and SNDR vs frequency for injected 100Hz square wave modulated EMI @ 0dBm

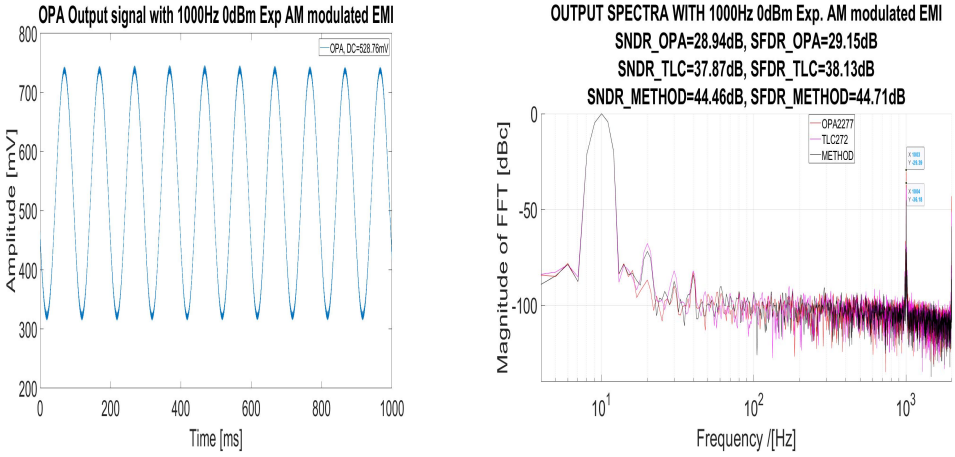


Figure 4.57: Presence of induced effects for a 1kHz exponential modulated 100MHz EMI @ 0dBm - NO additional LPF

4.11 Effects of frequency on the signal with and without the LPF

In this section, some effects will be discussed that have been observed during measurement.

4.11.1 Internal ADC digital filter

During the exponential test it was noted that for frequencies higher than one half of the sampling frequency the superimposed demodulated exponential signal seemed to vanish completely, also occuring in case with the absence of additional LPF.

To highlight this, all of the measurements below have been conducted without the additional LPF.

In figure 4.57, indeed, the exponential wave that modulates the EMI has a frequency of 1kHz and can be noticed at the top and bottom of the LF 10Hz sine-wave and on the spectra of the signals with a power of -29.39dBc for the OPA and -36.18dBc for the TLC.

Instead, in figure 4.58, the exponential frequency has been increased to 2.5kHz and the LF signal seemed to be completely clean. The spectra confirm this trend, indeed, the replicas of the 2.5kHz that drop to 1.5kHz have a power of -46.71dBc for the OPA and -51.82dBc for the TLC. This confirms high attenuation in less then an octave (500Hz, from 2kHz to 2.5kHz).

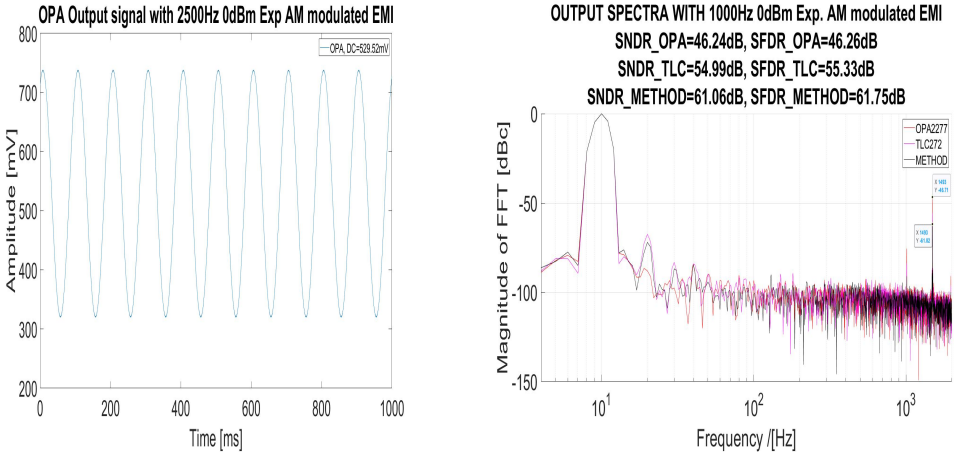


Figure 4.58: Filtering effect of the internal digital filter of the ADC for 2.5kHz exponential modulated 100MHz EMI @ 0dBm - No additional LPF

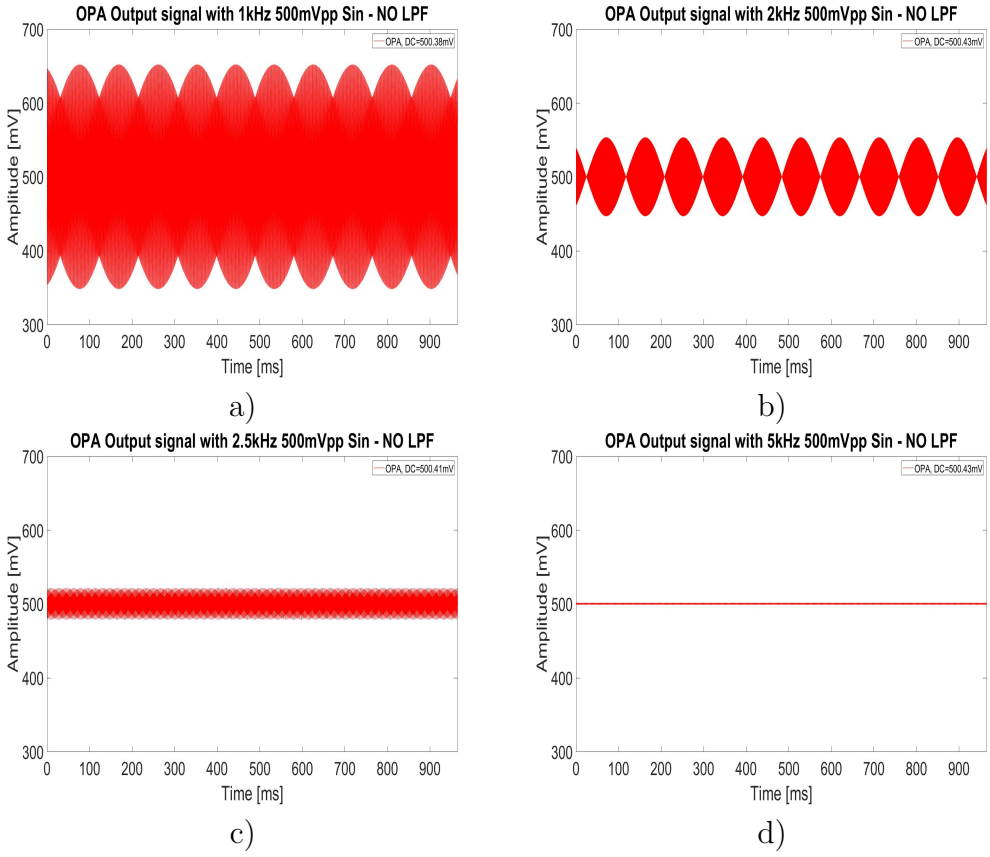


Figure 4.59: Filtering effect of the internal digital filter of the ADC for sinusoidal input - No additional LPF. a)1kHz signal, c)2kHz - cut frequency, c)2.5kHz, d)5KHz

Investigating the effect, a 1kHz-5kHz LF sinusoidal signal with $V_{peak} = 500mV$ at the input of the double amplifier board (that has been attenuated at the output due to the voltage divider) it has been applied, see figure 4.59. Under this condition, it has been noted that the ADC digital decimator, probably, acts as high order filter cutting all the frequencies above one half (2000Hz) of the set sampling frequency (here 4000Hz). This is evident comparing figure a) and figure b) where the signal is highly attenuated around 2kHz and above.

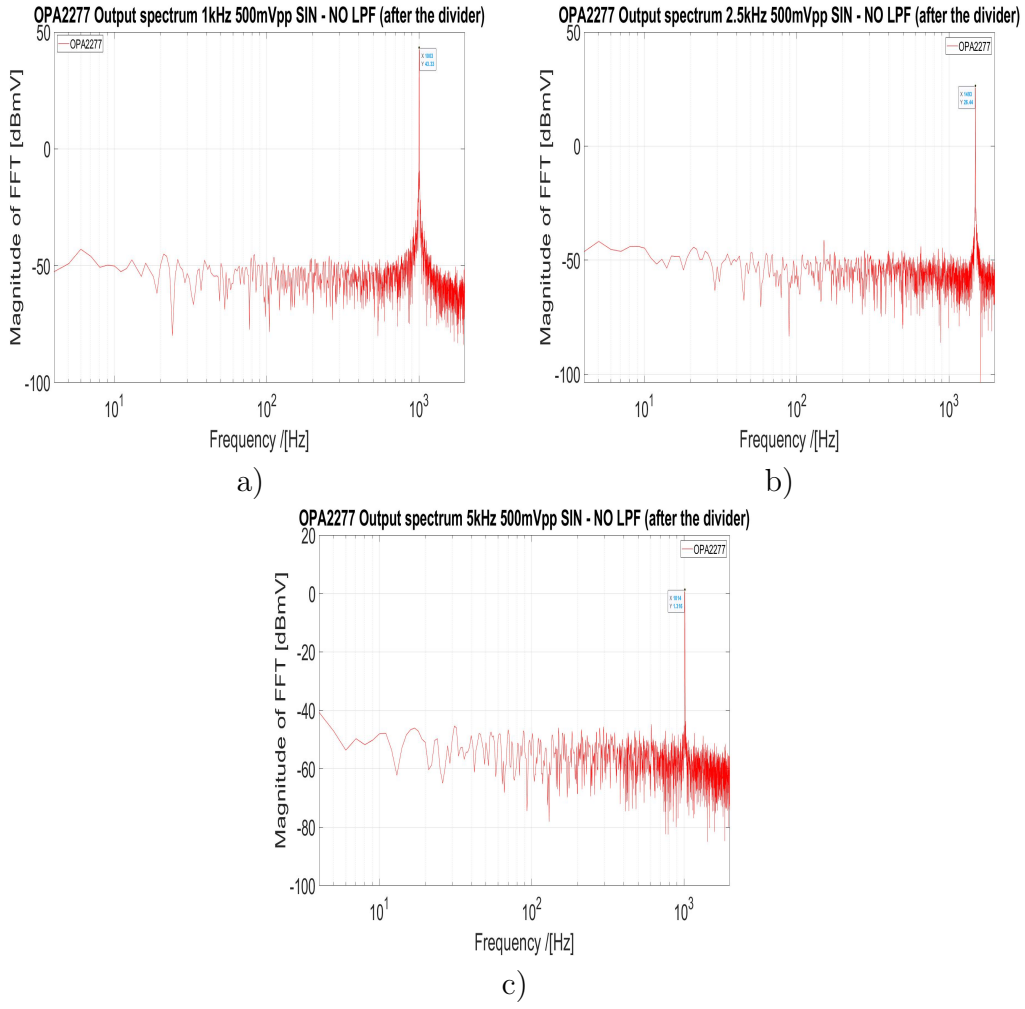


Figure 4.60: Spectra of the OPA amplifier at various frequencies to highlight the filtering effect of the internal digital filter of the ADC for sinusoidal input - No additional LPF. a)1kHz signal, d)2.5kHz c)5KHz

For checking purposes, the spectra of the signals at different frequencies have been plotted in figure 4.60, (values on the y axis in dBmV). Here We observe that

before the cut-off frequency, at 1kHz, the signal is of about 43.33dBmV, at 2.5kHz it has been attenuated to almost 20dB, and, at 5kHz (nearly an octave above 2kHz), the signal is attenuated to 1.3dBmV, with a global attenuation of about 40dB at the octave. It is to be noted that the frequencies above 2kHz drop in the lower part of the spectrum for replica i.e., the 5kHz can be found at 1kHz and the 2.5kHz at 1.5kHz, as can be seen in the figure.

This behaviour permits avoiding the use of the LPF that, however, has been utilized for all the above measurements.

4.11.2 The additional low pass filter

To test the effect of the additional LPF, an LF signal is applied as before ($V_{peak} = 500mV$ frequency from 100Hz to 5kHz) and the behaviour evaluated around the estimated cut-off frequency of 1kHz, see figure 4.61. This filter is, actually, not completely useless since it permits to further reducing the aliasing and, protects the micro-controller that, without, has been shown to have execution problems and blockages during power and frequency changes. Because its effects start near 1kHz, the measurements made for modulating frequencies around this value have the results influenced by the filter. As explained above this effect is negligible for the conducted tests since it influences all the sampled channels simultaneously.

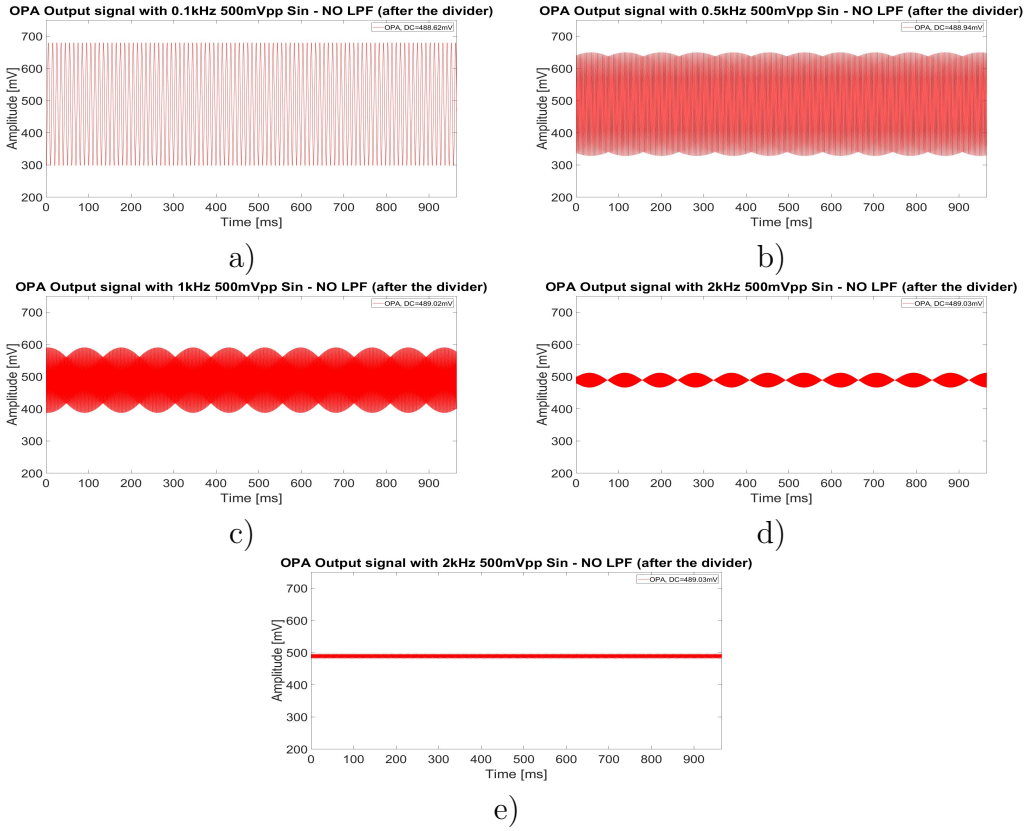


Figure 4.61: Additional LPF filtering effect for sinusoidal input. a)100Hz signal, b)500Hz effects already visible c)1kHz, d)2kHz, e)2.5KHz

In figure 4.62, the spectra of the acquired channels can be observed. Here one can see a power of 45.26dBmV for a frequency of 100Hz, 43.07dBmV for a frequency of 500Hz (in this case the effect of the filter has already started with an

attenuation near 2dB, close to the cutoff frequency) and 40.14dBmV at 1KHz. As can be seen, the actual true cutting frequency of the filter is between 500Hz and 1KHz, as expected by the components tolerances.

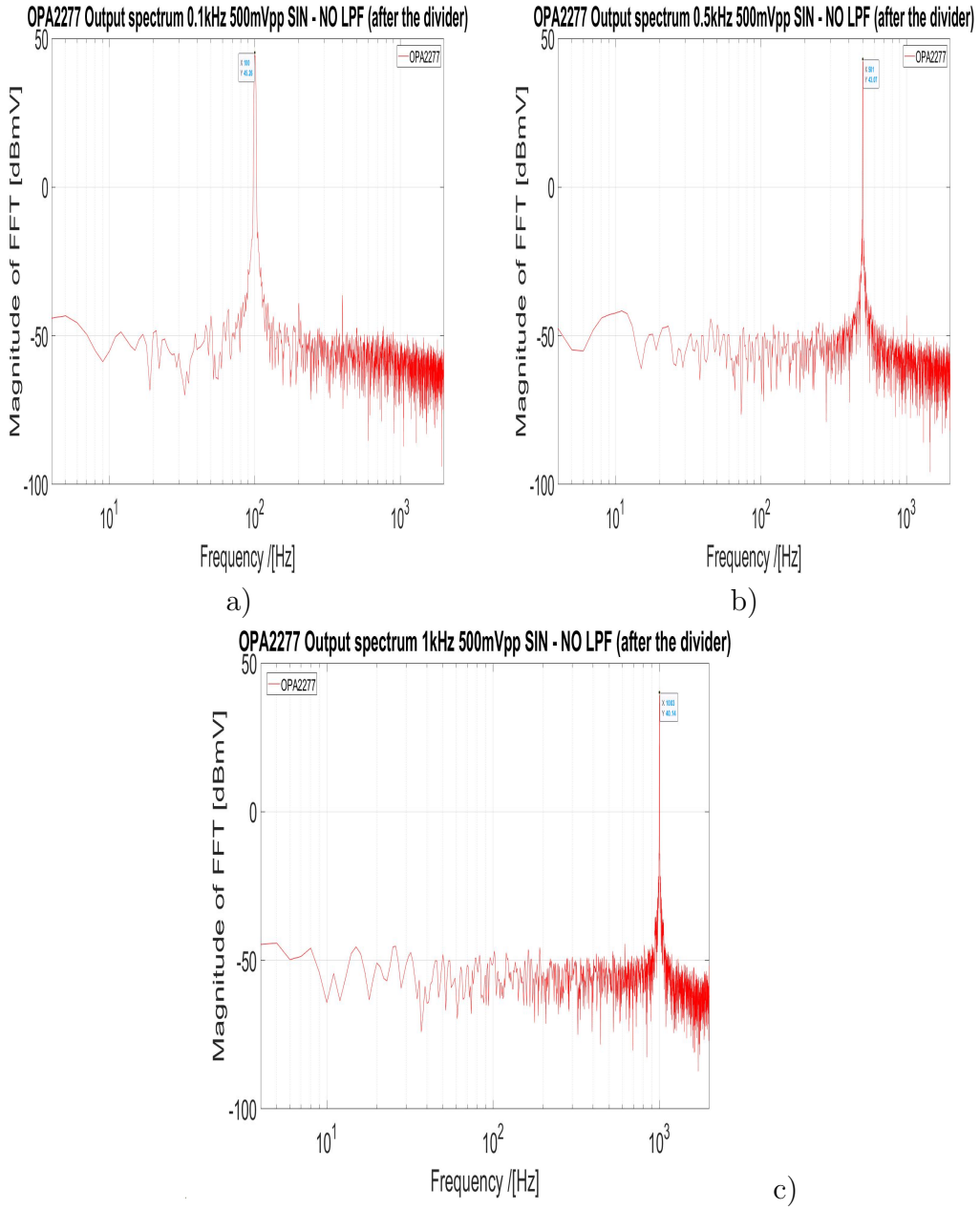


Figure 4.62: Spectra of the OPA amplifier at various frequencies to highlight the additional LPF filtering effect for sinusoidal input. a)100Hz signal, b)500Hz effects already visible c)1kHz

CHAPTER 5

Conclusions

Nowadays the EMI susceptibility of electronic devices has become more and more important, also due to the increase in electromagnetic pollution. The operational amplifier being broadly employed in analog applications is considered to be one of the most sensitive devices, and for this reason, the effects of RF on it have been widely investigated.

After illustrating the most important origins of distortions at the output of the amplifiers in the feedback configuration, nonlinearities and SR, an insight into the available solutions has been taken into consideration. Although what is proposed in the literature can effectively solve or considerably reduce the effects of the electromagnetic interference on the opamps, drawbacks are always present. Some of the discussed techniques can give good results in term of EMI rejection or EMI cancellation through the adoption of some sort of filtering effect but, at the same time, they can reduce the performance of the device in terms of CMMIR or phase margin. Advanced amplifier solutions have been studied to overcome these issues. The price to pay is, however, an increase in the complexity of the circuits and the requirements for nonstandard topologies. This, in turn, can bring with it increased design time, reduced technological portability, and the need for a precise transistor tuning, efforts that do not follow standards and consolidated procedures. For this reason, an additional solution has been, presented.

Following some of the methods proposed for the integrated case, an EMI susceptibility reduction technique, based on the cancellation of induced effects in devices that show opposing behaviour to EMI, has been discussed. From this, the design of an alternative solution that does not require circuit modifications has been presented. The complete developed system can, indeed, easily employ standard, off-the-shelf operational amplifiers and non-dedicated digital acquisition devices, and exploits its effectiveness through a sampling process and digital post processing computation. OPA2277 and TLC272 opamps, in addition to an ADuC7061

which hosts a Sigma-Delta ADC, have been utilized and properly configured to exploit the explained method.

Apart from some limitations in terms of communication speed, which have limited employment to relatively low frequency applications, an acquisition front-end system has been developed and tested, showing good results in terms of EMI susceptibility. It has been highlighted that the critical process of Alpha parameter choice is what gives form to the method. For this reason, different measurements have been performed to investigate the possible computation solutions and to determine how they can influence the response of the system.

The high satisfactory effectiveness of the presented method has been demonstrated with laboratory testing, and the results compared for a variety of conditions. It has been pointed out, and confirmed with some of the results for different EMI injected signals, that the Alpha parameter, although in theory not dependent on the RF at the input of the amplifiers, can deeply influence the results with respect to the power or the frequency employed. These considerations have suggested the evaluation of Alpha while keeping in mind some possible conditions in which the circuit would have to operate.

An suggestion on the possible reduction of employed components has been done in the section regarding analysis of the filtering behaviour of the micro-controller used. In particular with an eye on mass production, exploiting an already present capability can permit a reduction in cost.

Further improvement of the system presented in this thesis could be the substitution of the micro-controller used here with another that permits sampling of more than two channels at the same time or that permits one to obtain samples at the same time instant without sacrificing the sampling speed. This could be useful in the development of an automatic calibration system that can correctly tune the employed Alpha, perhaps under different working conditions or even adapting to outside stimuli. Other future developments can be centered around the design of a pre-assembled PCB board that can transparently give the user the benefits of this technique and the ability to use them in different applications, while setting aside some additional memory capabilities and advanced, better performing communication systems.

CHAPTER 6

Appendix

6.1 Figures

The ADuC connections

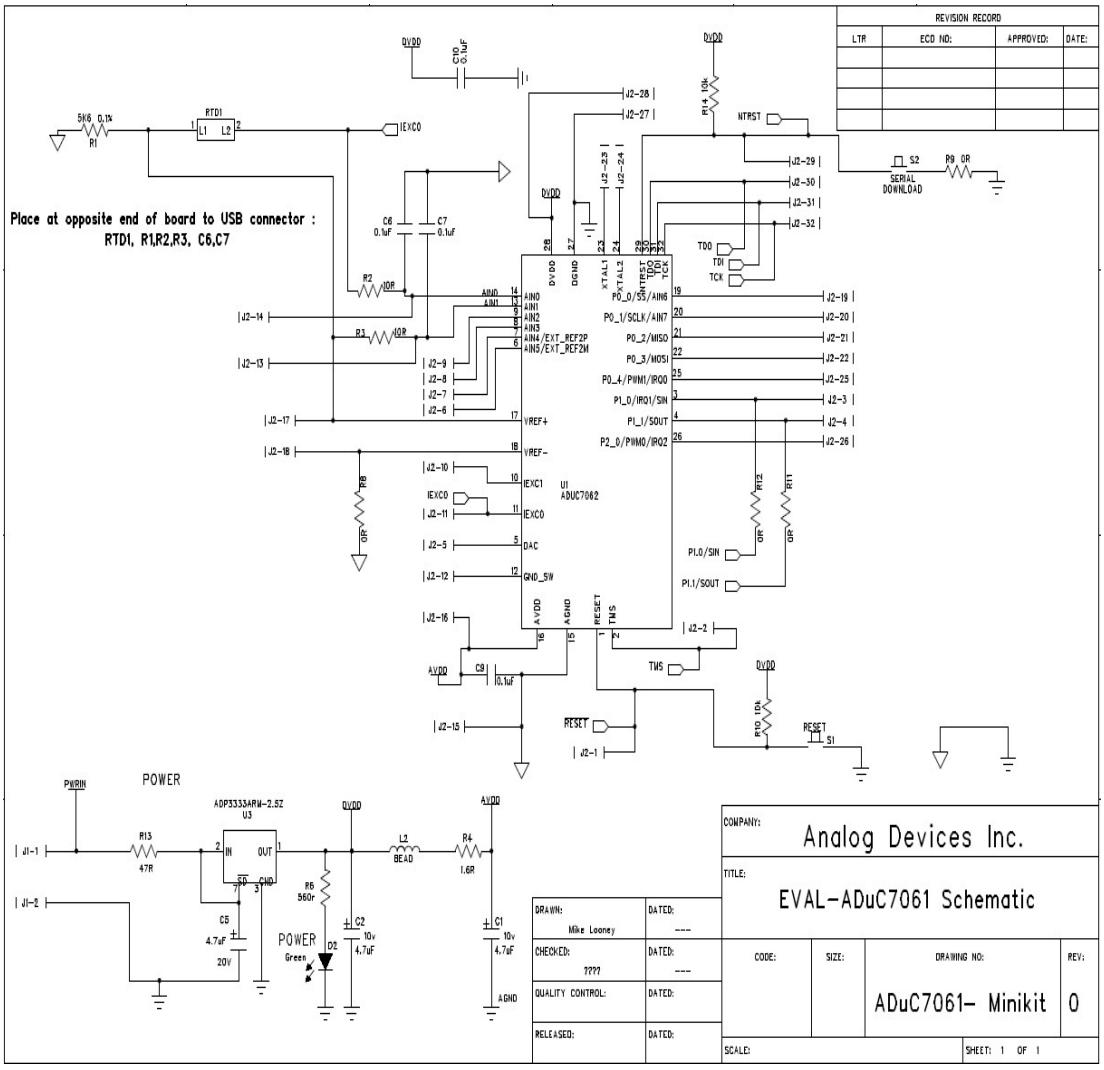


Figure 6.1: The connections on the board from [36]

6.1.1 RF 1kHz square-wave modulated plots 0-9dBm

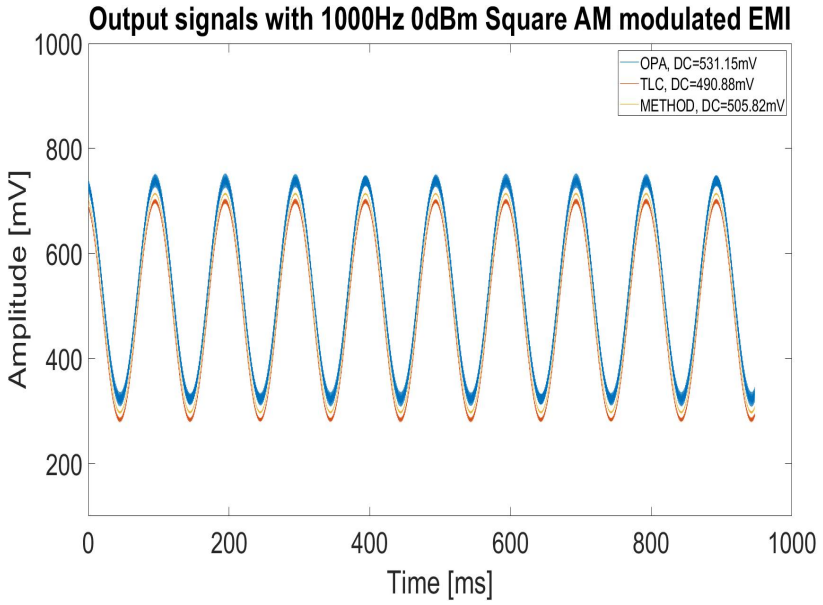


Figure 6.2: Outputs with 0dBm RF 1000hz square wave modulated input

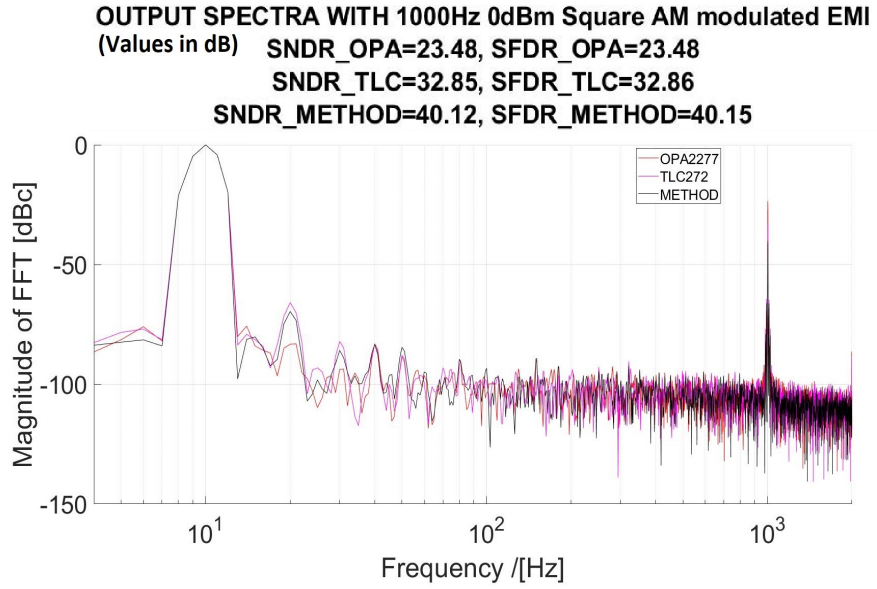


Figure 6.3: Output spectrum with 0dBm RF 1000Hz square wave modulated input

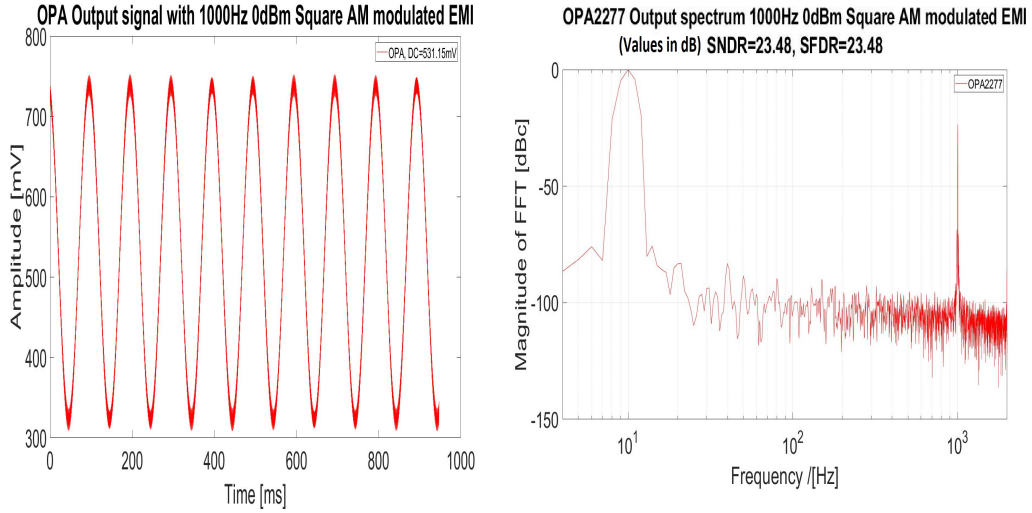


Figure 6.4: OPA output and spectrum with 0dBm RF 1000Hz square wave modulated input

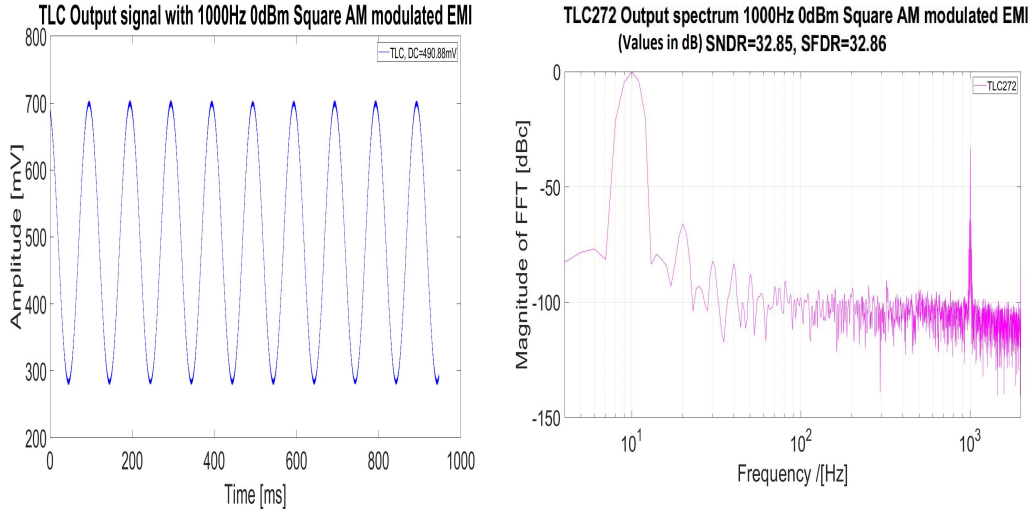


Figure 6.5: TLC output and spectrum with 0dBm RF 1000Hz square wave modulated input

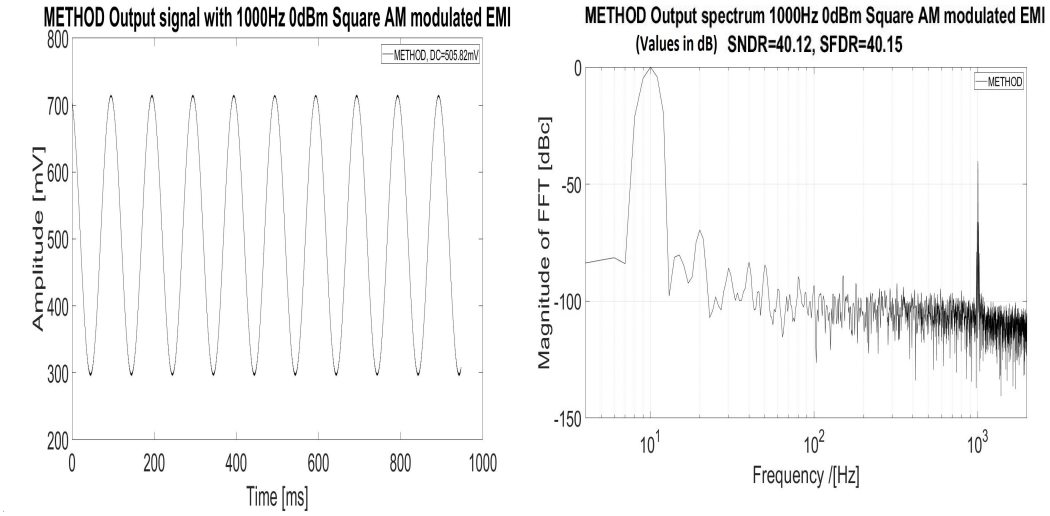


Figure 6.6: METHOD output and spectrum with 0dBm RF 1000Hz square wave modulated input

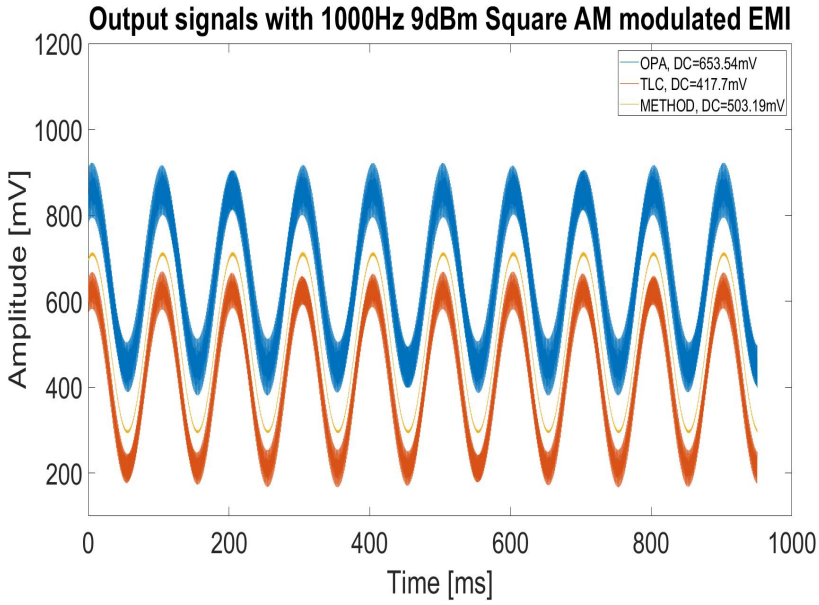


Figure 6.7: Outputs with 9dBm RF 1000Hz square wave modulated input

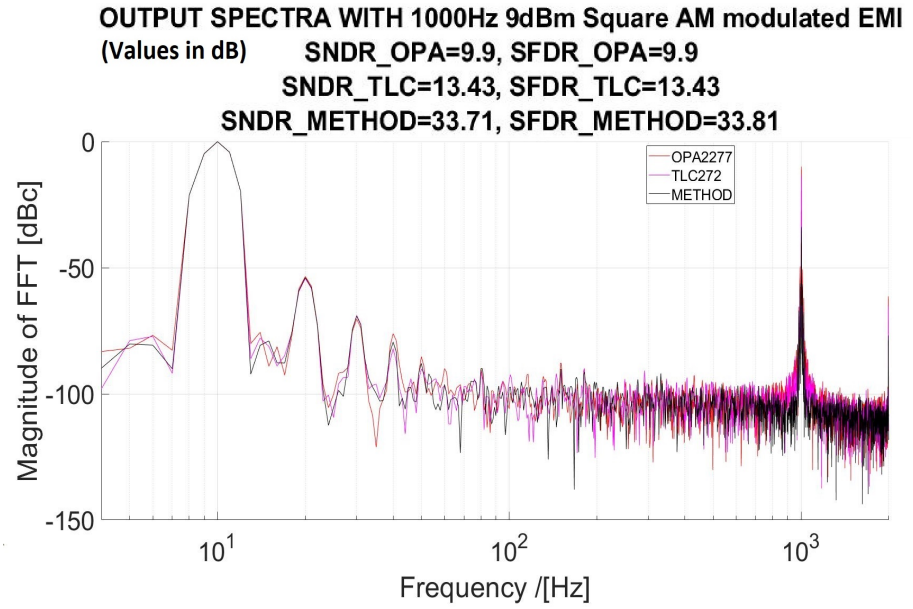


Figure 6.8: Output spectrum with 9dBm RF 1000Hz square wave modulated input

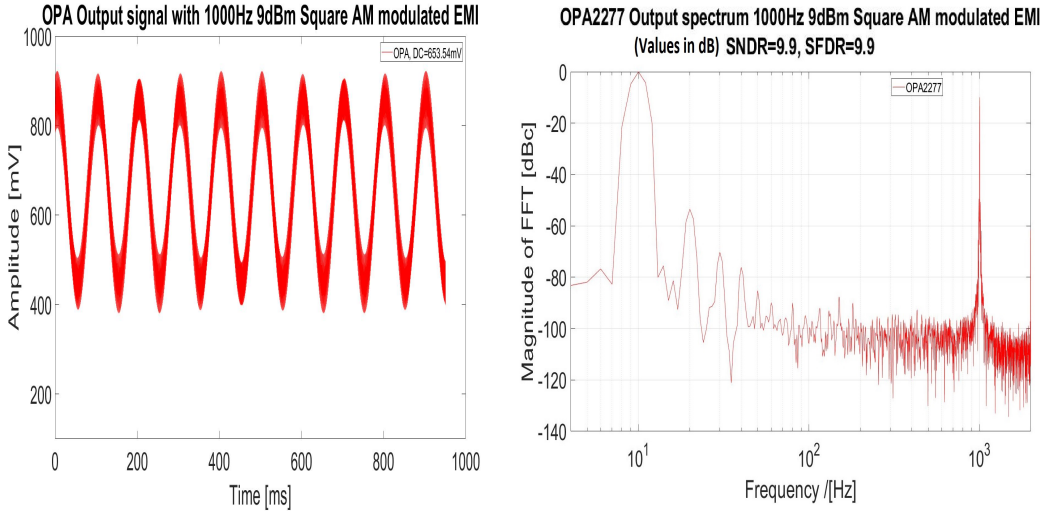


Figure 6.9: OPA output and spectrum with 9dBm RF 1000Hz square wave modulated input

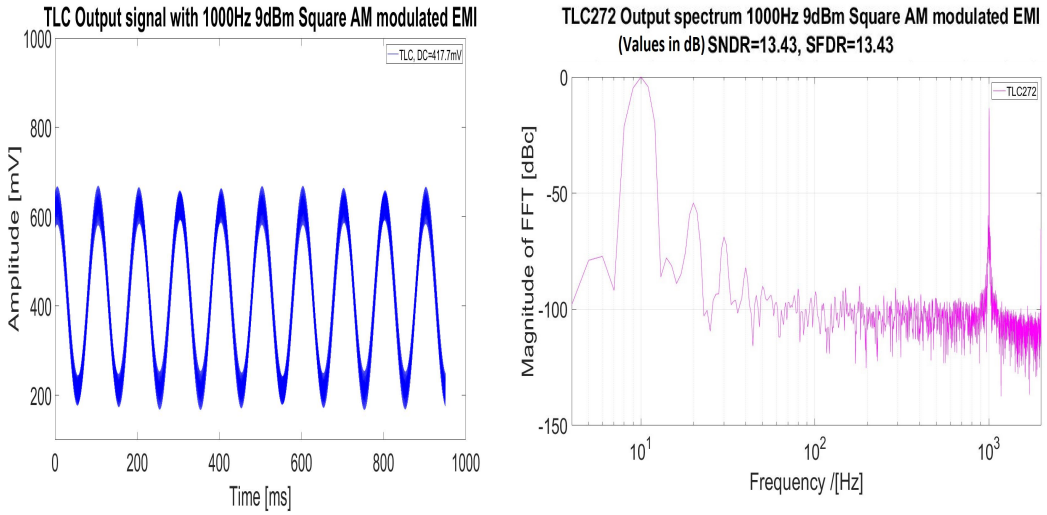


Figure 6.10: TLC output and spectrum with 9dBm RF 1000Hz square wave modulated input

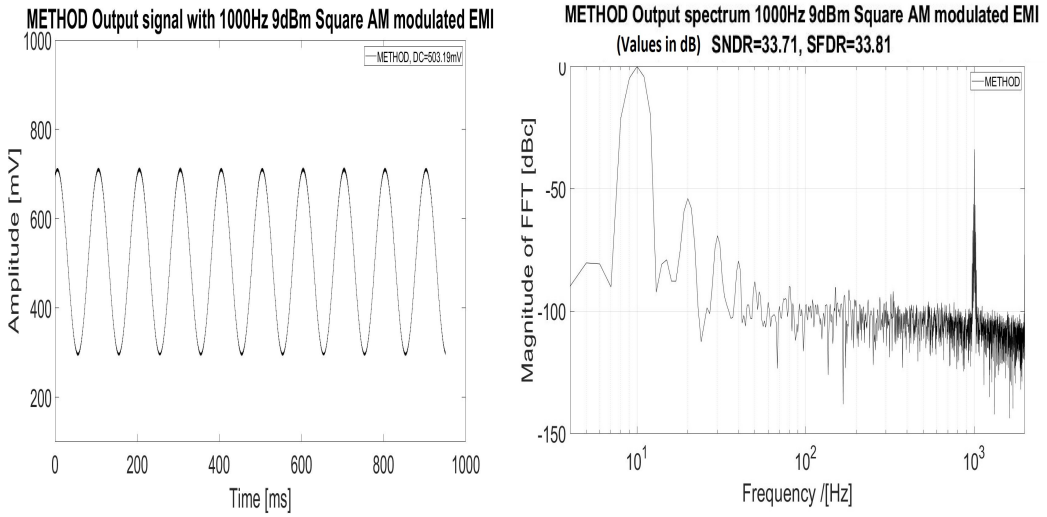


Figure 6.11: METHOD output and spectrum with 9dBm RF 100Hz square wave modulated input

6.1.2 RF 400Hz sine-wave modulated plots at 7dBm

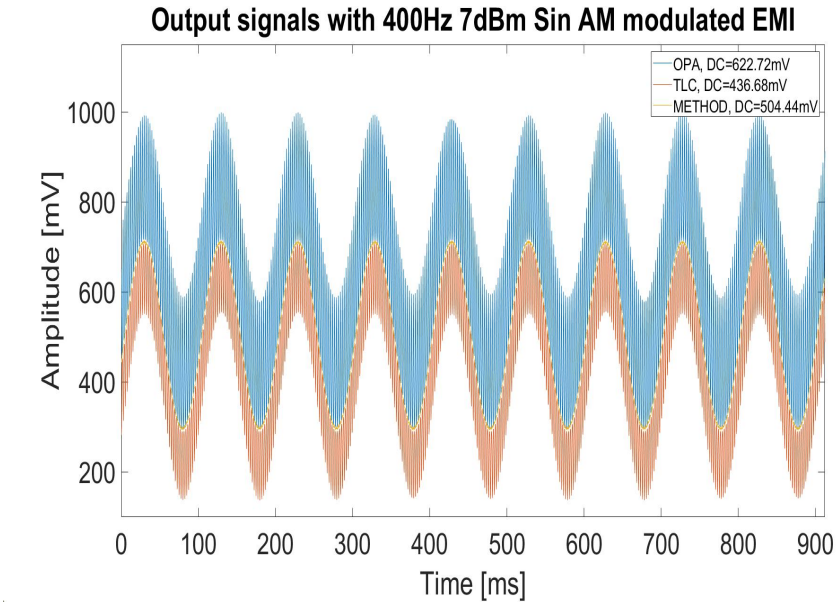


Figure 6.12: Outputs with 9dBm RF 400Hz sinusoidal wave modulated input

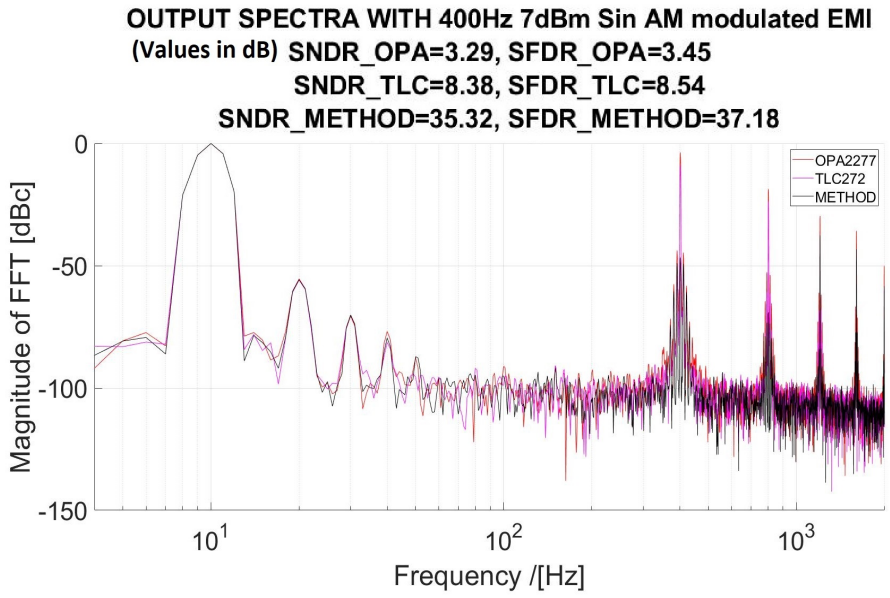


Figure 6.13: Output spectrum with 9dBm RF 400Hz sinusoidal wave modulated input

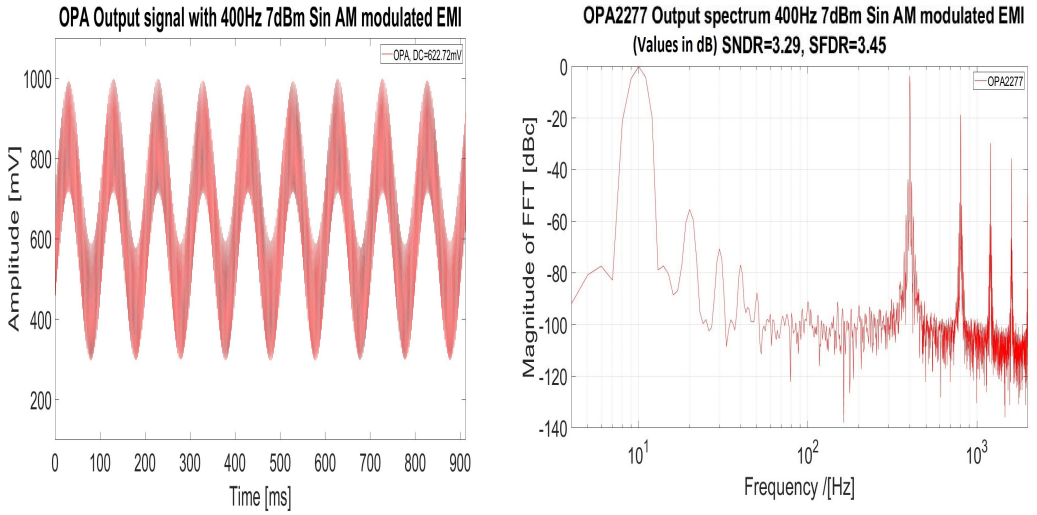


Figure 6.14: OPA output and spectrum with 9dBm RF 400Hz sinusoidal wave modulated input

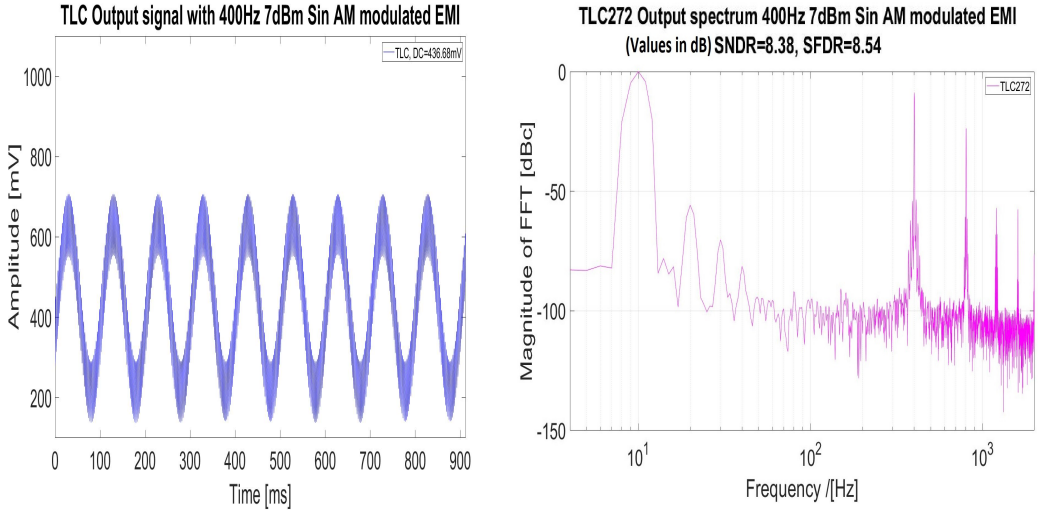


Figure 6.15: TLC output and spectrum with 9dBm RF 400Hz sinusoidal wave modulated input

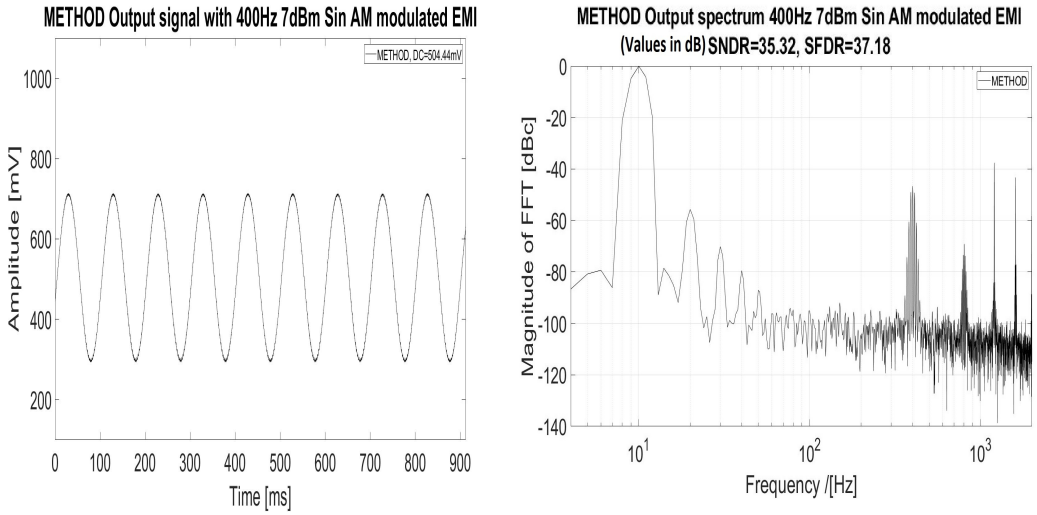


Figure 6.16: METHOD output and spectrum with 9dBm RF 400Hz sinusoidal wave modulated input

6.1.3 RF 100Hz exponential modulated plots at 9dBm

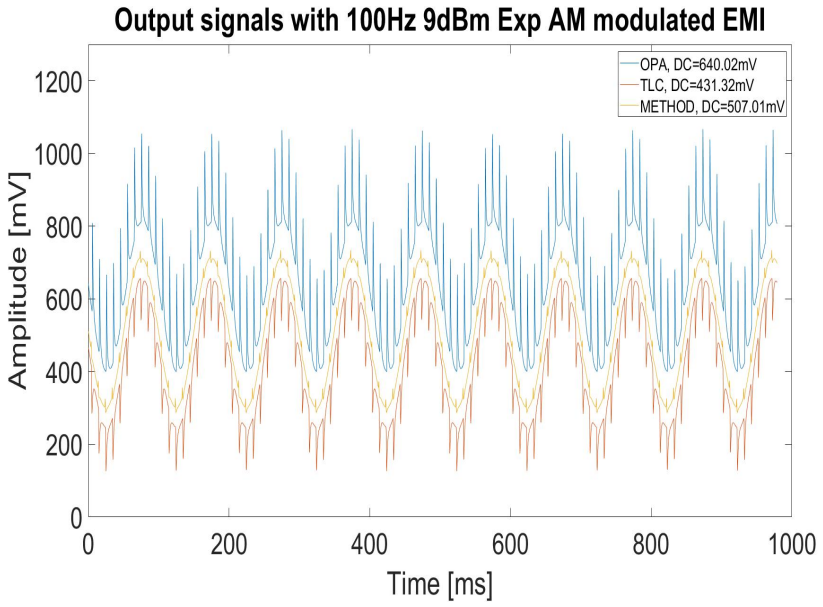


Figure 6.17: Outputs with 9dBm RF 100Hz exponential wave modulated input

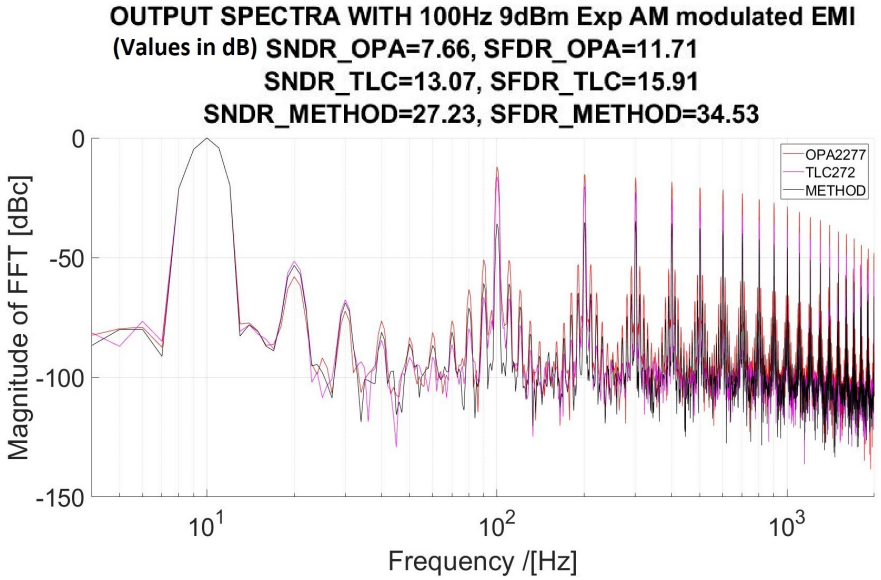


Figure 6.18: Output spectrum with 9dBm RF 100Hz exponential wave modulated input

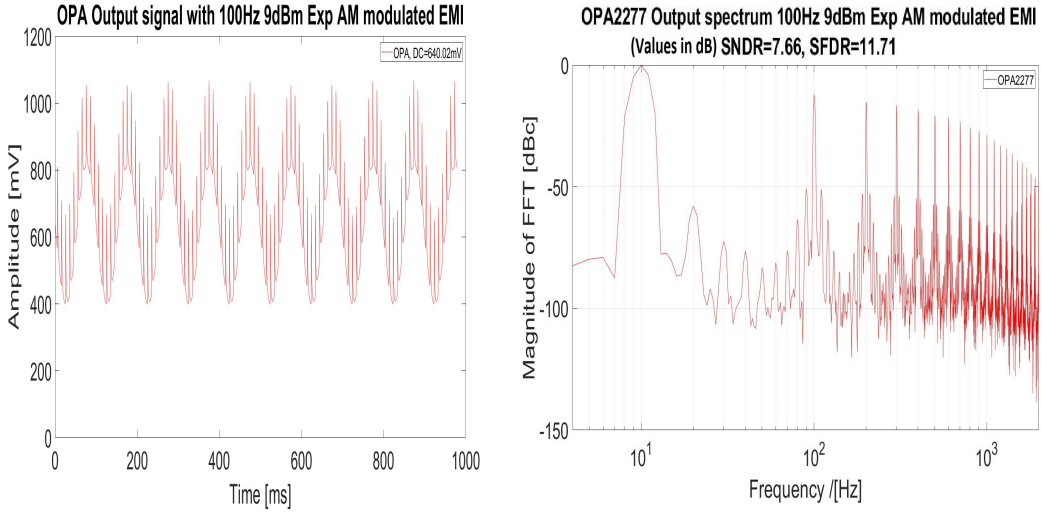


Figure 6.19: OPA output and spectrum with 9dBm RF 100Hz exponential wave modulated input

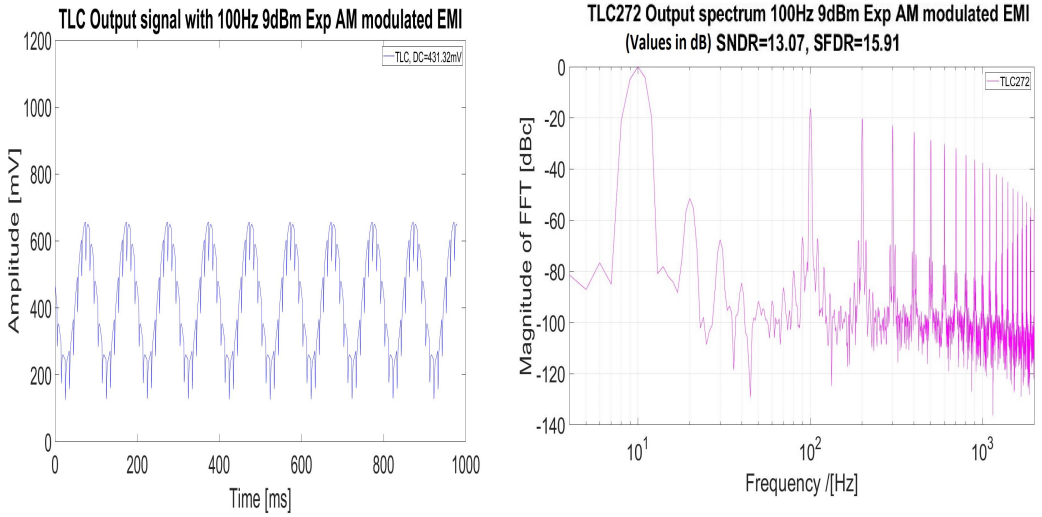


Figure 6.20: TLC output and spectrum with 9dBm RF 100Hz exponential wave modulated input

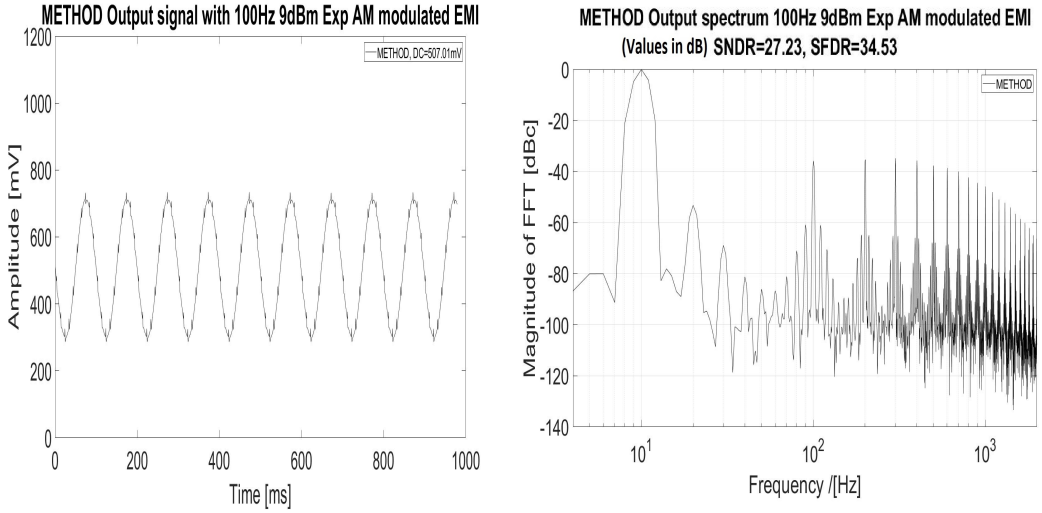


Figure 6.21: METHOD output and spectrum with 9dBm RF 100Hz exponential wave modulated input

6.1.4 RF 1kHz exponential modulated plots at 0-9dBm

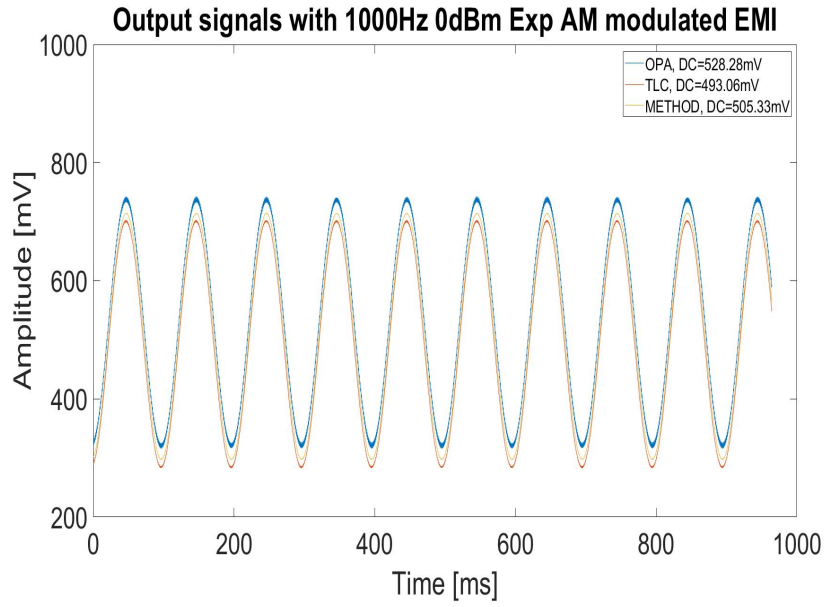


Figure 6.22: Outputs with 0dBm RF 1000hz exponential wave modulated input

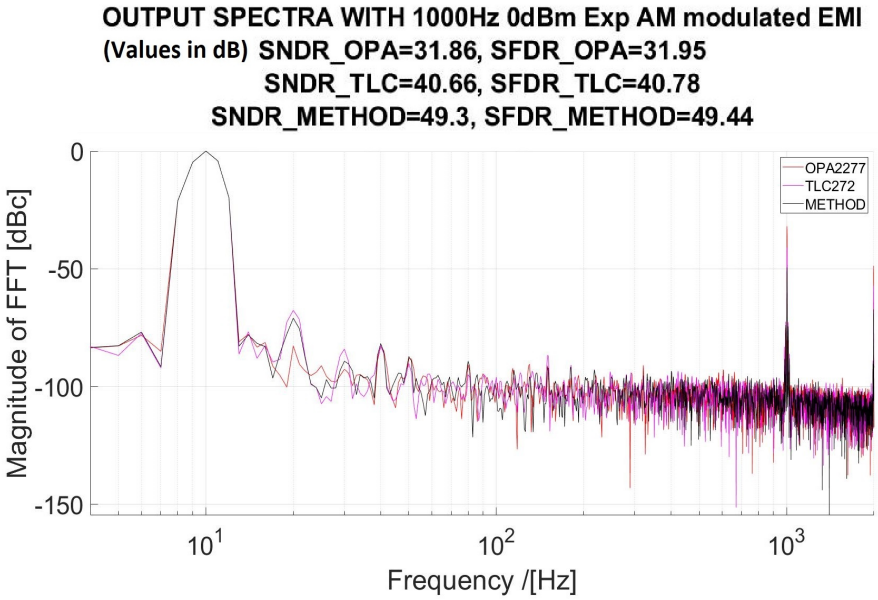


Figure 6.23: Output spectrum with 0dBm RF 1000Hz exponential wave modulated input

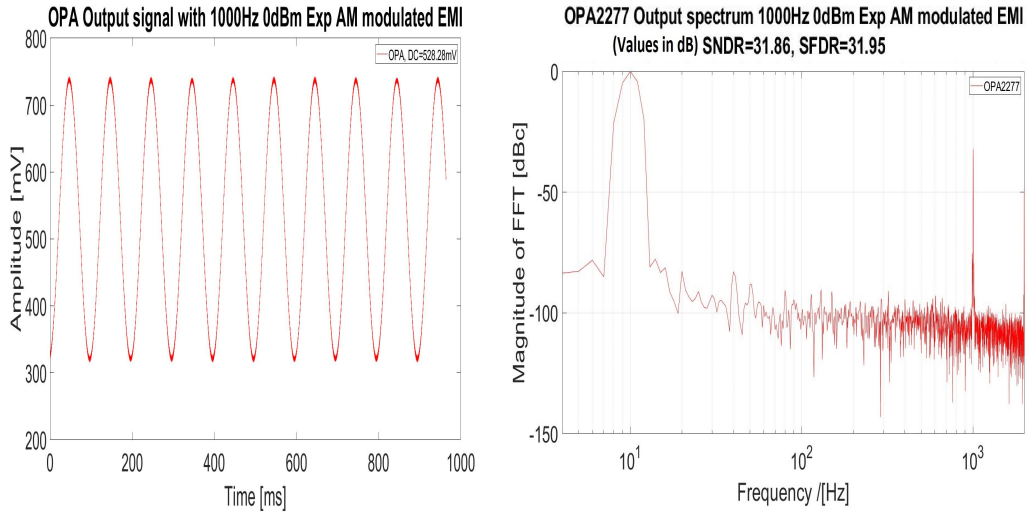


Figure 6.24: OPA output and spectrum with 0dBm RF 1000Hz exponential wave modulated input

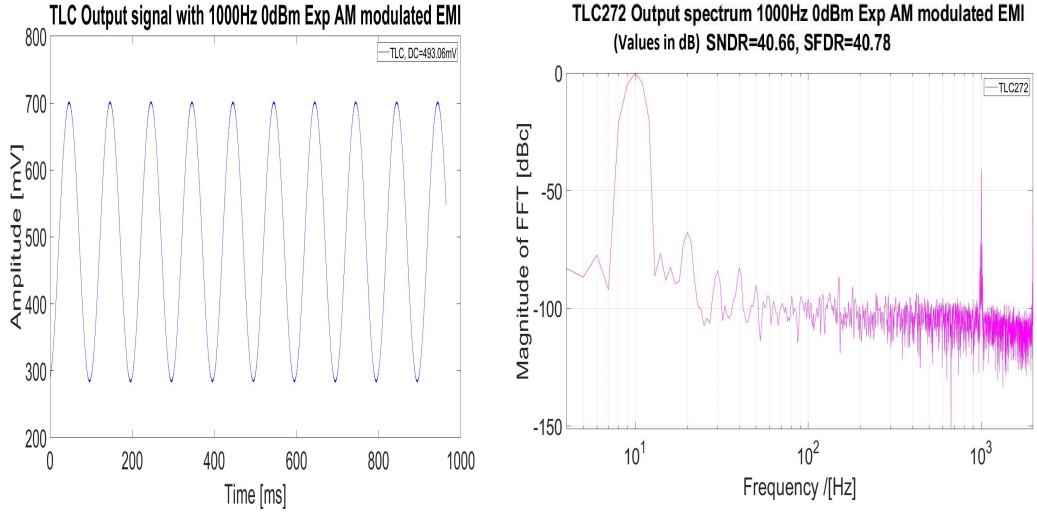


Figure 6.25: TLC output and spectrum with 0dBm RF 1000Hz exponential wave modulated input

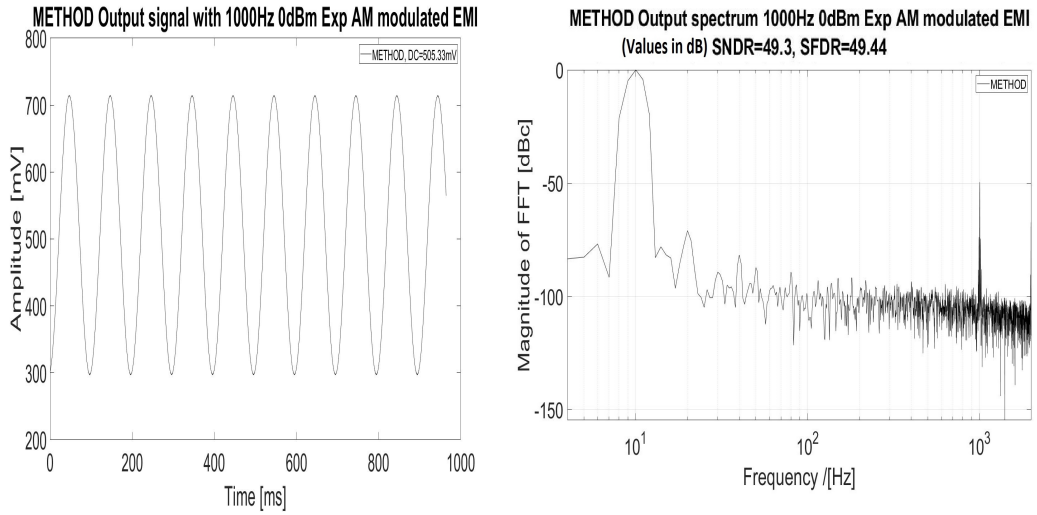


Figure 6.26: METHOD output and spectrum with 0dBm RF 1000Hz exponential wave modulated input

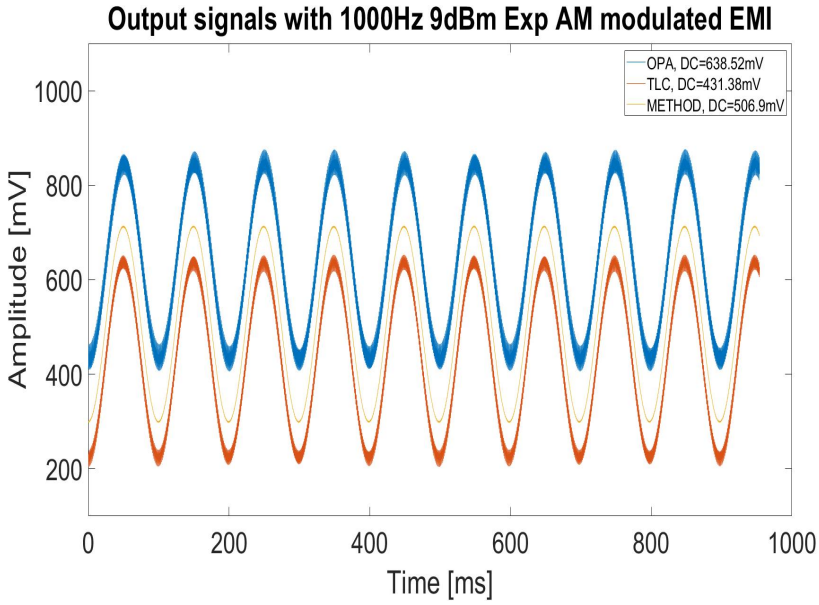


Figure 6.27: Outputs with 9dBm RF 1000Hz exponential wave modulated input

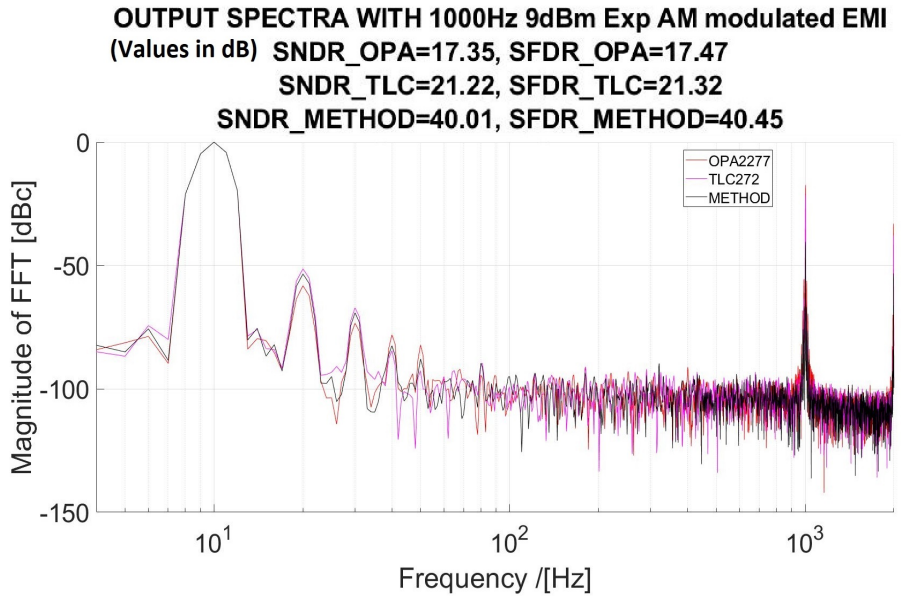


Figure 6.28: Output spectrum with 9dBm RF 1000Hz exponential wave modulated input

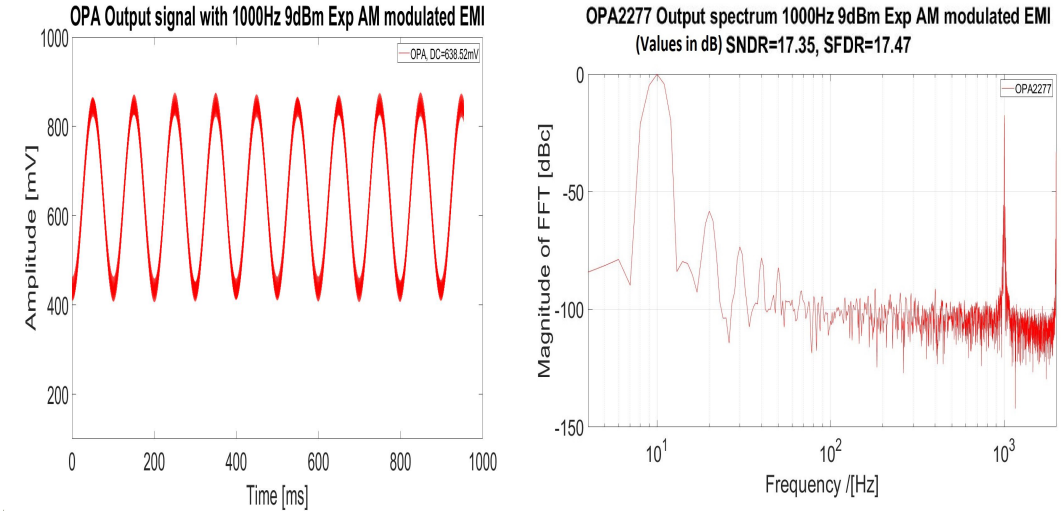


Figure 6.29: OPA output and spectrum with 9dBm RF 1000Hz exponential wave modulated input

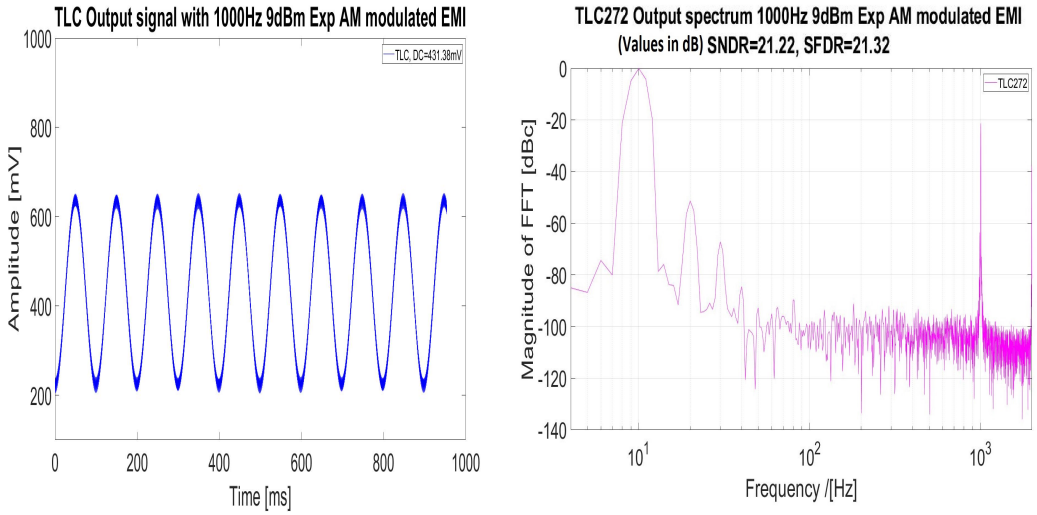


Figure 6.30: TLC output and spectrum with 9dBm RF 1000Hz exponential wave modulated input

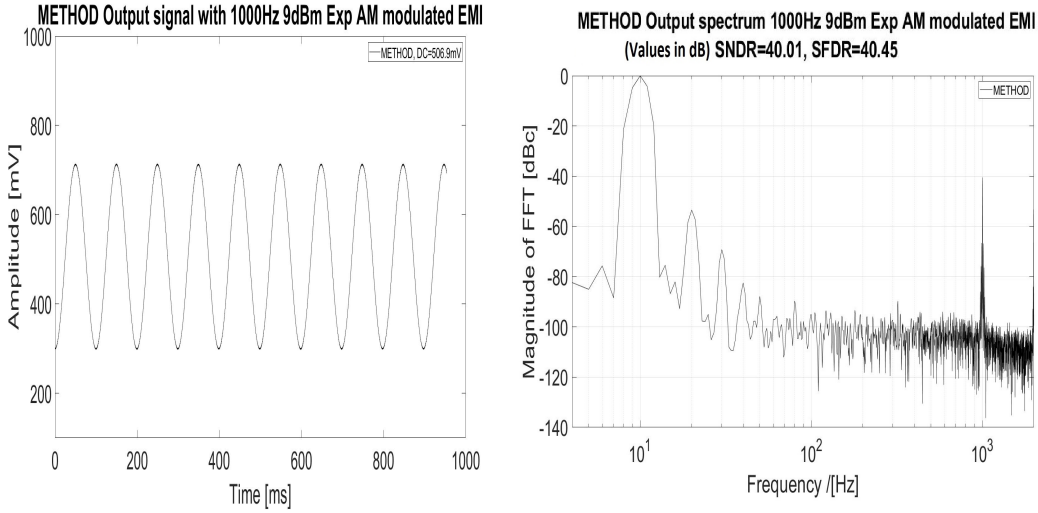


Figure 6.31: METHOD output and spectrum with 9dBm RF 100Hz exponential wave modulated input

6.2 SFDR and SNDR for a 1KHz square wave modulating signal

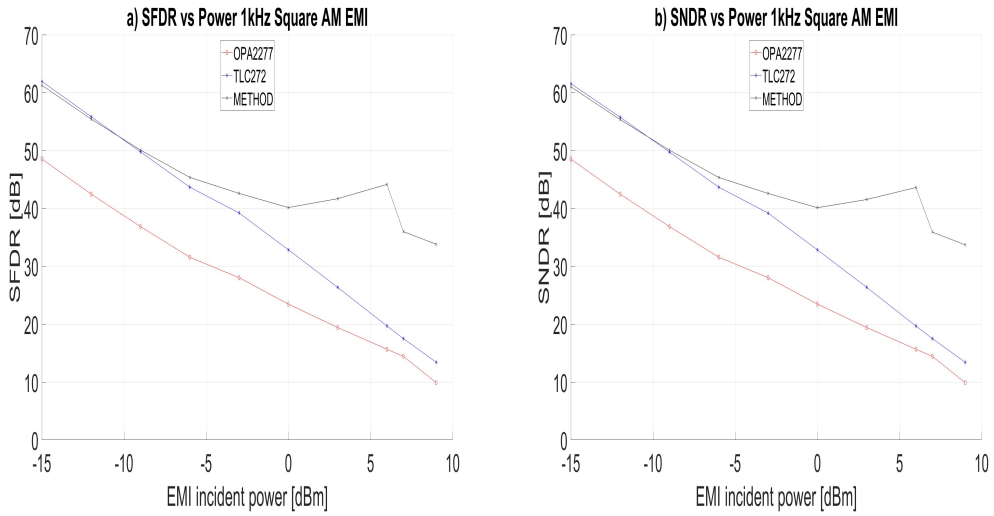


Figure 6.32: SFDR and SNDR vs power for injected 1kHz square wave modulated 100Mhz EMI

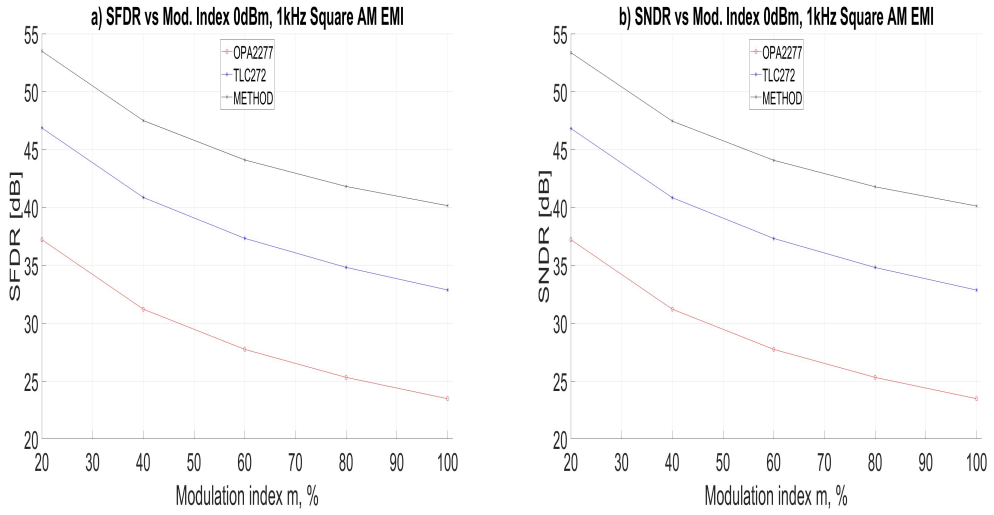


Figure 6.33: SFDR and SNDR vs modulation index for injected 1kHz square wave modulated 100Mhz EMI @0dBm

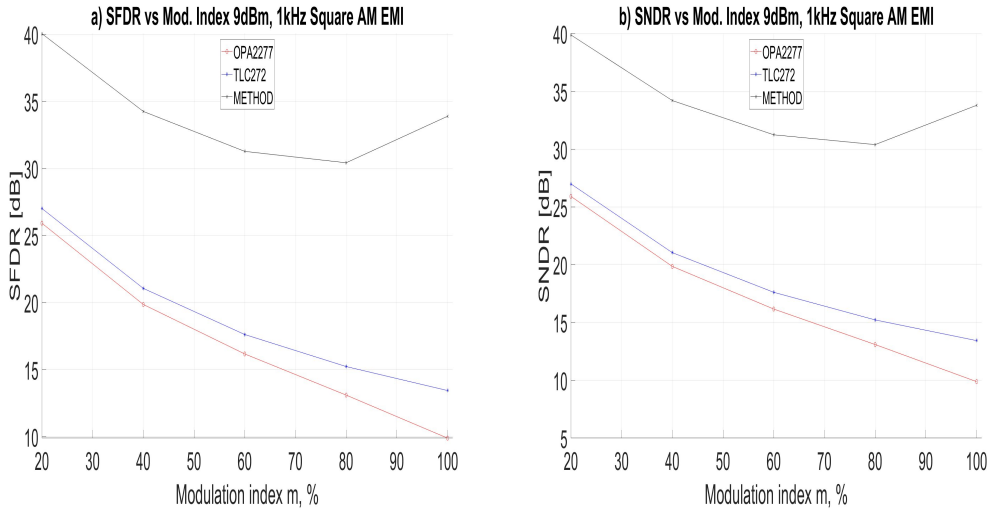


Figure 6.34: SFDR and SNDR vs modulation index for injected 1kHz square wave modulated 100Mhz EMI @9dBm

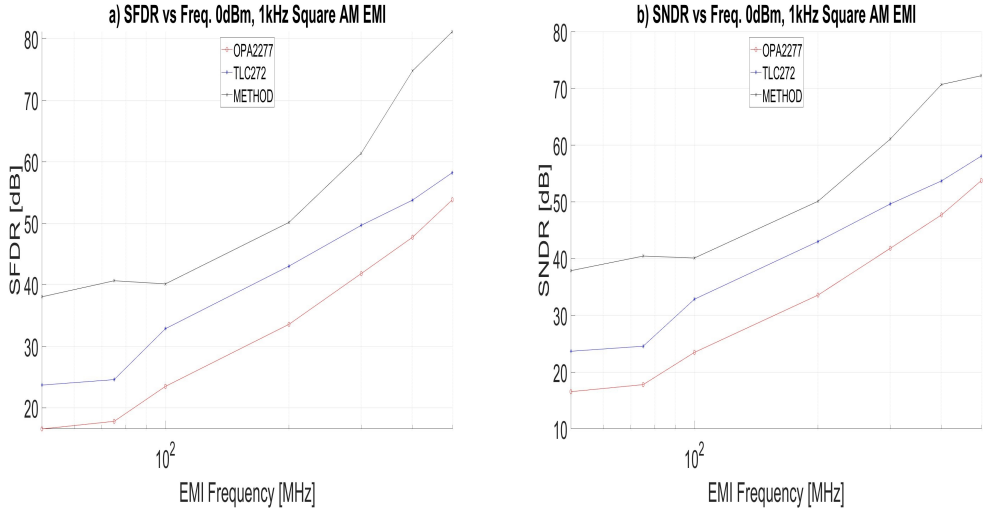


Figure 6.35: SFDR and SNDR vs frequency for injected 1kHz square wave modulated EMI @ 0dBm

6.3 The ADUC program

6.3.1 The standard program

```
// Bit Definitions
#define BIT0  0x01 //LSB
#define BIT1  0x02
#define BIT2  0x04
#define BIT3  0x08
#define BIT4  0x10
#define BIT5  0x20
#define BIT6  0x40
#define BIT7  0x80
#define BIT8  0x100
#define BIT9  0x200
#define BIT10 0x400
#define BIT11 0x800
#define BIT12 0x1000
#define BIT13 0x2000
#define BIT14 0x4000
#define BIT15 0x8000  //MSB

# include<aduc7060.h>
# include "stdio.h"
# include "string.h"
 #include <math.h>


void ADCInit(void);
//ADC initializing

void UARTInit(void);
//Uarti initializing

void StartSeq(void);
//send starting sequence 00 – FF – 00 – FF

unsigned long  Computedata(void);
//Computing data to be sent

volatile unsigned long DATAcomputed=0;
//result of the computation

volatile unsigned short dataComp=0;
//counter of data computed

volatile unsigned short datDAC0=0;
//counter 16 bit 0–65535
```

```
volatile unsigned short datDAC1=0;
//counter

volatile unsigned char newADCdata=0;
// Used to indicate that new ADC data is available 8bit=char  0—255

volatile unsigned char byteArrived = 0;
// byte arrive from uart

unsigned char szTemp[64] = "";
// Text to be sent at startup

volatile unsigned char ucRxChar = 0;
// Variable to read COMRX register into in IRQ handler

volatile int start = 0;


unsigned char ucTxBufferEmpty  = 0;
// Used to indicate that the UART Tx buffer is empty

volatile unsigned long ulADC0Result = 0x0;
// Variable that ADC0DAT is read into in ADC0 IRQ

volatile unsigned long ulADC1Result = 0x0;
const unsigned long long ALPHAfix =0x5CE91933;
//32 bit precision aplha 0.3629318

const unsigned long long NALFAfix =0xA316E6CC;
//32 bit

const unsigned long mask= 0xff000000;



int main(void)

{


  POWKEY1 = 0x1;
  // prekey enabling

  POWCON0 = 0x78;
  // 78 clock to 10.24MHZ – 38 ext xtal off Set core to max CPU speed of 10.24Mhz

  POWKEY2 = 0xF4;
  // postkey enabling
```

```
//PLLKEY1=0xAA;
//set internal oscillator

//PLLCON = 0x00;
//PLLKEY2=0x55;

POWKEY3=0x76;
//power on uart only – switch off ic2/spi ad Pwm
POWCON1=BIT5;
POWKEY4=0xB1;

UARTInit();
// Initialize the UART for 115200–8–N

ADCInit();
// init Adc0–1

IRQEN = BIT11;
// Enable UART bit11 to not consider adc inq until start

StartSeq();


while (1)
{
  if (newADCdata == 1)                              // ADC Data ready to be computed
    and sent
   {  unsigned char *pointer = (unsigned char*)&DATAcomputed;
  //+ 3;  //&DATAcomputed points to MSByte – pointer to data byte to be sent



    DATAcomputed = Computedata();                   //compute data to be sent
    DATAcomputed = (DATAcomputed & ~mask) | ((datDAC0<<24 & mask));
    do {                                            //send byte by byte to uart
      COMTX = *pointer;
      ucTxBufferEmpty = 0;
      //set if uart buffer empty

        while (ucTxBufferEmpty == 0)
      //wait byte to be sent
          {
          }
      ucTxBufferEmpty = 0;
      pointer++;  //--                  //nextbyte

    } while (pointer != (unsigned char*)&DATAcomputed + 4);
      //+1  //send 3 bytes     value+3 to value+1
```

```
      newADCdata = 0;

      }

    //stop procedure
    if (byteArrived==1)
    //((COMSTA0 & 0x01) == 0x01)
    // If the Rx buffer full bit is set
      {

      if (ucRxChar == 22)
    //text received is  $
      { IRQEN = BIT11;
        // Enable UART bit11 to not consider adc inq until start

        newADCdata = 0;
        dataComp=0;
        //counter of data computed

        datDAC0=0;
        //counter 16 bit 0--65535

        datDAC1=0;
        start=0;
          StartSeq();
        //IRQEN = BIT10 + BIT11;

        byteArrived=0;
      }
    byteArrived=0;
    }

  }
}




 void IRQ_Handler(void) __irq
{
  unsigned long IRQSTATUS = 0;
  // was volatile

  unsigned char ucCOMIID0 = 0;
  // 8bit reg tells what int arrives from uart 010=0x2 buffer tx empty
                                  // cleared by reading or writing on comtx

  IRQSTATUS = IRQSTA;
```

```
// Read off IRQSTA register   BIT11=1 for uart int

if ((IRQSTATUS & BIT11) == BIT11)
//UART interrupt source

{
  ucCOMIID0 = COMIID0;
  if ((ucCOMIID0 & 0x2) == 0x2)
  // Transmit buffer empty

  {
     ucTxBufferEmpty = 1;
  }
  if ((ucCOMIID0 & 0x4) == 0x4)
  // Receive byte

  {
    ucRxChar = COMRX;
    byteArrived = 1;
  }
}

if ((IRQSTATUS & BIT10) == BIT10)
//If ADC0 or 1 interrupt source   BIT10=1 for adc

{

  ulADC1Result = ADC1DAT;
  //reading adc0dat clear ready bit of both ADCx

  //datDAC0++;
  //ADCX is 32 bit but from sigma only lsb0-23 bit.

  ulADC0Result = ADC0DAT;
  //first read adc1dat than adc0dat to clear and have synch read acdxready cleared reading
    those

  if (datDAC0 ==255)
  { datDAC0 = 1;
  }
  else
  { datDAC0++;
  }
  if (datDAC1 ==255)
  { datDAC1 = 1;
  }
  else
  { datDAC1++;
  }
```

```
    //datDAC1++;
    newADCdata = 1;

  }
}


unsigned long Computedata()
{
    unsigned long long Compute_Fixed =0;
    //fixed point computation 64bit

    unsigned long long Adc0_Fixed = 0;
    unsigned long long Adc1_Fixed = 0;

    //fixed point computation

    Adc0_Fixed =  (unsigned long long)ulADC0Result;
    //<<Dim   //((sizeof(unsigned long long)*8)/2);

    Adc1_Fixed =  (unsigned long long) ulADC1Result;
    // << dim;    //((sizeof(unsigned long long)*8)/2);

    Compute_Fixed = (ALFAfix*Adc0_Fixed)+ (NALFAfix*Adc1_Fixed);

    Compute_Fixed = Compute_Fixed + 0x80000000;
    //rounding

    Compute_Fixed = Compute_Fixed >> 32 ;
    //Back to long truncating fractional part


    if (dataComp==255)
    { dataComp = 1;
    }
    else
    { dataComp++;
    }
    //increase the compute data counter


    return (unsigned long) Compute_Fixed;


}


void ADCInit()
{
```

```
ADCMSKI =  BIT0;
//bit0 = adc0 BIT1=adc1 used only one irq irqadc0 Enable ADC0 result ready interrupt
    source
  //--ADCFLT = 0xFF1F; // Chop on, Averaging, AF=63, SF=31, 4Hz /SET IN IDLE
    MODE ADCMDE DEFAULT IDLE            // Chop on, Averaging, AF=63, SF=31, 4
    Hz  /SET IN IDLE MODE ADCMDE DEFAULT IDLE

ADCFLT = 0x1;
//run @8000Ks if 0x0
// CHOP OFF - NO AVARAGE- SF=bit0-6 ->4000HZ if bit0-6=0x1 / 2666Hz if bit0
    -6=0x2 (512000/((sf+1)*64)

ADCCFG = 0;
// ENABLE EVENTUALLY THE COUNTER BIT0

GP0KEY1 = 0x7;
//Write to GP0KEY1

GP0CON1 = BIT0;
//Select SPI functionality for P0.0 to P0.3

GP0KEY2 = 0x13;

//ADC0CON = 0x8145;
// For system calibration set the gain that will be used
                            // for measurements to ensure the best calibration is achieved,
                            // In this case gain is 32 therefore Full Scale is 0.0375v
//SystemZeroCalibration();
// For best results use a system zero scale calibration

//SystemFullCalibration();
// and a system full calibration instead of the self calibrations
                            // AIN -ne has to be biased using the DAC
ADC0CON =BIT7 + BIT8 + BIT10 + BIT15;
// Bit7+Bit8= Adc2/5 // Gain = 1, BIT 10 Unipolar (not complement2) OUT, BIT 15
    enable ADC0, Int Reference STD
                                                // BIT7-8 PIN_ADC2/ADC5 UNIPOLAR, buf
    negative bypassed std
ADC1CON = BIT2 + BIT7 + BIT8 + BIT9 + BIT11 + BIT15;
// G=1/ Bit2=NEGATIVE BUF BYPASS (bit2+3=full bypass)/ bit7-8-9=PIN_ADC6/
    ADC5 s.ended/ bit11=UNIPOLAR OUT/ bit15=ADC1 ENABLE,int ref,

//ADCMDE  = 0x94;
// Offest Self Calibration

  //while((ADCSTA & BIT15) == BIT15){}
// Wait for Calibration routine to complete

//ADCMDE  = 0x95;
```

```
  // Gain Self Calibration

  //while((ADCSTA & BIT15) == BIT15){}
  // Wait for Calibration routine to complete

  ADCMDE  = 0x81;
  // Enable Continuous conversion mode and 512khz  = bit0 + bit7

  //ADCCFG = 0;
}


void UARTInit()
{

  GP1CON = BIT0 + BIT4;
  // Select BIt0–4 UART functionality for P1.0/P1.1

  COMCON0 = BIT7;
  // Enable access to COMDIV registers bit7=0x80

  COMDIV0 = 0x1;
  //0x21= 9600;   //0x3=115200;
  // 0x1=320000 0x11=19200 0x3=115200 NOT PRECISE!! Set baud rate to 115200

  COMDIV1 = 0x00;
  //0x1+the comdiv2 set to 230400

  COMDIV2 = 0x31c | BIT11 | BIT15;
  // 230400 Enable fractional divider for more accurate baud rate setting

  COMCON0 = BIT0 + BIT1;
  //8 data bits 1 stop bit

  COMIEN0 = BIT1 + BIT0;
  // Enable UART interrupts when Rx full (BIT0) and Tx buffer empty (BIT1).
}

void StartSeq()
{
   unsigned long Dummy = 0;
  //dummy variable to wait for some seconds from power up

   unsigned int nLen = 0;
   unsigned char i = 0;

   for (Dummy = 0; Dummy <= 1000000; Dummy++);
```

```
//sending starting string to pc
sprintf ( (char*)szTemp, "XXXXXXXXXXXX-Waiting for start from PC-
    XXXXXXXXXXXXX \r\n");
// Send Opening String to the UART

nLen = strlen((char*)szTemp);
for ( i = 0 ; i < nLen ; i++ )
// loop to send opening String

{
  COMTX = szTemp[i];
  ucTxBufferEmpty = 0;
    while (ucTxBufferEmpty == 0) //(COMSTA0 & 0x40) == 0x00)
  // Wait for Tx buffer empty bit to be set
    {
    }
  ucTxBufferEmpty = 0;
}
//receiving start bit task

  while (start == 0)
  {
   if (byteArrived==1)//((COMSTA0 & 0x01) == 0x01)
  // If the Rx buffer full bit is set
   {
    //ucRxChar = COMRX;                          // read COMRX register
   if (ucRxChar != 33)                          //text received is not !
   {
    sprintf ( (char*)szTemp, "Received Charachter was : %d,\r\n\n",ucRxChar );
    // Send the Received charachter to the UART

    nLen = strlen((char*)szTemp);
      for ( i = 0 ; i < nLen ; i++ )
   // loop to send String
      {
        COMTX = szTemp[i];
    // Load Tx buffer
      ucTxBufferEmpty = 0;
        while (ucTxBufferEmpty == 0)
      //wait byte to be sent
          {
          }
      ucTxBufferEmpty = 0;
    }
   }
   else
   {

    start=1;
```

```
    }


    byteArrived=0;
   }
  }
  IRQEN = BIT10 + BIT11;  // Enable UART bit11 – ADC0+1 interrupts source bit10

}
```

6.3.2 The measurements program

```
//Measurement program

// Bit Definitions
#define BIT0  0x01 //LSB
#define BIT1  0x02
#define BIT2  0x04
#define BIT3  0x08
#define BIT4  0x10
#define BIT5  0x20
#define BIT6  0x40
#define BIT7  0x80
#define BIT8  0x100
#define BIT9  0x200
#define BIT10 0x400
#define BIT11 0x800
#define BIT12 0x1000
#define BIT13 0x2000
#define BIT14 0x4000
#define BIT15 0x8000  //MSB

# include<aduc7060.h>
# include "stdio.h"
# include "string.h"
 #include <math.h>


void ADCInit(void);
//ADC initializing

void UARTInit(void);
//Uarti initializing

void StartSeq(void);
//send starting sequence 00 – FF – 00 – FF
```

```
unsigned long  Computedata(void);
//Computing data to be sent

volatile unsigned long DATAcomputed=0;
//result of the computation

volatile unsigned short dataComp=0;
//counter of data computed

volatile unsigned short datDAC0=0;
//counter 16 bit 0--65535

volatile unsigned short datDAC1=0;
//counter

volatile unsigned char newADCdata=0;
// Used to indicate that new ADC data is available 8bit=char  0--255

volatile unsigned char byteArrived = 0;
// byte arrive from uart

unsigned char szTemp[64] = "";
// Text to be sent at startup

volatile unsigned char ucRxChar = 0;
// Variable to read COMRX register into in IRQ handler

volatile int start = 0;

volatile int datatosend=0;
//variable to send ch1 ch2 tecnique

unsigned char ucTxBufferEmpty  = 0;
// Used to indicate that the UART Tx buffer is empty

volatile unsigned long ulADC0Result = 0x0;
// Variable that ADC0DAT is read into in ADC0 IRQ

volatile unsigned long ulADC1Result = 0x0;
const unsigned long long ALPHAfix =0x5CE91933;
//32 bit precision aplha 0.3629318

const unsigned long long NALFAfix =0xA316E6CC;
//32 bit

const unsigned long mask= 0xff000000;
```

```
int main(void)

{


  POWKEY1 = 0x1;
  // prekey enabling

  POWCON0 = 0x78;
  // 78 clock to 10.24MHZ - 38 ext xtal off Set core to max CPU speed of 10.24Mhz

  POWKEY2 = 0xF4;
  // postkey enabling

  //PLLKEY1=0xAA;
  //set internal oscillator

  //PLLCON = 0x00;
  //PLLKEY2=0x55;

  POWKEY3=0x76;
  //power on uart only - switch off ic2/spi ad Pwm
  POWCON1=BIT5;
  POWKEY4=0xB1;

  UARTInit();
  // Initialize the UART for 115200-8-N

  ADCInit();
  // init Adc0-1

  IRQEN = BIT11;
  // Enable UART bit11 to not consider adc inq until start

  StartSeq();

  ADCMDE  = 0x81;
  //bit0+bit7 activate the adc continous conversion mode

while (1)
  {
    if (newADCdata == 1)                                  // ADC Data ready to be computed
      and sent
     {  unsigned char *pointer = (unsigned char*)&DATAcomputed;
    //+ 3;  //&DATAcomputed points to MSByte - pointer to data byte to be sent
```

```
switch (datatosend) //Evaluate what channel to be sent
{
case 0: //send the first channel
  DATAcomputed = ulADC0Result;
  break;
case 1:
  DATAcomputed = ulADC1Result;
  break;
case 2:
  DATAcomputed = Computedata(); //compute data to be sent
  break;
}


DATAcomputed = (DATAcomputed & ~mask) | ((datDAC0<<24 & mask));
do {                          //send byte by byte to uart
  COMTX = *pointer;
  ucTxBufferEmpty = 0;
  //set if uart buffer empty

    while (ucTxBufferEmpty == 0)
  //wait byte to be sent
      {
      }
  ucTxBufferEmpty = 0;
  pointer++;  //--                    //nextbyte

} while (pointer != (unsigned char*)&DATAcomputed + 4);
  //+1  //send 3 bytes     value+3 to value+1

newADCdata = 0;

}

//stop procedure
if (byteArrived==1)
//((COMSTA0 & 0x01) == 0x01)
// If the Rx buffer full bit is set
 {

 if (ucRxChar == 22)
//text received is  $
 { IRQEN = BIT11;
  // Enable UART bit11 to not consider adc inq until start

  datatosend++; //set the channel to be sent
  if (datatosend ==3)
  { datatosend=0;
  }
```

```
        newADCdata = 0;
        dataComp=0;
        //counter of data computed

        datDAC0=0;
        //counter 16 bit 0--65535

        datDAC1=0;
        start=0;
          StartSeq();
        //IRQEN = BIT10 + BIT11;

        byteArrived=0;
      }
    byteArrived=0;
    }

  }
}




 void IRQ_Handler(void) __irq
{
  unsigned long IRQSTATUS = 0;
  // was volatile

  unsigned char ucCOMIID0 = 0;
  // 8bit reg tells what int arrives from uart 010=0x2 buffer tx empty
                                  // cleared by reading or writing on comtx

  IRQSTATUS = IRQSTA;
  // Read off IRQSTA register   BIT11=1 for uart int

  if ((IRQSTATUS & BIT11) == BIT11)
  //UART interrupt source

  {
    ucCOMIID0 = COMIID0;
    if ((ucCOMIID0 & 0x2) == 0x2)
    // Transmit buffer empty

    {
        ucTxBufferEmpty = 1;
    }
    if ((ucCOMIID0 & 0x4) == 0x4)
    // Receive byte
```

```
    {
      ucRxChar = COMRX;
      byteArrived = 1;
    }
  }

  if ((IRQSTATUS & BIT10) == BIT10)
  //If ADC0 or 1 interrupt source   BIT10=1 for adc

  {

    ulADC1Result = ADC1DAT;
    //reading adc0dat clear ready bit of both ADCx

    //datDAC0++;
    //ADCX is 32 bit but from sigma only lsb0−23 bit.

    ulADC0Result = ADC0DAT;
    //first read adc1dat than adc0dat to clear and have synch read acdxready cleared reading
      those

    if (datDAC0 ==255)
    { datDAC0 = 1;
    }
    else
    { datDAC0++;
    }
    if (datDAC1 ==255)
    { datDAC1 = 1;
    }
    else
    { datDAC1++;
    }
    //datDAC1++;
    newADCdata = 1;

  }
}


unsigned long Computedata()
{
    unsigned long long Compute_Fixed =0;
    //fixed point computation 64bit

    unsigned long long Adc0_Fixed = 0;
    unsigned long long Adc1_Fixed = 0;
```

```
    //fixed point computation

    Adc0_Fixed =  (unsigned long long)ulADC0Result;
    //<<Dim   ///((sizeof(unsigned long long)*8)/2);

    Adc1_Fixed =  (unsigned long long) ulADC1Result;
    // << dim;    ///((sizeof(unsigned long long)*8)/2);

    Compute_Fixed = (ALFAfix*Adc0_Fixed)+ (NALFAfix*Adc1_Fixed);

    Compute_Fixed = Compute_Fixed + 0x80000000;
    //rounding

    Compute_Fixed = Compute_Fixed >> 32 ;
    //Back to long truncating fractional part


    if (dataComp==255)
    { dataComp = 1;
    }
    else
    { dataComp++;
    }
    //increase the compute data counter


    return (unsigned long) Compute_Fixed;


}


void ADCInit()
{
  ADCMSKI =  BIT0;
  //bit0 = adc0 BIT1=adc1 used only one irq irqadc0 Enable ADC0 result ready interrupt
     source
  //--ADCFLT = 0xFF1F;// Chop on, Averaging, AF=63, SF=31, 4Hz  /SET IN IDLE
     MODE ADCMDE DEFAULT IDLE

  ADCFLT = 0x1;
  //run @8000Ks if 0x0
  // CHOP OFF - NO AVARAGE- SF=bit0-6 ->4000HZ if bit0-6=0x1 / 2666Hz if bit0
     -6=0x2 (512000/((sf+1)*64)

  ADCCFG = 0;
  // ENABLE EVENTUALLY THE COUNTER BIT0

  GP0KEY1 = 0x7;
```

```
  //Write to GP0KEY1

  GP0CON1 = BIT0;
  //Select SPI functionality for P0.0 to P0.3

  GP0KEY2 = 0x13;

  //ADC0CON = 0x8145;
  // For system calibration set the gain that will be used
  // for measurements to ensure the best calibration is achieved,
    // In this case gain is 32 therefore Full Scale is 0.0375v
  //SystemZeroCalibration();
  // For best results use a system zero scale calibration

  //SystemFullCalibration();
  // and a system full calibration instead of the self calibrations
                                  // AIN −ne has to be biased using the DAC
  ADC0CON =BIT7 + BIT8 + BIT10 + BIT15;
  // Bit7+Bit8= Adc2/5 // Gain = 1, BIT 10 Unipolar (not complement2) OUT, BIT 15
     enable ADC0, Int Reference STD
  // BIT7−8 PIN_ADC2/ADC5 UNIPOLAR, buf negative bypassed std
  ADC1CON = BIT2 + BIT7 + BIT8 + BIT9 + BIT11 + BIT15;
  // G=1/ Bit2=NEGATIVE BUF BYPASS (bit2+3=full bypass)/ bit7−8−9=PIN_ADC6/
     ADC5 s.ended/ bit11=UNIPOLAR OUT/ bit15=ADC1 ENABLE,int ref,

  //ADCMDE  = 0x94;
  // Offest Self Calibration

    //while((ADCSTA & BIT15) == BIT15){}
  // Wait for Calibration routine to complete

  //ADCMDE  = 0x95;
  // Gain Self Calibration

  //while((ADCSTA & BIT15) == BIT15){}
  // Wait for Calibration routine to complete

  //ADCMDE  = 0x81;
  // Enable Continuous conversion mode and 512khz  = bit0 + bit7

  //ADCCFG = 0;
}


void UARTInit()
{

  GP1CON = BIT0 + BIT4;
  // Select BIt0−4 UART functionality for P1.0/P1.1
```

```
  COMCON0 = BIT7;
  // Enable access to COMDIV registers bit7=0x80

  COMDIV0 = 0x1;
  //0x21= 9600;   //0x3=115200;
  // 0x1=320000 0x11=19200 0x3=115200 NOT PRECISE!! Set baud rate to 115200

  COMDIV1 = 0x00;
  //0x1+the comdiv2 set to 230400

  COMDIV2 = 0x31c | BIT11 | BIT15;
  // 230400 Enable fractional divider for more accurate baud rate setting

  COMCON0 = BIT0 + BIT1;
  //8 data bits 1 stop bit

  COMIEN0 = BIT1 + BIT0;
  // Enable UART interrupts when Rx full (BIT0) and Tx buffer empty (BIT1).
}

void StartSeq()
{
   unsigned long Dummy = 0;
  //dummy variable to wait for some seconds from power up

   unsigned int nLen = 0;
   unsigned char i = 0;

   for (Dummy = 0; Dummy <= 1000000; Dummy++);


  //sending starting string to pc
  sprintf ( (char*)szTemp, "XXXXXXXXXXXX-Waiting_for_start_from_PC-
      XXXXXXXXXXXXX_\r\n");
  // Send Opening String to the UART

  nLen = strlen((char*)szTemp);
  for ( i = 0 ; i < nLen ; i++ )
  // loop to send opening String

  {
     COMTX = szTemp[i];
     ucTxBufferEmpty = 0;
       while (ucTxBufferEmpty == 0) //(COMSTA0 & 0x40) == 0x00)
     // Wait for Tx buffer empty bit to be set
       {
       }
     ucTxBufferEmpty = 0;
```

```
  }
  //receiving start bit task

    while (start == 0)
    {
     if (byteArrived==1)//((COMSTA0 & 0x01) == 0x01)
    // If the Rx buffer full bit is set
     {
       //ucRxChar = COMRX;                              // read COMRX register
     if (ucRxChar != 33)                              //text received is not !
     {
       sprintf ( (char*)szTemp, "Received Charachter was : %d,\r\n\n",ucRxChar );
       // Send the Received charachter to the UART

       nLen = strlen((char*)szTemp);
         for ( i = 0 ; i < nLen ; i++ )
     // loop to send String
         {
           COMTX = szTemp[i];
       // Load Tx buffer
         ucTxBufferEmpty = 0;
           while (ucTxBufferEmpty == 0)
         //wait byte to be sent
             {
             }
         ucTxBufferEmpty = 0;
       }
     }
     else
     {

       start=1;
     }


     byteArrived=0;
    }
  }
  IRQEN = BIT10 + BIT11;  // Enable UART bit11 - ADC0+1 interrupts source bit10

}
```

6.4 Matlab scripts

6.4.1 Laboratory acquisition program

```
%%program that acquire the data from the serial port –SEPARATED CHANNEL– LAB use
%% adjust che port COMx number accordig to the PC line 13
%% if something goes wrong during power changing on the aduc the port remains opened
%% to close it use the command  fclose(instrfind) and reset the aduc

clc
clear all
dimension = 4000; %numer of sample to acquire
format short
convert=1200/16777215; %conversion factor
mask1=0x00ffffff;
mask2=0xff000000;
%a=0x01ffffff

s = serial('COM7', ...
                'BaudRate', 230400, ... %230400,
                'Parity', 'none', ...
                'DataBits', 8, ...
                'StopBits', 1);
set(s, 'TimeOut', 10);
set(s, 'InputBufferSize', dimension*4 );
fopen(s)

%CH1 reading
%activate the transmission '!' character sent %start character  g1,26–28
fwrite(s, 33);
a = fread(s, dimension, "uint32"); %%read 4 byte=uint32 x dimension
for i=1:1:dimension

%Extract the correct bytes

a_data(i,1)= bitand(a(i,1),mask1);
a_order(i,1)= bitshift(bitand(a(i,1),mask2),–24);
end
CH1_OPA = a_data;
a_order1=a_order;
fwrite(s, 22); %stop character
fclose(s);
pause(2)

%CH2 reading
fopen(s)
fwrite(s, 33);
a = fread(s, dimension, "uint32"); %%read 4 byte=uint32 x dimension
```

```
for i=1:1:dimension
a_data(i,1)= bitand(a(i,1),mask1);
a_order(i,1)= bitshift(bitand(a(i,1),mask2),-24);
end
CH2_TLC = a_data;
a_order2=a_order;
fwrite(s, 22);
fclose(s);
pause(2)

%TECNIQUE READING
fopen(s)
fwrite(s, 33);
a = fread(s, dimension, "uint32"); %%read 4 byte=uint32 x dimension
for i=1:1:dimension
a_data(i,1)= bitand(a(i,1),mask1);
a_order(i,1)= bitshift(bitand(a(i,1),mask2),-24);
end
CH_METHOD = a_data;
a_order3=a_order;
fwrite(s, 22);
fclose(s);

%%data and time conversion
vettx=1:dimension;
vettx=vettx*0.250;
CH1_double=double(CH1_OPA)*convert;
CH2_double=double(CH2_TLC)*convert;
CH_METHOD_double=double(CH_METHOD)*convert;
a_data_double=double(a_data);
a_data_convert=a_data_double*convert;

 %MAX_CH1=max(CH1_double)
 %MIN_CH1=min(CH1_double)
 MEAN_CH1=mean(CH1_double)
%
 %MAX_CH2=max(CH2_double)
 %MIN_CH2=min(CH2_double)
 MEAN_CH2=mean(CH2_double)
%
 %MAX_METHOD=max(CH_METHOD_double)
 %MIN_METHOD=min(CH_METHOD_double)
 MEAN_METHOD=mean(CH_METHOD_double)

figure
%plot(vettx,a_data_convert)
plot(vettx, CH1_double) %time in ms and data in mV
hold on
plot(vettx, CH2_double)
```

```
hold on
plot(vettx, CH_METHOD_double)

title('Test_of_measured_Alpha')
xlabel('Time_[ms]')
ylabel('Amplitude_[mV]')
legend('OPA', 'TLC', 'METHOD')
%fclose(instrfind)
```

6.4.2 Standard acquisition program

```
%%program that acquire the data from the serial port only method channel
%% adjust che port COMx number accordig to the PC line 13
%% if something goes wrong during power changing on the aduc the port remains opened
%% to close it use the command  fclose(instrfind) and reset the aduc

clc
clear all
dimension = 4000;  %number of sample to acquire
convert=1200/16777215;
mask1=0x00ffffff;
mask2=0xff000000;
%a=0x01ffffff
%s = serialport("COM7",9600)
s = serial('COM7', ...
                'BaudRate', 230400, ... %230400,
                'Parity', 'none', ...
                'DataBits', 8, ...
                'StopBits', 1);
set(s, 'TimeOut', 10);
set(s, 'InputBufferSize', dimension*4 )
fopen(s)
fwrite(s, 33); %activate the transmission '!' character sent


%reading
fwrite(s, 33);
a = fread(s, dimension, "uint32"); %%read 4 byte=uint32 x dimension
for i=1:1:dimension
a_data(i,1)= bitand(a(i,1),mask1);
a_order(i,1)= bitshift(bitand(a(i,1),mask2),-24);
end
CH1_OPA = a_data;
fwrite(s, 22); sto char


fclose(s);
```

```
vettx=1:dimension;
vettx=vettx*0.250;
a_data_double=double(a_data)
a_data_convert=a_data_double*convert;


figure
hold on

plot(vettx,a_data_convert)
%plot(vettx, CH_METHOD_double) %time in ms and data in mV
% hold on
% plot(vettx, CH_METHOD_double)
% hold on
% plot(vettx, CH_METHOD_double)
% legend('OPA', 'TLC', 'METHOD')

title('Test of measured Alpha')
xlabel('Time [ms]')
ylabel('Amplitude [mV]')
```

6.4.3 Plot and shift program

```
%%%program to adjust the phase for time plotting
%CH1 = CH1_double  %%data from acquisition
%CH2 = CH2_double
%CH3 = CH_METHOD_double
$adjust bin_min OPA , bin_int method, bin_max TLC

clear CH1
clear CH2
clear CH3

dbtex= 7;    %ploting parameters
waveform=['Sin'];
hertz= 400;


text_title= [num2str(hertz),'Hz ',num2str(dbtex), 'dBm ',waveform, ' AM_modulated']; %+
    emi

lim_maxplot = 1300;  %plot limits
lim_minplot = 100;
limits_single= [100 1300];
```

```
bin_min=1;  %shift
bin_int=138;
bin_max=139;

maxdif=bin_max-bin_min;
dimension = 4000-maxdif;
vettx=1:dimension;
vettx=vettx*0.250; %conversion to ms

limits=[lim_minplot lim_maxplot ];
limitsy=[0 dimension*0.250];

dimension_all = 4000;
vettx_all=1:dimension_all;
vettx_all=vettx_all;


CH1 = CH1_double(1:end-maxdif); %minimum bin here
CH2 = CH2_double((bin_max-bin_min+1):end);    %max binhere
CH3 = CH_METHOD_double(bin_int:(end-(maxdif-bin_int+1))); %intermediate bin



%MAX_CH1=max(CH1_double)
 %MIN_CH1=min(CH1_double)
 MEAN_CH1=round(mean(CH1_double),2)
%
 %MAX_CH2=max(CH2_double)
 %MIN_CH2=min(CH2_double)
 MEAN_CH2=round(mean(CH2_double),2)
%
 %MAX_METHOD=max(CH_METHOD_double)
 %MIN_METHOD=min(CH_METHOD_double)
 MEAN_METHOD=round(mean(CH_METHOD_double),2)


%% single Plot

figure
plot(vettx, CH1,'r') %time in ms and data in mV
title(['OPA_Output_signal_with_',text_title, '_EMI'])
xlabel('Time_[ms]')
ylabel('Amplitude_[mV]')
legend(['OPA,_DC=', num2str(MEAN_CH1),'mV'],'FontSize', 20)
set(gca,'FontSize',35)
ylim([limits_single] )
xlim(limitsy)

figure
```

```
plot(vettx, CH2,'b') %time in ms and data in mV
title(['TLC Output signal with ',text_title, ' EMI'])
xlabel('Time [ms]')
ylabel('Amplitude [mV]')
legend(['TLC, DC=', num2str(MEAN_CH2), 'mV'],'FontSize', 20)
set(gca,'FontSize',35)
ylim([limits_single] )
xlim(limitsy)


figure
plot(vettx, CH3,'k') %time in ms and data in mV
title(['METHOD Output signal with ',text_title, ' EMI'])
xlabel('Time [ms]')
ylabel('Amplitude [mV]')
legend(['METHOD, DC=', num2str(MEAN_METHOD),'mV'],'FontSize', 20)
set(gca,'FontSize',35)
ylim([limits_single] )
xlim(limitsy)

%% Plot All

figure
plot(vettx, CH1) %time in ms and data in mV
hold on
plot(vettx, CH2)
hold on
plot(vettx, CH3)

title(['Output signals with ',text_title, ' EMI'])
xlabel('Time [ms]')
ylabel('Amplitude [mV]')
legend(['OPA, DC=', num2str(MEAN_CH1),'mV'],['TLC, DC=', num2str(MEAN_CH2),'mV'
    ],...
      ['METHOD, DC=', num2str(MEAN_METHOD),'mV'],'FontSize', 20)
set(gca,'FontSize',35)

ylim([limits] )
xlim(limitsy)
%ylim([num2str(lim_maxplot) num2str(lim_minplot)])


figure
plot(vettx_all, CH1_double) %time in ms and data in mV
hold on
plot(vettx_all, CH2_double)
hold on
plot(vettx_all, CH_METHOD_double)
```

```
title('Original Output signal')
xlabel('Time [ms]')
ylabel('Amplitude [mV]')
legend('OPA', 'TLC', 'METHOD')
set(gca,'FontSize',35)
```

6.4.4 Offset plotting

```
%% offset versus power.
% the Offset_db and Offset_MHz contains the measured offset (by hand or by acquisition)
% Offset_no_xxx contains the offset for the selected cases
% Power contains the injected power
% Freq contains the frequencies used
%Alpha=0.3629318;

limfreq_down=50;
limfreq_up=500;
limdb_down=-15;
limdb_up=9;


bias_db(1:size(Power,1))=Offset_no_RF_db(1);

figure
hold on
plot(Power, Offset_db(:,1), '-or')  %OPA2277
plot(Power, Offset_db(:,2),'-*b')   %TLC272
plot(Power, Offset_db(:,3),'-xk')   %method
plot(Power, bias_db, 'k')           %DC Bias
title(['EMI induced offset vs incident power @100MHz, Alpha=', num2str(Alpha)])
xlabel('EMI incident Power [dBm]')
ylabel('EMI induced offset [mV]')
legend('OPA2277', 'TLC272', 'DC_bias', 'METHOD','Location','northwest')%'DC bias')
set(gca,'FontSize',35)
xlim ([limdb_down limdb_up])
grid on



%% VERSUS FREQUENCY

bias_mhx(1: size(Freq,1))= Offset_no_RF_mhz(1);%Dc bias reference = OPA

figure
hold on
plot(Freq, Offset_MHz(:,1), '-or')  %OPA2277
plot(Freq, Offset_MHz(:,2),'-*b')   %TLC272
```

```
plot(Freq, bias_mhx, 'k')
plot(Freq, Offset_MHz(:,3),'-xk')   %method

title(['EMI_induced_offset_vs_frequency__@0dBm_,_Alpha=', num2str(Alpha)])
xlabel('EMI_Frequency_[MHz]')
ylabel('EMI_induced_offset_[mV]')
legend('OPA2277', 'TLC272', 'DC_bias', 'METHOD','Location','northeast')%'DC bias')
set(gca,'FontSize',35)
xlim ([limfreq_down limfreq_up])
grid on



%% EMIRR computation
Pinc_lin=10.^(Power/10)/1000;   %power in Watt
Vemi_pk_sq= Pinc_lin*2*50;
Vemi_pk=sqrt(Vemi_pk_sq)*2;     %time 2 to account for mismath |gamma| almost 1
Vemi_pk_square = Vemi_pk.^2;


DeltaVoff_db_OPA = abs(Offset_db(:,1) - Offset_no_RF_db(1))/1000;      %Delta offsets vs
    power
DeltaVoff_db_TLC =  abs(Offset_db(:,2) - Offset_no_RF_db(1))/1000;     %delta Refernced all
    to OPA (1)
DeltaVoff_db_METHOD =  abs(Offset_db(:,3) - Offset_no_RF_db(1))/1000;


DeltaVoff_MHz_OPA = abs(Offset_MHz(:,1) - Offset_no_RF_mhz(1))/1000;    %Delta offset Vs
     freq
DeltaVoff_MHz_TLC =  abs(Offset_MHz(:,2) - Offset_no_RF_mhz(1))/1000;
DeltaVoff_MHz_METHOD =  abs(Offset_MHz(:,3) - Offset_no_RF_mhz(1))/1000;

EMIRR_db_OPA= 20*log10(Vemi_pk_square./(DeltaVoff_db_OPA*0.1));
EMIRR_db_TLC = 20*log10(Vemi_pk_square./(DeltaVoff_db_TLC*0.1));
EMIRR_db_METHOD =20*log10(Vemi_pk_square./(DeltaVoff_db_METHOD*0.1));

EMIRR_MHz_OPA= 20*log10(Vemi_pk_square(6)./(DeltaVoff_MHz_OPA*0.1));    %Power(6)
    contains the 0dBm
EMIRR_MHz_TLC = 20*log10(Vemi_pk_square(6)./(DeltaVoff_MHz_TLC*0.1));
EMIRR_MHz_METHOD = 20*log10(Vemi_pk_square(6)./(DeltaVoff_MHz_METHOD*0.1));




%% FIGURE_PLOTTING EMIRR
figure
hold on
plot(Power, EMIRR_db_OPA, '-or') %OPA2277
plot(Power, EMIRR_db_TLC,'-*b') %TLC272
```

```
plot(Power, EMIRR_db_METHOD,'-xk') %method


title(['EMIRR vs incident power @100MHz, Alpha=', num2str(Alpha)])
xlabel('EMI incident Power [dBm]')
ylabel('EMIRR [dB]')
legend('OPA2277', 'TLC272', 'METHOD','Location','northwest')
set(gca,'FontSize',35)
xlim ([limdb_down limdb_up])
grid on

figure
plot(Freq, EMIRR_MHz_OPA, '-or') %opa
hold on
plot(Freq, EMIRR_MHz_TLC,'-*b') %TLC272
plot(Freq, EMIRR_MHz_METHOD,'-xk') %method

title(['EMIRR vs frequency  @0dBm, Alpha=', num2str(Alpha)])
xlabel('EMI Frequency [MHz]')
ylabel('EMIRR [dB]')
legend('OPA2277', 'TLC272', 'METHOD','Location','northwest')
set(gca,'FontSize',35)
xlim ([limfreq_down limfreq_up])
grid on
```

6.4.5 Offset vs power plotting, alpha refining

```
%% offset versus power vs alpha.
%%automatic data read from Af_x.mat for EMI vs freq and
%% A_x.mat for EMI vs power the data saved on the .mat
%% are CH1_double, CH2_double, CH_METHOD_double acquired from the acquisition
    program "conversione_multipla.m"
%% the xxxx_ADUC variable are the data obtained from the aduc, saved and renamed for
    comparisons
%%%%%alpha=0.3629318;
%%%%Offset_db(:,4)=alpha*Offset_db(:,1)+(1-alpha)*Offset_db(:,2);
alpha_min=0.22;
step=0.05;
alpha_max=0.32;
Power=[-15; -12; -9; -6; -3; 0; 3; 6; 9];
limfreq_down=50;
limfreq_up=500;
limdb_down=-15;
limdb_up=9;
Freq=[8;10;20;50; 75; 100; 200; 300; 400; 500];

clear('CH1_p_mean')
```

```
clear('CH2_p_mean')
clear('CH3_p_mean')
clear('Legend')
clear('LegendP_EMIR')
clear('alphaval')
clear ('y1')
clear('m')
clear('bias')
clear('EMIRR_db_METHOD_ALPHA')
DC_power=0;
DC_freq=0;
%%automatic data read from Af_x.mat for EMI vs freq and
%% A_x.mat for EMI vs power the data saved on the .mat
%% are CH1_double, CH2_double, CH_METHOD_double acquired from the acquisition
    program "conversione_multipla.m"
% % % % for p=0:1:10
% % % %     load(['Af_' num2str(p)'])
% % % %     eval(['B_' num2str(p) '_1= CH1_double;']);
% % % %     eval(['B_' num2str(p) '_2= CH2_double;']);
% % % %     eval(['B_' num2str(p) '_3= CH_METHOD_double;']);
% % % %
% % % % end
% % % % for p=0:1:10
% % % %     load(['A_' num2str(p)'])
% % % %     eval(['A_' num2str(p) '_1= CH1_double;']);
% % % %     eval(['A_' num2str(p) '_2= CH2_double;']);
% % % %     eval(['A_' num2str(p) '_3= CH_METHOD_double;']);
% % % %
% % % % end


for i=0:1:9
    CH1_p_mean(i+1) = mean(eval(['A_' num2str(i) '_1'])); %no RF at i=1
    CH2_p_mean(i+1) = mean(eval(['A_' num2str(i) '_2']));
    k=0;
    %figure
    %hold on
    for n=alpha_min:step:alpha_max
        k=k+1;
        y1 = n* eval(['A_' num2str(i) '_1']) + (1-n) * eval(['A_' num2str(i) '_2']);
        %plot(y1)
        alphaval(k)= n;
        CH3_p_mean(i+1,k)=mean(y1);
    end

end

    CH1_p_mean = (CH1_p_mean)';
    CH2_p_mean = (CH2_p_mean)';
```

```
%% plot vs alfa

    bias(1:9)= CH1_p_mean(1);
    figure
    hold on
    plot(Power, bias, 'k') % bias
    plot(Power, CH1_p_mean(2:end), '-or','LineWidth',3) %OPA2277
    plot(Power, CH2_p_mean(2:end), '-*b','LineWidth',3) %TLC2277
    plot(Power, Offset_db(:,3), 'LineWidth',5)
    Legend{1}= 'Bias';
    Legend{2}= 'OPA';
    Legend{3}= 'TLC';
    Legend{4}= '\alpha=0.3629318';;


    max_value=k;
    %for i=1:1:9
        for k=1:1:max_value
            plot(Power, CH3_p_mean(2:end,k),'-x','LineWidth',2) %method
            Legend{k+4}= num2str(round(alphaval(k),5));
            %alphaval(k)
            %pause
        end
    %end
    legend(Legend,'Location','northwest')
    title(['EMI_induced_offset_vs_power_@100MHz_for_different_value_of_\alpha'])
    xlabel('EMI_incident_power_[dBm]')
    ylabel('EMI_induced_offset_[mV]')
    %set(gca, 'Xscale', 'log')
    set(gca,'FontSize',35)
    xlim ([limdb_down limdb_up])
    grid on

%%%%%plot(Power, Offset_db(:,4),'-gs') %newalpha


%% plot EMIRR

Pinc_lin_ALPHA=10.^(Power/10)/1000; %power in Watt
Vemi_pk_sq_ALPHA= Pinc_lin_ALPHA*2*50;
Vemi_pk_ALPHA=sqrt(Vemi_pk_sq_ALPHA)*2; %time 2 to account for mismath |gamma|
    almost 1
Vemi_pk_square_ALPHA = Vemi_pk_ALPHA.^2;


DeltaVoff_db_OPA_ALPHA = abs(CH1_p_mean(2:end) - CH1_p_mean(1))/1000;
DeltaVoff_db_TLC_ALPHA =  abs(CH2_p_mean(2:end) - CH1_p_mean(1))/1000;
DeltaVoff_db_METHOD_LIN_ALFA =CH3_p_mean(2:end,:) - CH1_p_mean(1);
DeltaVoff_db_METHOD_ALPHA =  abs(CH3_p_mean(2:end,:) - CH1_p_mean(1))/1000;
```

```
figure
hold on
title('DELTAVOFF_in_Volt_ABS')
plot(DeltaVoff_db_METHOD_ALPHA)
plot(DeltaVoff_db_METHOD_ADUC, 'LineWidth',5)

%DeltaVoff_MHz_OPA = abs(Offset_MHz(:,1) - Offset_no_RF(1))/1000
%DeltaVoff_MHz_TLC =  abs(Offset_MHz(:,2) - Offset_no_RF(2))/1000
%DeltaVoff_MHz_METHOD =  abs(Offset_MHz(:,3) - Offset_no_RF(3))/1000


EMIRR_db_OPA_ALPHA= 20*log10(Vemi_pk_square_ALPHA./(DeltaVoff_db_OPA_ALPHA
    *0.1));
EMIRR_db_TLC_ALPHA = 20*log10(Vemi_pk_square_ALPHA./(DeltaVoff_db_TLC_ALPHA
    *0.1));

figure
hold on
title('EMIRR_in_dB_COMPARISON_')
plot(EMIRR_db_METHOD_ADUC,'LineWidth',5)
for m=1:1:size(DeltaVoff_db_METHOD_ALPHA,2)
   EMIRR_db_METHOD_ALPHA(:,m) =20*log10(Vemi_pk_square_ALPHA./(
     DeltaVoff_db_METHOD_ALPHA(:,m)*0.1));  %each coloum is a plot

   plot(EMIRR_db_METHOD_ALPHA)

end

% EMIRR_MHz_OPA= 20*log10(Vemi_pk_square(4)./(DeltaVoff_MHz_OPA*0.1))
% EMIRR_MHz_TLC = 20*log10(Vemi_pk_square(4)./(DeltaVoff_MHz_TLC*0.1))
% EMIRR_MHz_METHOD = 20*log10(Vemi_pk_square(4)./(DeltaVoff_MHz_METHOD*0.1))




figure
hold on
%title('EMI rejection ratio vs incident power @100MHz')
title('EMIRR_vs_power_for_different_values_of_\alpha')
xlabel('EMI_incident_Power_[dBm]')
ylabel('EMIRR_[dB]')
%set(gca,'FontSize',35)
xlim ([limdb_down limdb_up])
grid on


plot(Power, EMIRR_db_OPA_ALPHA, '-or','LineWidth',3) %OPA2277
plot(Power, EMIRR_db_TLC_ALPHA,'-*b','LineWidth',3) %TLC272
```

```
plot(Power, EMIRR_db_METHOD_ADUC,'LineWidth',5)
LegendP_EMIR{1}= 'OPA';
LegendP_EMIR{2}= 'TLC';
LegendP_EMIR{3}= '\alpha=0.3629318';

%plot in this way to each components and eventually add a pause
for m=1:1:size(DeltaVoff_db_METHOD_ALPHA,2)
    alphaval(m)
    plot(Power, EMIRR_db_METHOD_ALPHA(:,m),'-x','LineWidth',2) %method
    LegendP_EMIR{m+3}=num2str(round(alphaval(m),5));
end


legend(LegendP_EMIR ,'Location','northwest','fontsize',18)
```

6.4.6 Offset vs frequency plotting, alpha refining

```
%% offset versus freq vs alpha.
%%automatic data read from Af_x.mat for EMI vs freq and
%% A_x.mat for EMI vs power the data saved on the .mat
%% are CH1_double, CH2_double, CH_METHOD_double acquired from the acquisition
    program "conversione_multipla.m"
%% the xxxx_ADUC variable are the data obtained from the aduc, saved and renamed for
    comparisons
%%%%%alpha=0.3629318;
%%%%Offset_db(:,4)=alpha*Offset_db(:,1)+(1-alpha)*Offset_db(:,2);
alpha_min=0.21;
step=0.01;
alpha_max=0.23;

limfreq_down=50;
limfreq_up=500;
limdb_down=-15;
limdb_up=9;

Freq=[8;10;20;50; 75; 100; 200; 300; 400; 500];
Power=[-15; -12; -9; -6; -3; 0; 3; 6; 9];
clear('CH1_p_mean_freq')
clear('CH2_p_mean_freq')
clear('CH3_p_mean_freq')
clear('Legend')
clear('LegendP_EMIR_freq')
clear('alphaval_freq')
clear ('y1')
clear('m')
clear('bias')
clear('EMIRR_MHZ_METHOD_ALPHA')
```

```
DC_power=0;
DC_freq=0;

%%automatic data read from Af_x.mat for EMI vs freq and
%% A_x.mat for EMI vs power the data saved on the .mat
%% are CH1_double, CH2_double, CH_METHOD_double acquired from the acquisition
    program "conversione_multipla.m"
% % % % for p=0:1:10
% % % %     load(['Af_' num2str(p)'])
% % % %     eval(['B_' num2str(p) '_1= CH1_double;']);
% % % %     eval(['B_' num2str(p) '_2= CH2_double;']);
% % % %     eval(['B_' num2str(p) '_3= CH_METHOD_double;']);
% % % %
% % % % end
% % % % for p=0:1:10
% % % %     load(['A_' num2str(p)'])
% % % %     eval(['A_' num2str(p) '_1= CH1_double;']);
% % % %     eval(['A_' num2str(p) '_2= CH2_double;']);
% % % %     eval(['A_' num2str(p) '_3= CH_METHOD_double;']);
% % % %
% % % % end



%% frequency

for i=0:1:size(Freq,1)
    CH1_p_mean_freq(i+1) = mean(eval(['B_' num2str(i) '_1'])); %no RF at i=1
    CH2_p_mean_freq(i+1) = mean(eval(['B_' num2str(i) '_2']));
    k=0;
    %figure
    %hold on
    for n=alpha_min:step:alpha_max
        k=k+1;
        y1 = n* eval(['B_' num2str(i) '_1']) + (1-n) * eval(['B_' num2str(i) '_2']);
        %plot(y1)
        alphaval_freq(k)= n;
        CH3_p_mean_freq(i+1,k)=mean(y1);
    end

end

    CH1_p_mean_freq = (CH1_p_mean_freq)';
    CH2_p_mean_freq = (CH2_p_mean_freq)';
%% plot vs alfa

    bias(1:10)= CH1_p_mean_freq(1);
    figure
    hold on
```

```
    plot(Freq, bias, 'k') % bias
    plot(Freq, CH1_p_mean_freq(2:end), '-or','LineWidth',3) %OPA2277
    plot(Freq, CH2_p_mean_freq(2:end), '-*b','LineWidth',3) %TLC2277
    plot(Freq, Offset_MHz(:,3), 'LineWidth',5)
    Legend{1}= 'Bias';
    Legend{2}= 'OPA';
    Legend{3}= 'TLC';
    Legend{4}= '\alpha=0.3629318';

    max_value=k;
    %for i=1:1:9
        for k=1:1:max_value
            plot(Freq, CH3_p_mean_freq(2:end,k),'-x','LineWidth',2) %method
            Legend{k+4}= ['\alpha=' num2str(round(alphaval_freq(k),5))];
            %alphaval(k)
            %pause
        end
    %end
    legend(Legend)
    title(['EMI_induced_offset_vs_frequency__@0dBm_for_different_value_of_\alpha'])
    xlabel('EMI_Frequency_[MHz]')
    ylabel('EMI_induced_offset_[mV]')
    set(gca, 'Xscale', 'log')
    set(gca,'FontSize',35)
    xlim ([limfreq_down limfreq_up])
    grid on

%%%%%plot(Power, Offset_db(:,4),'-gs') %newalpha


%% plot EMIRR

Pinc_lin_ALPHA_freq=10.^(Power/10)/1000; %power in Watt
Vemi_pk_sq_ALPHA_freq= Pinc_lin_ALPHA_freq*2*50;
Vemi_pk_ALPHA_freq=sqrt(Vemi_pk_sq_ALPHA_freq)*2; %time 2 to account for mismath |
    gamma| almost 1
Vemi_pk_square_ALPHA_freq = Vemi_pk_ALPHA_freq.^2;


DeltaVoff_MHZ_OPA_ALPHA = abs(CH1_p_mean_freq(2:end) - CH1_p_mean_freq(1))/1000;
DeltaVoff_MHZ_TLC_ALPHA =  abs(CH2_p_mean_freq(2:end) - CH1_p_mean_freq(1))/1000;
DeltaVoff_MHZ_METHOD_LIN_freq =CH3_p_mean_freq(2:end,:) - CH1_p_mean_freq(1);
DeltaVoff_MHZ_METHOD_ALPHA =  abs(CH3_p_mean_freq(2:end,:) - CH1_p_mean_freq(1))
    /1000;


figure
hold on
title('DELTAVOFF_in_Volt_ABS')
```

```
plot(DeltaVoff_MHZ_METHOD_ALPHA)
plot(DeltaVoff_MHz_METHOD_ADUC, 'LineWidth',5)
legend('ALPHA', 'ADUC')

%DeltaVoff_MHz_OPA = abs(Offset_MHz(:,1) - Offset_no_RF(1))/1000
%DeltaVoff_MHz_TLC =  abs(Offset_MHz(:,2) - Offset_no_RF(2))/1000
%DeltaVoff_MHz_METHOD =  abs(Offset_MHz(:,3) - Offset_no_RF(3))/1000


EMIRR_MHZ_OPA_ALPHA= 20*log10(Vemi_pk_square_ALPHA_freq(6)./(
    DeltaVoff_MHZ_OPA_ALPHA*0.1));
EMIRR_MHZ_TLC_ALPHA = 20*log10(Vemi_pk_square_ALPHA_freq(6)./(
    DeltaVoff_MHZ_TLC_ALPHA*0.1));

figure
hold on
title('EMIRR in dB COMPARISON ')
plot(EMIRR_MHz_METHOD_ADUC,'LineWidth',5)
set(gca, 'Xscale', 'log')

%%plot in this way to each components and eventually add a pause
for m=1:1:size(DeltaVoff_MHZ_METHOD_ALPHA,2)
   EMIRR_MHZ_METHOD_ALPHA(:,m) =20*log10(Vemi_pk_square_ALPHA_freq(6)./(
     DeltaVoff_MHZ_METHOD_ALPHA(:,m)*0.1));  %each coloum is a plot

   plot(EMIRR_MHZ_METHOD_ALPHA)

end


% EMIRR_MHz_OPA= 20*log10(Vemi_pk_square(4)./(DeltaVoff_MHz_OPA*0.1))
% EMIRR_MHz_TLC = 20*log10(Vemi_pk_square(4)./(DeltaVoff_MHz_TLC*0.1))
% EMIRR_MHz_METHOD = 20*log10(Vemi_pk_square(4)./(DeltaVoff_MHz_METHOD*0.1))




figure
hold on
%title('EMI rejection ratio vs incident power @100MHz')
title('EMIRR vs frequency @0dBm for various value of \alpha')
xlabel('EMI Frequency [MHz]')
ylabel('EMIRR [dB]')
%set(gca,'FontSize',35)
set(gca, 'Xscale', 'log')
set(gca,'FontSize',35)
xlim ([limfreq_down limfreq_up])
grid on
```

```
plot(Freq, EMIRR_MHZ_OPA_ALPHA, '-or','LineWidth',3) %OPA2277
plot(Freq, EMIRR_MHZ_TLC_ALPHA,'-*b','LineWidth',3) %TLC272
plot(Freq, EMIRR_MHz_METHOD_ADUC,'LineWidth',5)
LegendP_EMIR_freq{1}= 'OPA';
LegendP_EMIR_freq{2}= 'TLC';
LegendP_EMIR_freq{3}= '\alpha=0.3629318';

%%plot in this way to each components and eventually add a pause
for m=1:1:size(DeltaVoff_MHZ_METHOD_ALPHA,2)
    alphaval_freq(m)
    plot(Freq, EMIRR_MHZ_METHOD_ALPHA(:,m),'-x','LineWidth',2 ) %method
    LegendP_EMIR_freq{m+3}=['\alpha=' num2str(round(alphaval_freq(m),5))];
end


legend(LegendP_EMIR_freq, 'Location','northwest','fontsize',18)
```

6.4.7 FFT computation single channels

```
%% FFT and SNDR SFDR computation
dbtex= 0; %%data for the plotting
waveform=['Exp.'];
hertz= 1000;

Ts=1;                %sample time
L=4000;               %numer of samples zero included, zero added if L>N
fc=4000;              %samplig at 4000Hz
Tc=1/fc;
N=Ts/Tc;                %4000;   %Real samples

f=(0:(L-1)/2)/(L*Tc);   %fft frequencies definition


text_title= [num2str(hertz),'Hz ',num2str(dbtex), 'dBm ',waveform, ' AM modulated']; %+
    emi

%Windows definition
% flattopwin(N, 'periodic') ;%blackman(N, 'periodic') hanning(N);%blackman(N);
w = blackman(N, 'periodic');


CH1_wind = (CH1_double-mean(CH1_double)).*w; %bias removing and windowing
CH2_wind = (CH2_double-mean(CH2_double)).*w;
CH3_wind = (CH_METHOD_double-mean(CH_METHOD_double)).*w;

%no DC removing
```

```
% % % % % CH1_wind = CH1_double.*w;
% % % % % CH2_wind = CH2_double.*w;
% % % % % CH3_wind = CH_METHOD_double.*w


%fft computation and normalizing
CH1_fft= 1/N*fft(CH1_wind,L);
CH2_fft= 1/N*fft(CH2_wind,L);
CH3_fft= 1/N*fft(CH3_wind,L);

%single side fft  conversion
CH1_fft_fs(1)= CH1_fft(1); %no Dc multiplication
CH1_fft_fs(2:L/2)= 2*CH1_fft(2:L/2); %for real signal *2
CH2_fft_fs(1)= CH2_fft(1);
CH2_fft_fs(2:L/2)= 2*CH2_fft(2:L/2);
CH3_fft_fs(1)= CH3_fft(1);
CH3_fft_fs(2:L/2)= 2*CH3_fft(2:L/2);


%% conversion for plotting

%%LINEAR
% zerodb=10.81;
% CH3_db= abs(CH3_fft_fs) ; %blackman +7.54 Coherent power gain + 1.10db Scalopping
    loss max
% maxdb = max(CH3_db) ;     %+2.38 equivanent noise bandwidth
% CH1_db= abs(CH1_fft_fs);
% CH2_db= abs(CH2_fft_fs);

% DBVALUES
Fond_bin = min(find(CH3_fft_fs==max(CH3_fft_fs)));  % find foundamental bin
zerodb= mag2db(abs(CH3_fft_fs(Fond_bin)));          % convert to dBc

CH3_db= mag2db(abs(CH3_fft_fs)) -zerodb;            %dBc conversion. if not used the value
    are in dBmV
CH1_db= mag2db(abs(CH1_fft_fs)) -zerodb;
CH2_db= mag2db(abs(CH2_fft_fs)) -zerodb;

SINAD_CH1= round(sinad(CH1_double),2)
SINAD_CH2= round(sinad(CH2_double,4000),2)
SINAD_CH3= round(sinad(CH_METHOD_double,4000),2)
SFDR_CH1= round(sfdr(CH1_double),2)
SFDR_CH2= round(sfdr(CH2_double,4000),2)
SFDR_CH3= round(sfdr(CH_METHOD_double,4000),2)


%% figure plot
figure
plot(f, CH1_db,'red') %green
```

```
grid on
set(gca, 'Xscale', 'log')
set(gca,'FontSize',35)
xlabel('Frequency /[Hz]')
xlim ([4 2000])
%ylim([-140,0])
ylabel('Magnitude of FFT [dBc]')
legend('OPA2277', 'Fontsize', 20)
title({ ['OPA2277 Output spectrum ', text_title ,' EMI']
        ['SNDR=',num2str(SINAD_CH1),'dB, SFDR=', num2str(SFDR_CH1),'dB']}, 'FontSize',
       35)



figure
plot(f, CH2_db,'magenta') %yellow
grid on
set(gca, 'Xscale', 'log')
hold on
set(gca,'FontSize',35)
xlabel('Frequency /[Hz]')
xlim ([4 2000])
%ylim([-140,0])
ylabel('Magnitude of FFT [dBc]')
legend('TLC272', 'FontSize', 20)
title({ ['TLC272 Output spectrum ',text_title, ' EMI']
        ['SNDR=',num2str(SINAD_CH2),'dB, SFDR=', num2str(SFDR_CH2),'dB']}, 'FontSize',
       35)

figure
plot(f, CH3_db,'black') %red
grid on
set(gca, 'Xscale', 'log')
hold on
set(gca,'FontSize',35)
xlabel('Frequency /[Hz]')
xlim ([4 2000])
%ylim([-140,0])
ylabel('Magnitude of FFT [dBc]')
legend( 'METHOD', 'Fontsize', 20)
title({ ['METHOD Output spectrum ',text_title, ' EMI']
        ['SNDR=',num2str(SINAD_CH3),'dB, SFDR=', num2str(SFDR_CH3),'dB']}, 'FontSize',
       35)


%%Plot all
figure
hold on
plot(f, CH1_db,'red')
```

```
plot(f, CH2_db,'magenta')
plot(f, CH3_db,'black')
grid on
set(gca, 'Xscale', 'log')
set(gca,'FontSize',35)
xlabel('Frequency_/[Hz]')
xlim ([4 2000])
%ylim([-140,0])
ylabel('Magnitude_of_FFT_[dBc]')
legend('OPA2277','TLC272','METHOD', 'Fontsize', 20)
title({ ['OUTPUT_SPECTRA_WITH_', text_title, '_EMI']
        ['SNDR\_OPA=',num2str(SINAD_CH1),'dB,_SFDR\_OPA=', num2str(SFDR_CH1),'dB'
    ]
      ['SNDR\_TLC=',num2str(SINAD_CH2),'dB,_SFDR\_TLC=', num2str(SFDR_CH2),'dB']
      ['SNDR\_METHOD=',num2str(SINAD_CH3),'dB,_SFDR\_METHOD=', num2str(
    SFDR_CH3),'dB']}, 'FontSize', 35)
```

6.4.8 FFT computation all channels

```
%% FFT and SNDR SFDR computation
%% see fft_blackman_singola for data using

Ts=1;                %sample time
L=4000;               %numer of samples zero included, zero added if L>N
fc=4000;              %samplig at 4000Hz
Tc=1/fc;
N=Ts/Tc;                %4000;   %Real samples

f=(0:(L-1)/2)/(L*Tc);   %fft frequencies definition

%Windows definition
% flattopwin(N, 'periodic') ;%blackman(N, 'periodic') hanning(N);%blackman(N);
w = blackman(N, 'periodic');


CH1_wind = (CH1_double-mean(CH1_double)).*w; %bias removing and windowing
CH2_wind = (CH2_double-mean(CH2_double)).*w;
CH3_wind = (CH_METHOD_double-mean(CH_METHOD_double)).*w;

%no DC removing
% % % % % CH1_wind = CH1_double.*w;
% % % % % CH2_wind = CH2_double.*w;
% % % % % CH3_wind = CH_METHOD_double.*w


%fft computation and normalizing
CH1_fft= 1/N*fft(CH1_wind,L);
```

```
CH2_fft= 1/N*fft(CH2_wind,L);
CH3_fft= 1/N*fft(CH3_wind,L);

%single side fft  conversion
CH1_fft_fs(1)= CH1_fft(1); %no Dc multiplication
CH1_fft_fs(2:L/2)= 2*CH1_fft(2:L/2); %for real signal *2
CH2_fft_fs(1)= CH2_fft(1);
CH2_fft_fs(2:L/2)= 2*CH2_fft(2:L/2);
CH3_fft_fs(1)= CH3_fft(1);
CH3_fft_fs(2:L/2)= 2*CH3_fft(2:L/2);


%% conversion for plotting

%%LINEAR
% zerodb=10.81;
% CH3_db= abs(CH3_fft_fs) ; %blackman +7.54 Coherent power gain + 1.10db Scalopping
    loss max
% maxdb = max(CH3_db) ;     %+2.38 equivanent noise bandwidth
% CH1_db= abs(CH1_fft_fs);
% CH2_db= abs(CH2_fft_fs);

% DBVALUES
Fond_bin = min(find(CH3_fft_fs==max(CH3_fft_fs)));  % find foundamental bin
zerodb= mag2db(abs(CH3_fft_fs(Fond_bin)));          % convert to dBc

CH3_db= mag2db(abs(CH3_fft_fs)) -zerodb;            %dB conversion for dBmV add 7.54 of
    blackman in
CH1_db= mag2db(abs(CH1_fft_fs)) -zerodb;            %this way zerodb=-7.54
CH2_db= mag2db(abs(CH2_fft_fs)) -zerodb;

SINAD_CH1= round(sinad(CH1_double),2)
SINAD_CH2= round(sinad(CH2_double,4000),2)
SINAD_CH3= round(sinad(CH_METHOD_double,4000),2)
SFDR_CH1= round(sfdr(CH1_double),2)
SFDR_CH2= round(sfdr(CH2_double,4000),2)
SFDR_CH3= round(sfdr(CH_METHOD_double,4000),2)

figure
plot(f, CH1_db,'red') %green
hold on
plot(f, CH2_db,'magenta') %yellow
hold on
plot(f, CH3_db,'black') %red
hold on
grid on
set(gca, 'Xscale', 'log')
hold on
set(gca,'FontSize',35)
```

```
xlabel('Frequency /[Hz]')
xlim ([4 2000])
%ylim([-140,0])
ylabel('Magnitude of FFT [dBc]')
legend('OPA', 'TLC', 'METHOD')
title({ ['OPA Output spectra w/o EMI']
        ['SNDR=',num2str(SINAD_CH1),', SFDR=', num2str(SFDR_CH1)]})
```

6.4.9 Alpha to fixed point conversion

```
alpha=0.345;
%alpha=0.3573428
%alpha=0.3641300;
%alpha=0.3629318
%0.3846154 %=0x627627A4 and 0x9D89D85B
%0.385 %=0x628f5c28;  and 0x9d70a3d7
%alpha=0.3629318
binario1=dec2q(alpha,0,32,'bin');
binario2=dec2q(1-alpha,0,32,'bin');

bin1=binario1(2:end);
bin2=binario2(2:end);

alphaHex=bin2hex(binario1)
nonalphahex=bin2hex(binario2)

% bin= = dec2hex(bin2dec(reshape(bin1,32,1)))
%
% binst1=str2num(binario1);
% binst2=str2num(binario2);
%  = dec2hex(bin2dec(reshape(B,4,[]).')).'
%
% binaryVectorToHex(binaryVector)
%
% bin1=textscan(binario1, '%f');
% AlphaHex=binaryVectorToHex(bin1)
% NONAlphahex=binaryVectorToHex(bin2)
```

6.4.10 Alpha evaluation

```
%%Evaluation of alpha graphically

clc
clear all
load tlc_noEmi.mat %data from oscilloscope one reading only
load tlc_Emi.mat
load opa_noEmi.mat
load opa_Emi.mat
lim_down=0.384615
lim_up=0.384616
step_d=0.0000001

dimension=0;

%% vector sizing
A=size(opa_Emi_sh); %set the number of samples
dimension=A(:,1);
x=floor(dimension/100);

figure
hold on
i=0;
clear Legend;
hold on

for alpha=0.384615:0.0000001:0.384616  %loop to plot tecnique for different Alpha
    y= alpha*opa_Emi_sh+(1-alpha)*tlc_Emi_sh;
    plot(y)
    Legend{i}=num2str(alpha);  %generate the leggend for each plot
    % pause
end

plot(opa_noEmi,'LineWidth',8)
plot(tlc_noEmi,'LineWidth',8)

Legend{i+1}='Opa_noEMI';
Legend{i+2}='TLC_noEMI';
legend(Legend)
```

6.4.11 Alpha computation with minimization

```
%%%Alpha computation through minimizing the squared errors

%clc
%clear all

%A_x = different acquired signals from oscilloscope
%A_x(:,j); j=the channel acquired
% 1=OPA_EMI, 2=TLC_EMI, 3-4=OPA-TLC NO EMI
%A_1 = original signal


i=0;
index=3500;
numerator = 0;
denomin = 0;
k=0;
for index= 1:500:300000  %take a sample every 500 of the acquired signal
    k=k+1;
  for i=1:1:12 %12 are the number of different EMI conditions
      y1=eval(['A_' num2str(i) '(' num2str(index) ',1)']); %opa Emi
      x1=eval(['A_' num2str(1) '(' num2str(index) ',3)']); %opa no emi
      y2=eval(['A_' num2str(i) '(' num2str(index) ',2)']); %TLC Emi
      x2=eval(['A_' num2str(1) '(' num2str(index) ',4)']); %TLC no emi
      nEmi_1= y1-x1;
      nEmi_2= y2-x2;
      nEmi_1st(i) = nEmi_1;
      nEmi_2st(i) = nEmi_2;
      denomin = denomin + (nEmi_1-nEmi_2)^2;
      numerator=numerator+(nEmi_2*(nEmi_2-nEmi_1));
      alpha(i)= -nEmi_2/(nEmi_1 - nEmi_2);
      %error(i)= alpha(i,k)*nEmi_1 + (1-alpha(i,k))*nEmi_2;
  end
end
%error= alpha(i,k)*nEmi_1 + (1-alpha(i,k))*nEmi_2;

alpha_s = numerator/denomin

%alpha plotting
%for i=1:1:k
 %   plot(alpha(:,i))
 %   hold on
%end
```

Bibliography

[1] J. Gago et al.
"EMI Susceptibility Model of Signal Conditioning Circuits Based on Operational Amplifiers."
In: IEEE Trans. Electromagn. Compat. 49 (2007), pp. 849–859.

[2] A. Richelli et al.
"Susceptibility of Operational Amplifiers to Conducted EMI Injected Through the Ground Plane into Their Output Terminal".
In: IEEE Trans. Reliab. 65 (2016), pp. 1369–1379.

[3] S. Becchetti et al.
"A Comprehensive Comparison of EMI Immunity in CMOS Amplifier Topologies".
In: Electronics 8 (2019), p. 1181.

[4] Masetti G., Graffi S., and Golzio D.
"Failures Induced on analog integrated circuits by conveyed electromagnetic interferences: a review."
In: Microelectron. Reliab. 36 n.7-8 (1996), pp. 955–972.

[5] A. Lavarda et al.
"On the Robustness of CMOS-Chopped Operational Amplifiers to Conducted Electromagnetic Interferences".
In: IEEE Trans. Electromagn. Compat 60 (2018), pp. 478–486.

[6] S. Boyapati, J.M. Redouté, and M. Shojaei Baghini.
"Design of A Novel Highly EMI-Immune CMOS Miller OpAmp Considering Channel Length Modulation".
In: IEEE Trans. Circuits Syst. I Regul. Pap 64 (2017), pp. 2679–2690.

[7] J.M. Redouté and A. Richelli.
"A methodological approach to EMI resistant analog integrated circuit design."
In: IEEE Electromagn. Compat. Mag. 4 (2015), pp. 92–100.

[8] J. Yu, A. Amer, and Sanchez-Sinencio.

"E. Electromagnetic Interference Resisting Operational Amplifier".
In: IEEE Trans. Circuits Syst. I Regul. Pap. 61 (2014), pp. 1917–1927.

[9] A. Richelli.
"CMOS OpAmp Resisting to Large Electromagnetic Interferences".
In: IEEE Trans. Electromagn. Compat 52 (2010), pp. 1062–1065.

[10] Crovetti P.S.
"Operational amplifier immune to EMI with no baseband performance degradation".
In: Electron. Lett. 46 (2010), pp. 207–208.

[11] Richelli A. et al.
"Robust design of low EMI susceptibility CMOS OpAmp".
In: IEEE Trans. Electromagn. Compat. 6 n.2 (2004), pp. 291–298.

[12] P. S. Crovetti.
Design of Analog Integrated Circuits Robust to RF Interference.
Politecnico di Torino, 2004.

[13] P. S. Crovetti and F. Musolino.
"Digital Suppression of Emi-Induced Errors in a Baseband Acquisition Front-End including Off-the-Shelf, Emi-Sensitive Operationa Amplifier".
In: Electronics 2021, 10, 2096 (2021).
DOI: https://www.mdpi.com/2079-9292/10/17/2096.

[14] Z. Kovacs-Vajna, E. Sardini, and N. Speciale.
"Chaotic behavior of 741 opamps subjected to EMI conveyed to power supply rails."
In: Proceedings of the 2000 (ISCAS) 1 (2000), pp. 727–730.

[15] Fiori F. L. and Crovetti P.S.
"Nonlinear Effects of Radio-Frequency Interference in Operational Amplifiers".
In: IEEE Trans. on circuits and System I: Fund. Theory and Applic. 49 (2002), pp. 367–372.

[16] Fiori F. L.
"A new Nonlinear model of EMI-induced distortion phenomena in feedback CMOS operational Amplifiers".
In: IEEE Trans. Electromag. Compat. 44 (2002), pp. 495–502.

[17] Fiori F. L. and Crovetti P.S.
"Prediction of High-Power Emi Effects in CMOS Operational Amplifier".
In: IEEE Trans. Electromagnetic Comp. 48 (2006), pp. 153–160.

[18] S. Graffi, G. Masetti, and D. Golzio.

"New macromodels and measurements for the analysis of EMI effects in 741 op-amp circuits".
In: IEEE Trans. Electromagn. Compat. 33 (1991), pp. 25–34.

[19] Fiori F. L., Crovetti P. S., and Pozzolo V.
"Prediction of RF Interference in Operational Amplifiers by New Analitical Model".
In: IEEE Int. Symp. on Electr. Comp. 2 (2001), pp. 1164–1668.

[20] M.B. Aimonetto and F. Fiori.
"On the effectiveness of EMIRR to qualify OpAmps."
In:
In Proc. of the 2015 IEEE Int. Symposium on Electrom. Compat.(EMC) (2015), pp. 40–44.

[21] VINOD KUMAR V. M.
"Design and Analysis of High performance and High Slew Rate Class AB Operational Amplifier".
In:
International Journal of Scientific Engineering and Technology Research 4 (2015), pp. 7279–7282.

[22] P.S. Crovetti.
"Finite Common-Mode Rejection in Fully Differential Nonlinear Circuits".
In: IEEE Trans. Circuits Syst. II Express Briefs 58 (2011), pp. 507–511.

[23] H. Ghadamabadi et al.
"Comparison of demodulation RFI in inverting operational amplifier circuits of the same gain with different input and feedback resistor values".
In: IEEE Int. Symp. on EMC 1990, Symp. Rec. 1990 (1990), pp. 145–152.

[24] F. Fiori and P. S. Crovetti.
"Nonlinear effects of RF interference in MOS operational amplifier".
In: in Proc. IEEE Int. Conf. Elect., Circuits and Systems (2001), pp. 201–204.

[25] F. Fiori.
"Design of an operational amplifier input stage immune to EMI".
In: IEEE Trans. Electromagn. Compat. 49 n.4 (2007), pp. 834–839.

[26] J. Gundla, S. Boyapati, and V.R.S. Pasupureddi.
"Compact CMOS Miller OpAmp With High EMI-Immunity".
In: IEEE Trans. Electromagn. Compat 62 (2020), pp. 2394–2400.

[27] Boyapati Subrahmanyam, Redouté Jean-Michel, and Baghini Maryam Shojaei.
"A Robust CMOS Miller OpAmp with High EMI-Immunity".

In: IEEE Trans. Electromagn. Compat. 62 (2020), pp. 2394–2400.

[28] F. Fiori and P. S. Crovetti.
"Complementary differential pair with high immunity to RFI".
In: Electron. Lett. 38 (2002), pp. 1663–1664.

[29] E.D. Totev.
"Out-of-Band Interference Immunity of Negative-Feedback Amplifiers."
In: Ph.D. Thesis, TU Delft, Delft, The Netherlands (2021), pp. 18–32.

[30] P. S. Crovetti.
"Acquisition front-end immune to EMI".
In: Electronics letters (2012).

[31] Frank R. Kschischang.
"The Hilbert Transform".
In:
The Edward S. Rogers Sr. Dep. of Elec. and Comp. Eng. Univ. of Toronto (2006).

[32] Texas Instruments.
"OPAx277 High Precision Operational Amplifiers".
In: Technical Report (2015).

[33] Texas Instruments.
"TLC272 Precision Operational Amplifier".
In: Technical Report (2002).

[34] International Electrotechnical Commission (IEC).
Int. Cir. , Meas. of Electr. Imm., 150 kHz–1 GHz, IEC Std, IEC62132.
Geneve, Switzerland, 2006.

[35] Texas Instruments Inc.
"EMI Rejection Ratio of Operational Amplifiers".
In: Application Report, SBOA128 (2011).

[36] Inc. Analog Devices.
"ADuC7060/ADuC7061".
In: Technical Report (2017).

[37] Microchip Technology Inc.
"Driving Capacitive Loads With Op Amps".
In: Tecnical Report AN884 (2008).

[38] Joe Buxton.
"Careful Design Tames High Speed Op Amps".
In: Analog Devices Application notes AN-257 (1991).

[39] Grayson King.
"ADuC7060/ADuC7061".
In: Analog Dialogue 31.2 (1997), pp. 19–21.

[40] ARM Compiler v5.06 for µVision armasm User Guide.
URL: https://www.keil.com/support/man/docs/armasm/.

[41] Hantek.
"User's Manual - HDG 2000 Series".
In: User Manual V1.0.2 ().

[42] S. Graffi, G. Masetti, and D. Golzio.
"New macromodels and measurements for the analysis of EMI effects in 741 op-amp circuits".
In: IEEE Trans. Electromagn. Compat. 33 (1991), pp. 25–34.