Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Τομέας Επικοινωνιών, Ηλεκτρονικής & Συστημάτων Πληροφορικής

# Analysis and FPGA Implementation of a Single-Bit Two-Step Look-Ahead ΔΣ Modulator

**Διπλωματική Εργασία**

Αθανάσιος Σ. Ορφανός

**Επιβλέπων Καθηγητής**
Παύλος-Πέτρος Σωτηριάδης
Επίκ. Καθηγητής Ε.Μ.Π.

**Εργαστήριο Ηλεκτρονικής**
Αθήνα, Οκτώβριος 2015


Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Τομέας Επικοινωνιών, Ηλεκτρονικής & Συστημάτων Πληροφορικής


# Analysis and FPGA implementation of a Single-Bit Two-Step Look-Ahead ΔΣ Modulator

**Διπλωματική Εργασία**


Αθανάσιος Σ. Ορφανός

**Επιβλέπων Καθηγητής :**
Παύλος-Πέτρος Σωτηριάδης
Επίκ. Καθηγητής Ε.Μ.Π.


Εγκρίθηκε από την τριμελή επιτροπή την 15η Οκτωβρίου 2015

| *(Υπογραφή)* | *(Υπογραφή)* | *(Υπογραφή)* |
|---|---|---|
| ..... | ..... | ..... |
| Παύλος-Πέτρος Σωτηριάδης | Αθανάσιος Παναγόπουλος | Κιαμάλ Πεκμεστζή |
| Επίκ. Καθηγητής Ε.Μ.Π. | Επικ. Καθηγητής Ε.Μ.Π. | Καθηγητής Ε.Μ.Π. |


**Εργαστήριο Ηλεκτρονικής**
Αθήνα, Οκτώβριος 2015

*(Υπογραφή)*

.....
**Αθανάσιος Σ. Ορφανός**
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

# Περίληψη

Στην παρούσα διπλωματική εργασία μελετάμε έναν single bit Digital to Digital Converter (DD) . Το σήμα εισόδου του είναι ένα ψηφιακό σήμα, μια ακολουθία αποτελούμενη από πολλά bits ενώ στην έξοδο του το σήμα αυτό απεικονίζεται από μία ακολουθία των ενός μόλις bit, $\{+1, -1\}$. Το κέρδος απ αυτή τη διαμόρφωση του σήματος είναι στη γραμμικότητα που προσδίδει η έξοδος $\{+1, -1\}$, επιτρέποντας μας έτσι να παραλείψουμε τον multi-bit Digital to Analog Converter (DAC). Η ανάκτηση του αρχικού σήματος μπορεί να γίνει μόνο με ένα Lowpass Filter. Καθώς οι multi-bit DAC έχουν εξ ορισμού πολλά επίπεδα κβαντισμού, οι όποιες ατέλειες στο αναλογικό μέρος (πυκνωτές,αντιστάσεις) θα αλλοιώσουν την τιμή των επιπέδων εισαγάγοντας μη γραμμικά φαινόμενα και επιβαρύνοντας κατ΄ αυτόν τον τρόπο τον διαμορφωτή. Ωστόσο με μόλις δύο επίπεδα κβαντισμού το σφάλμα στην αναπαράσταση του σήματος θα είναι πολύ μεγάλο. Για την αντιστάθμιση του υψηλού σφάλματος απαιτείται πολύ υψηλή ταχύτητα δειγματοληψίας.

Για την υλοποίηση του ΔΣ διαμορφωτή δεν χρησιμοποιείται κάποια ευρέως διαδεδομένη τοπολογία αλλά ο Multi-Step Look-Ahead (MSLA) ΔΣ διαμορφωτής, μία νέα πρόταση του διδακτορικού φοιτητή Χάρη Μπασέτα και του επιβλέποντα καθηγητή κ. Σωτηριάδη. Η διαφορά με τον συμβατικό ΔΣ είναι ότι ο υπολογισμός της εξόδου $\{+1, -1\}$ προκύπτει απ την ελαχιστοποιήση μιας συναρτησης κόστους κοιτώντας βήματα μπροστά. Στην παρούσα εργασία εξετάζεται ο MSLA για δύο βήματα μπροστά. Επίσης παρουσιάζεται η βέλτιστη δομή για την εφαρμογή αυτού του διαμορφωτή καθώς επίσης γίνεται και μια εκτενής ανάλυση για τον υπολογισμό του αριθμού των bits για κάθε σήμα ώστε η εφαρμογή του σε πραγματικό κύκλωμα να είναι αποδοτική. Τέλος υλοποιήσαμε τον συγκεκριμένο ΔΣ διαμορφωτή πάνω σε fpga και στον αναλυτή φάσματος είδαμε τα αποτέλεσματα.

## Λέξεις Κλειδιά

ΔΣ, ΣΔ, Διαμορφωτής, Υλοποίηση, Αριθμός δυφίων, Πλακέτα, Κύκλωμα

# Abstract

In this paper we study a single bit $\Delta\Sigma$ Digital to Digital Converter (DD). Its input is a digital signal, a sequence consisting of many bits while in its output this signal is depicted from a sequence of one just bit, $\{+1, -1\}$. The gain from this signal modulation is the given linearity from the output $\{+1, -1\}$, allowing us to forget the multi-bit Digital to Analog Converter (DAC). Recovery of the original signal can be done with just a lowpass filter. As multi-bit DAC by definition consists of many quantization levels, any imperfections in the analog part (capacitors, resistors) will affect the value of the levels entering nonlinear phenomena and attributing in this way the modulator. However with just two quantization levels, the error in the signal representation will be very large. To offset the high error is required very high sampling rate.

For the implementation of the $\Delta\Sigma$ modulator is not used any widespread topology but the Multi-Step Look-Ahead (MSLA) $\Delta\Sigma$ modulator, a new proposal by the PhD student Charis Basetas and the assistant prof. Paul-Peter Sotiriadi. The difference to conventional $\Delta\Sigma$ is that the calculation of output $\{+1, -1\}$ results from the minimization of a cost function looking steps ahead. This paper examines the MSLA for two steps ahead. Also shows the optimal architecture for the implementation of this modulator and also a comprehensive analysis is made for the calculation of the number of bits for each signal in order to its application in real circuit to be efficient. Finally we implemented the specific $\Delta\Sigma$ modulator on fpga and we observed the results in the spectrum analyzer.

## Key Words

$\Delta\Sigma$, $\Sigma\Delta$, Modulator, Multi Step Look Ahead, MSLA, Look Ahead, Implementation, Bits, Bit-width, FPGA , CIDF, Cascade Integrators Distributed Feedback

# Acknowledgement

I would like to thank my assistant prof. Paul-Peter Sotiriadis for our co-operation and for introducing me to his excellent laboratory team. I would also like to give special thanks to my supervisor, PhD Charis Basetas for his continuous and crucial support throughout this year; my thesis is just a part of his survey and I feel truly thankful working together with such a brilliant scientist.

# Contents

# Chapter 1

# Introduction

Nowadays the tendency is to digitalize each and every analog part of a system or an analog signal. The simplicity in processing a digital signal as well as the reduced noise renders them preferable compared to an analog one. It is not strange that big effort is made to restrict every analog part of a system resulting in the all digital systems. Year by year, the speed and density of digital integrated circuits (ICs) is increased, enhancing the dominance of digital methods in almost all areas of communications and consumer products. Therefore always there will be the need for converting the analog signal in digitalized form and consequently data converters are necessary. As the speed and capability of Digital Signal Processing (DSP) cores increases, so too must the speed and accuracy of the converters associated with them.
In fig. 1.1 is depicted the block diagram of a signal processing. The analog input is driven to an Analog to Digital Converter (ADC), next is inserted in the DSP core that is the engine that undertook the processing of the signal and finally through an ADC the analog form of signal is recovered. However if a single bit Digital to Digital Converted (DDC) replaces the DAC then just a LPF is adequate to recover the analog signal. A single bit DDC is preferable compared to the DAC since the latter due to analog mismatches may introduce no negligible non-linear effects.

In Chapter 2 initially it will be examined some popular topologies of Modulators. Next it is analysed the noise shaping of Modulators and the usefulness of Oversampling ratio in the effective operation of them.

In Chapter 3, the linearised model of $\Delta\Sigma$ Modulators for a better stability examination, is presented. Finally some non-linear phenomenons are analysed in depth.

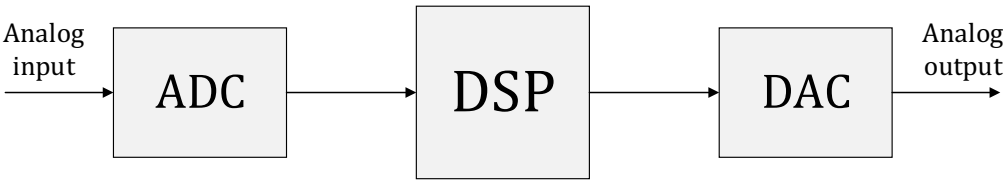



Figure 1.1: Digital Signal Processing

In Chapter 4 is given the analytical equations of MSLA Modulator for step equals two. After that we compute our Modulator i.e. we define its transfer functions. Finally a comparison with conventional $\Delta\Sigma$ modulators is presented.

In Chapter 5 is given the architecture of our $\Delta\Sigma$ modulator. Subsequently we compute the scale factors for each signal in fixed point representation and next the quantization of signals is made. Finally after calculating the signal to quantization noise, we determine the bit-width for each and every signal.

In Chapter 6 the results of FPGA implementation are represented as well as the code of verilog hardware language is given.

# Chapter 2

# Topologies of Digital to Digital Converter

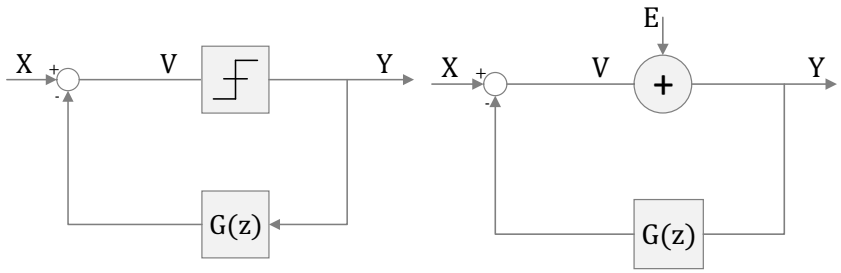
The analysis is related to DDC but almost the same results are hold for ADC and DAC. The difference between them is due to different form of input and ouput (analog/digital). Thus, the suitable transformation has to be taken place when one is fed back to the other.

## 2.1 Delta Modulator

Introduction to ADCs begin with an obsolete Modulator which is called Delta Modulator. The reference to that is made for historical reasons since it was one of the first ADC converters. A Delta Modulator is illustrated in fig. 2.1a. The presence of the quantizer is responsible renders the system non-linear which in turn results in difficulties in mathematical analysis. Simple qualitative understanding of its operation can, however, be gained by using a linearised model which is depicted in fig. 2.1b. The quantizer has been replaced with its input plus an error signal which is the difference between input and output.

$$V(z) = X(z) - G(z)Y(z) \tag{2.1}$$

$$Y(z) = V(z) + E(z) \tag{2.2}$$




(a) STF for every k

(b) NTF for different N

Figure 2.1: Transfer Functions

Substituting (2.1) into (2.2) it arises :

$$Y(z) = X(z) - G(z)Y(z) + E(z) \tag{2.3}$$

$$Y(z) = \frac{1}{1+G(z)}X(z) + \frac{1}{1+G(z)}E(z) \tag{2.4}$$

In the right-hand equation the factor of X(z) is the Signal Transfer Function (STF) since it filters only the input X(z) while the factor of E(z) is the Noise Transfer Function (NTF) as it filters only the error (or noise) E(z). Here $STF = NTF = 1/(1 + G(z))$. That means that filtering of input and noise for whole band of frequencies is the same.In turn it implies that it is impossible to attenuate NTF retaining the same time STF close to 1. Due to this important disadvantage Delta modulators have been abandoned. It is worth to stress that one of elementary roles of a converter is to pass the signal as much possible as unchanged. That is we are interested in a modulator with STF=1 and NTF=0 within the signal band of frequencies that operates.

## 2.2 $\Delta\Sigma$ modulator

### 2.2.1 Special Case

The new suggestion was to change the position of Loopfilter. As shown in fig. 2.2 the filter instead of lying in feedback path it moved in forward path resulting in $\Delta\Sigma$ Modulators. The latter is a concurrent and dominant converter nowadays.

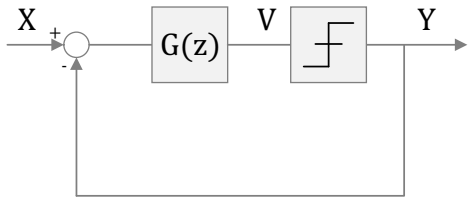



Figure 2.2: ΔΣ Modulator

Substituting the quantizer with an error source as in fig. 2.1b the described equations are given below :

$$V(z) = (X(z) - Y(z))G(z) \tag{2.5}$$

$$Y(z) = V(z) + E(z) \tag{2.6}$$

Substituting (2.5) into (2.6) it arises :

$$Y(z) = X(z)G(z) - Y(z)G(z) + E(z) \tag{2.7}$$

$$Y(z) = \frac{G(z)}{1 + G(z)}X(z) + \frac{1}{1 + G(z)}E(z) \tag{2.8}$$

Now $STF = G(z)/(1 + G(z))$ and $NTF = 1/(1 + G(z))$. It is clear that STF and NTF are different. A suitable choice of G(z) can give desirable characteristics in STF and NTF within the signal band. However this topology is just a case of the general structure of single bit ΔΣ modulator. It was presented so as to be clear the difference with Delta modulator.

### 2.2.2 General Case

The general structure of high order single bit ΔΣ modulator is illustrated in fig. 2.3. Our goal is to express the output of quantizer Y as a function of input X and its error E in order to specify the STF and NTF. So we have a system with two inputs (X,V) and one output Y.

$$V = L_0X + L_1Y \tag{2.9}$$

Also applying the linearised model :

$$Y = V + E \tag{2.10}$$

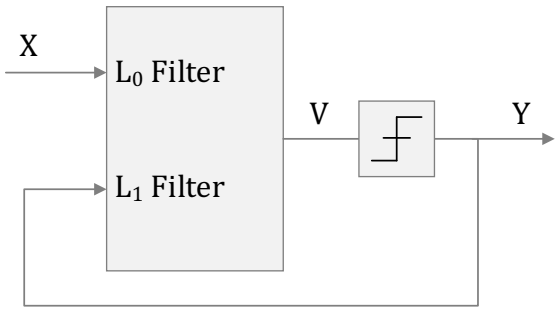



Figure 2.3: ΔΣ Modulator

,it is obtained :

$$Y = L_0X + L_1Y + E \tag{2.11}$$

$$Y = \frac{L_0}{1 - L1}X + \frac{1}{1 - L1}E \tag{2.12}$$

For $L_0 = L_1 = G$ result in the special case that examined above. Back in general case we see that it gives the absolute freedom in STF and NTF choice.

### 2.2.3 Error Feedback ΔΣ Modulator

A very interesting topology is the 1-bit SD error Feedback modulator. It s shown in fig. 2.4 .

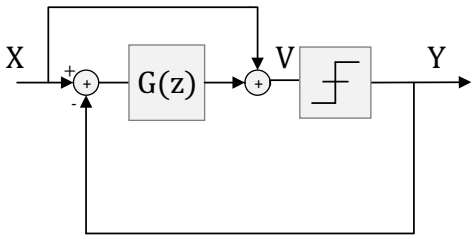



Figure 2.4: Error Feedback

Applying the superposition theorem the input X is filtered by 1+G(z) and the output of Quantizer Y is filtered by -G(z). Hence $L_0 = 1 + G(z)$ and $L_1 = -G(z)$. Having these filters easily is verified that $STF = 1$ and

$NTF = 1/(1 + G(z))$. Hence

$$Y(z) = X + \frac{1}{1 + G(z)}E \tag{2.13}$$

As it can be seen STF=1 no matter is the choice of filter G(z). Thus, our thoughts are eliminated in specification of filter G(z) meaning NTF. This topology constitutes the basis for the introduction to 1-bit Multi-Step Look-Ahead (MSLA) ΔΣ Modulator in Chapter 4 which is a new suggestion of ΔΣ modulators.

## 2.3 Cascade ΔΣ Modulators (MASH)

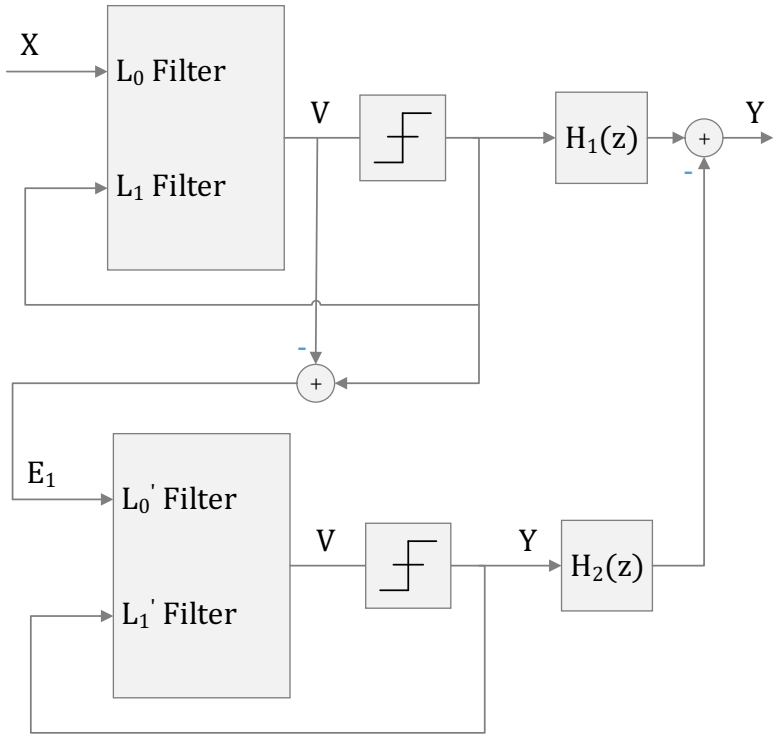



Figure 2.5: Multi Stage ΔΣ Modulator

Another interesting topology is the multi-stage-noise-shaping Modulator. It includes a lot of stages instead of just one. The MASH is depicted for two

stages, in fig. 2.5. The quantization error of the first stage is inserted in the second stage and it outputs a quantized estimation of the error. That output is filtered and added to the filtered output of the first ΔΣ Modulator in order to eliminate the quantization error. The equation for the first ΔΣ modulator is :

$$Y_1(z) = STF_1(z)X(z) + NTF_1(z)E_1(z) \tag{2.14}$$

and for the second one is :

$$Y_2(z) = STF_2(z)E_1(z) + NTF_2(z)E_2(z) \tag{2.15}$$

The output is formed as (z variable is omitted for presentation reasons) :

$$Y = H_1Y_1 - H_2Y_2 \tag{2.16}$$

$$Y = H_1STF_1X + (H_1NTF_1 - H_2STF_2)E_1 - H_2NTF_2E_2 \tag{2.17}$$

Consequently the error $E_1$ can be cancelled if $H_1NTF_1 - H_2STF_2 = 0$. A familiar choice of filters is $H_1 = STF_2$ and $H_2 = NTF_1$. Then output Y is gained as :

$$Y = STF_2STF_1X - NTF_1NTF_2E_2 \tag{2.18}$$

$$Y = STF \cdot X - NTF \cdot E_2 \tag{2.19}$$

where $STF = STF_2STF_1$ and $NTF = NTF_1NTF_2$.

If $STF_2, STF_1$ is just a delay then $|STF| = 1$ while if the $NTF_1, NTF_2$ is of the form $(1 - z^{-1})^{N_i}$ ,i=1,2 (where $N_1$,$N_2$ the order of $NTF_1, NTF_2$ respectively) then $NTF = (1 - z^{-1})^{N_1+N_2}$. As it will be seen in the section 2.4 that is desirable because the noise is shifted more in high frequencies namely a better noise shaping is succeeded (fig. 2.6b).

## 2.4 Noise Shaping and Oversampling

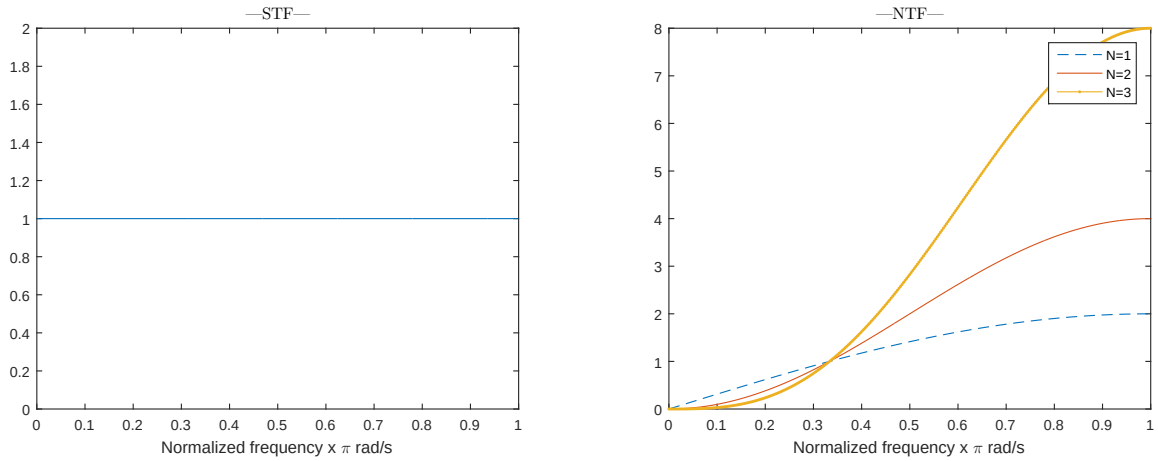
It should be noted that ΔΣ modulators are working well in the ranges of frequencies where the absolute value of STF is near 1 and NTF is close to 0. It can be succeeded by a good noise shaping namely shifting the noise out of the interested band, keeping it low within the band and high out of that. The interested band characterize the modulators as lowpass or bandpass when that band is a low or a zone band of frequencies respectively. Without loss of

generality it will be considered that modulators operate in low frequencies. Then possible choices and widely known classes are :

$$STF_1 = z^{-k} \tag{2.20}$$

$$NTF_1 = (1 - z^{-1})^N \tag{2.21}$$

where N is the order of filter $L_1$ and N+k the order of filter $L_0$.
Taking into account the transformation from Laplace to Z domain $z = e^{-j\Omega}$, $z = 1$ for $\Omega = 0$. Hence in low frequencies indeed $|STF| = 1$ and $|NTF| = 0$ ($|STF| = 1$ in each and every frequency). In fig. 2.6a,2.6b are plotted 2 diagrams. The first one depict the absolute value of STF which is 1 for every k and the second one depict the absolute values of NTF for N=1,2,3 both in function of frequency $\Omega$. For higher order filter the noise is suppressed more in low frequencies and it is shifted in regions out of the band that $\Delta\Sigma$ modulator operates.




(a) STF for every k

(b) NTF for N=1,2,3

Figure 2.6: Transfer Functions

It will be examined now in what way oversampling affects positively the lowpass modulators. In fig. 2.6a,2.6b, (2.20) and (2.21) are plotted in function of frequency $\Omega$. It is clear the closer to 0 for $\Omega$ the closer to 1 for STF and closer to 0 for NTF. Assuming that $f_b$ is the frequency bandwidth of input and $f_s$ is the sampling frequency, OverSampling Ratio (OSR) is defined

as follow :

$$OSR = \frac{fs}{2fb} \tag{2.22}$$

Assuming that input analog signal x with amplitude A has frequency $f_b$ , the sampled signal $x_s$ will have frequency $f_b/f_s$. Writing it by equations we have:

$$\begin{aligned} x &= Asin(2\pi f_b t) \\ x_s &= Asin\left(2\pi \frac{f_b}{f_s} n\right) = Asin\left(\frac{\pi}{OSR} n\right) \end{aligned} \tag{2.23}$$

Therefore the frequency of sampled signal is given as :

$$\Omega = \frac{\pi}{OSR} \tag{2.24}$$

Obviously the frequency of sampled signal depends exclusively on OSR. Increased OSR is equivalent to lower $\Omega$ and hence better behaviour of STF and NTF. It's worth to note however that if $f_b$ is high then it is hard to retain a high OSR since the sampling frequency $f_s$ has to be increased proportionally (see (2.22)).

# Chapter 3

# Modeling of Single-Bit Single-Stage ΔΣ Modulators

## 3.1 Linear Modeling of ΔΣ Modulators

In Chapter 2 for our analysis the quantizer was represented as its input and an error source. For the latter a white noise approximation was given. Unfortunately, the quantizers used in ΔΣ modulators hardly ever match the requirements to validate that approximation:
• The number of quantization levels is usually very small.
• The quantizer can be in overload.
• The input signal of the quantizer is generally not a random (white) signal.
Although adding a dither signal could satisfy the latter requirement, the low resolution of the quantizers renders the model invalid. A one-bit quantizer in particular cannot be modeled by the addition of independent noise. A technique to study more accurately the behaviour of a ΔΣ modulator about stability is to insert a gain factor as depicted in fig. 3.1.

The equation for NTF is now :

$$NTF = \frac{1}{1 - kL1} \tag{3.1}$$

If the quantization levels of the quantizer are more than two, the gain (k) of this can be determined uniquely as depicted in fig. 3.2a. The reason is that a certain gain (k) choice offers the least divergence between quantizer input and output, giving the minimized error in aforementioned linear model and restricting that way the non-linear effect due to the quantizer. In other

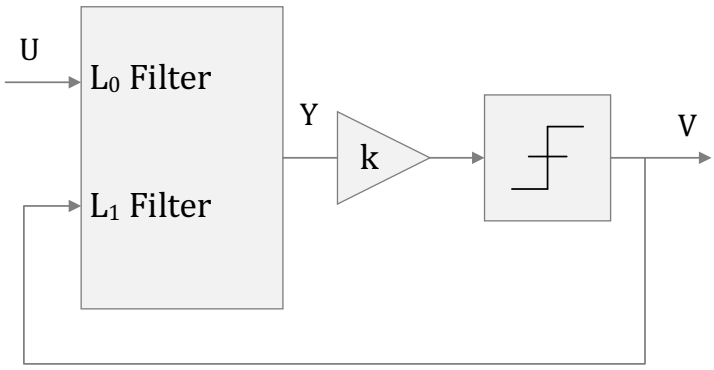
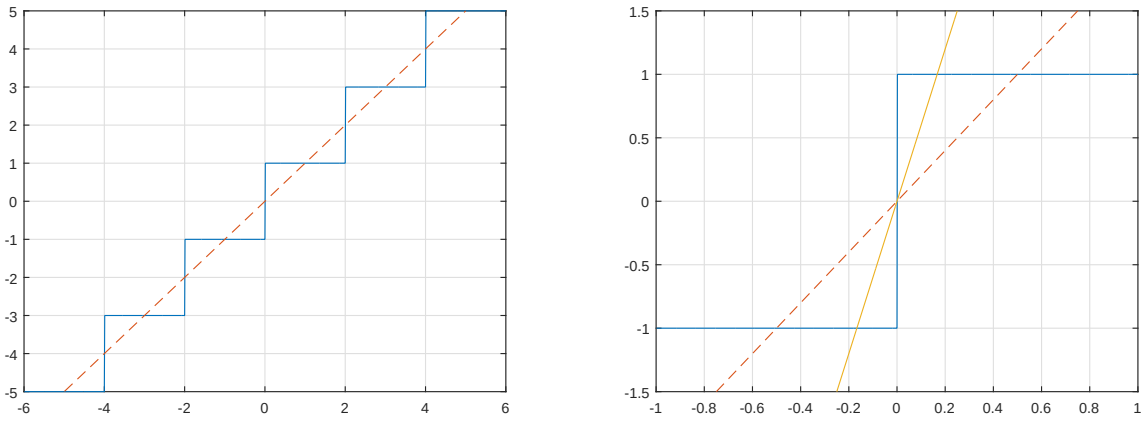



Figure 3.1: Linear Model

words the linear model is more accurate for a specific value of the gain k. Notice merely that this gain is embedded in the quantizer operation meaning when a quantization is happening the gain factor implicitly is applied. In the other hand when there are only two quantization levels i.e. the quantizer output is single bit (fig. 3.1) ,the gain cannot be determined directly. The error minimization depend on the input type into the quantizer. Therefore for different inputs an alternate k is chosen so as the non-linear effect (the error sequence) to be restricted. Two lines are plotted in fig. 3.2b, showing the gain choice is not readily obtained.




(a) Easy computation of k

(b) k is dependent on input type

Figure 3.2: filters

The computation task for the gain is given right now. If the statistical averages indices of y are known, an optimality criterion for k is to minimize the mean square value (namely the average power) of the error sequence e. This is defined as the expected (or mean) value of $e^2$:

$$y[n] = \lim_{N\to\infty} \left( \frac{1}{N} \sum_{n=0}^{N} a(n)b(n) \right) \tag{3.2}$$

Since $e = v - ky$ , the average power of e can be written as

$$\sigma_e^2 = (e, e) = (v - ky, v - ky) = (v, v) - 2k(v, y) + k^2(y, y) \tag{3.3}$$

Taking the partial derivative, k for minimizing the variance is obtained

$$ds/dk = 0 \implies -2(v, y) + 2k(y, y) = 0 \implies k = \frac{(v, y)}{(y, y)} = \frac{E\{|y|\}}{E\{y^2\}} \tag{3.4}$$

since the output of quantizer is the sign of $y(n)$ it follows $v(n)y(n) = |y(n)|$.

A more sophisticate and complicate model for calculating the quantization noise was suggested by Ardalan and Paulos. They introduced a separate gain for the signal and the quantization noise. Although fitting a more elaborate model to the non-linear quantizer can result in more accurate predictions, it does not give significantly more insight in the behaviour of the single-bit $\Delta\Sigma$.

## 3.2 Stability

It is clear that in the feedback path only $L_1$ exists meaning it is nearly the exclusive factor in stability issues. Once $NTF = 1/(1 - kL_1)$ the same holds for NTF. Because of the non-linear behaviour and nature of $\Delta\Sigma$ modulator an accurate criterion to predict the stability characteristics do not exist. The linear models are merely ways to approach and study a non-linear system. Nevertheless a lot of methods in order to ensure stability have been obtained. The most widely-used criterion is that of Lee which says :

A single-bit modulator is likely to be stable if $max(NTF) < c$. Lee's authentic criterion says that $c = 2$ but an edited criterion says that c depends on the order of NTF and is decreased up to 1.4 as the order of modulator

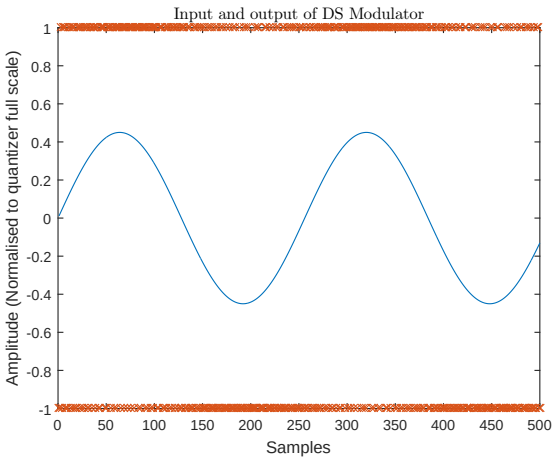
is increased. Another interesting criterion is about computing the mean-squared value of the magnitude response of the NTF (noise amplification factor) requiring :

$$A = \frac{1}{2\pi}|NTF(e^{j\theta})|^2 d\theta < 2.5 \tag{3.5}$$

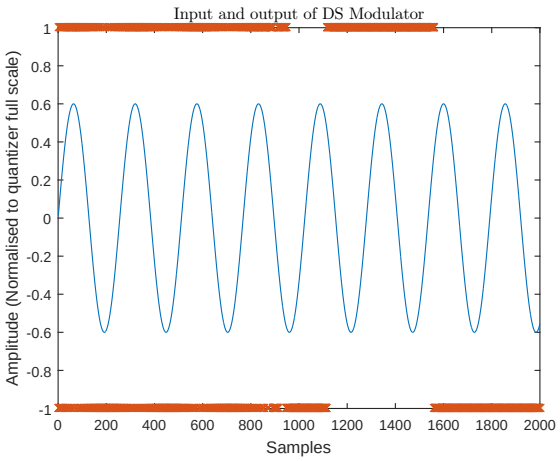
Note that these criteria are neither necessary nor sufficient. Nevertheless, due to its simplicity, they offer a quick and simple first view about modulator stability. For more accurate details extensive simulations in environments such as matlab are necessary.

It should be noted also, the out of band gain does not impact the efficiency of modulator (as aforementioned) but a high value in the out of band gain overload the quantizer forcing the system to be unstable. Provided that the out of band gain is considerable in high order filters (see fig. 2.6b), such filters cause instability and hence there is a restriction in choosing the order of a filter.

A $\Delta\Sigma$ Modulator in a stable operation is depicted in fig. 3.3a. The output density, $\{+1, -1\}$ is higher at the region around the peaks of the sinusoidal signal. However this is not occurred in fig. 3.3b meaning that $\Delta\Sigma$ modulator is in an unstable mode. The result is long sequences of consecutive +1 or -1, i.e. the output sequence no longer tracks the input signal due to the overloaded quantizer.




(a) Stable operation of $\Delta\Sigma$ Modulator




(b) Instability of the $\Delta\Sigma$ Modulator

Figure 3.3: filters

To overcome an unstable mode of $\Delta\Sigma$ modulator a stabilization technique could be applied. However in that case the Signal to Noise Ratio (SNR) is significantly decreased requiring concurrently a reasonable time period to return in the normal mode. Therefore the conclusion is that it is preferable to ensure that modulator will never become unstable rather than a stabilization technique to be applied.

## 3.3 Idle Patterns, Dead Zones and Tones

The repetitive patterns that are founded in the output of the modulator under zero input signal constitute in substance a limit cycle, termed idling pattern. To obtain a view of this phenomenon assume a modulator which consists of just one accumulator for which holds that $L_0 = L_1 = z^{-1}/(1 - z^{-1})$. Lets $x_0$ where $0 < x_0 < 1$ be the initial state of the filter. Once the first input in the quantizer is $x_0$ its output will equal to 1. So its next input will be $x_0 - 1 < 0$ and therefore its next output will be $-1$. Its next input now will be $(x_0 - 1) + 1 = x_0$ and hence its output will be again 1 and so on. Thus the sequence 1,-1,1,-1,.. takes place with frequency $f_s/2$ since every second sample there is change in sequence. Idle pattern itself is not a problem since we supposed a zero input. The problem is occurred when input changes to a non-zero type and the idle pattern cannot be broken. This is happening for small input amplitudes which is called that belong in the dead zone i.e. a zone of inputs that modulator cannot decode. As an example consider a sinusoidal input $u[n] = Asin(2\pi\frac{f_b}{f_s}n)$ where $f_b$ is the frequency bandwidth of input signal and $f_s = 1/T_s$ is the frequency sampling. For numerous samples, $m \gg 1$ and for high OSR, $f_s \gg f_b$ the following approximation can be done :

$$\sum_{n=0}^{m} u[n] \approx T_s \int_{n=0}^{mT_s} Asin(2\pi f_b t)dt = \frac{Af_s}{2\pi f_b}\left[cos(2\pi f_b T_s m) - 1\right] \tag{3.6}$$

Clearly the maximum value is

$$sum\_max = \frac{Af_s}{2\pi f_b} > 0 \tag{3.7}$$

Therefore in order to cancel idle pattern one of 2 following cases must hold :

$$x_0 + sum\_max < 0 \implies sum\_max < -x_0 \quad or \tag{3.8}$$

$$(x_0 - 1) + sum\_max > 0 \implies sum\_max > (1 - x_0) \tag{3.9}$$

Remember that initially $x_0 > 0$ and $(x_0 - 1) < 0$. So the aforementioned equations ensure that idle pattern is broken avoiding in that way the dead zone. Provided that $sum\_max, x_0 > 0$ only (3.9) holds and the worst case is for $x_0 = 0$. From (3.7),(3.9) for the worst case it arises :

$$\frac{Af_s}{2\pi f_b} > 1 \implies A > \frac{2\pi f_b}{f_s} = \frac{\pi}{OSR} \tag{3.10}$$

The equation gives the minimum value of sinusoidal amplitude so as a limit cycle to be avoided :

$$A_{min} = \frac{2\pi f_b}{f_s} \tag{3.11}$$

Notice that a higher $f_b$ demands a higher amplitude $A_{min}$ to overcome the dead-zone. Consequently dead zone effect is more intense in high frequencies within the signal band. A similar procedure should be followed for higher order loop filters. The higher gain in the loop filter gives the ability the modulator to ″recognise″ smaller signals restricting as much the dead-zone as idle pattern to be considered an ignored and unimportant effect. The appearance of an idle pattern is translated into tones in frequency domain and can easily be detected from the human ear or eye in an audio or video application.

# Chapter 4

# Single-bit Two-Step Look-Ahead Modulator

## 4.1 Analytical Equations of 2-SLA ΣΔ Modulator

To retain our analysis simple we will examine the Multi-Step Look-Ahead (MSLA) for step k=2. For the remainder of this assignment it will be mentioned as 2-Step Look-Ahead (2-SLA) Modulator. However, the general proof is given in [1]. Moreover the system that examined is about a step k=2, hence more depth in analysis may confuse the reader. It has been proved that the optimal 1-bit output sequence for a given input sequence and filter is obtained by Viterbi decoding the input sequence over the convolutional code generated by the filter. Yet, Viterbi decoding becomes exponentially complex with the length of the input sequence. It has been also proved that the error-feedback ΔΣ modulator produces the optimal 1-bit output sequence when a first-order loop filter is used. Therefore, this ΔΣ modulator topology forms the basis for the development of the MSLA modulator. The general form of G(z) is

$$G(z) = \frac{\sum_{i=1}^{l} b_i z^{-i}}{1 + \sum_{i=1}^{m} a_i z^{-m}} \tag{4.1}$$

where $l, m$ are the orders of the numerator and the denominator respectively.

Here for the purposes of our system and for reasons of simplicity we

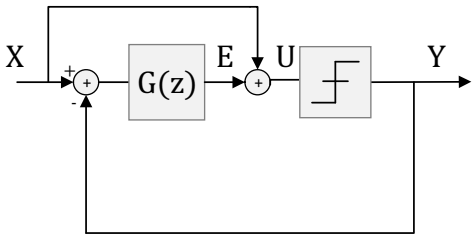



Figure 4.1: Error Feedback as an optimization problem

eliminate the order of G(z) to 3. From inspection of fig. 4.1 :

$$G(z) = \frac{E(z)}{X(z) - Y(z)} \tag{4.2}$$

where X(z), Y(z), E(z) and G(z) are the z-transforms of the input, output, filter output and loop filter impulse response sequences respectively.
From (4.1) and (4.2) with $l = m = 3$ it arises :

$$\frac{E(z)}{X(z) - Y(z)} = \frac{b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} \tag{4.3}$$

Hence in time domain easily verified that :

$$\begin{aligned} e[n] + a_1 e[n-1] + a_2 e[n-2] + a_3 e[n-3] = b_1(x[n-1] - y[n-1]) \\ + b_2(x[n-2] - y[n-2]) + b_3(x[n-3] - y[n-3]) \end{aligned} \tag{4.4}$$

This equation will be useful next.
The topology now could be seen as an optimization problem. It is known that the output is single bit. It means that when the quantizer input (x[n]+e[n]) is positive, the output is 1 and if it is not positive, the output is -1. Consequently every time the negative value of the possible output, $-v$ tends to bring the sum (x[n]+e[n]) toward zero. By equations it means that considering the cost function

$$S = |x[n] + e[n] - v| \tag{4.5}$$

the role of $v = \pm 1$ is to minimize it. The choice of that $v$ is the output $y[n]$. Therefore :

$$y[n] = \arg\min_{v \in \pm 1} S \tag{4.6}$$

From 4.6 it is concluded that the output of the error-feedback $\Delta\Sigma$ modulator is determined by the instantaneous minimization of the quantization error within the bandwidth of the loop filter. The optimal solution would be the minimization of the total time-domain quantization error shaped by the loop filter. The latter would require Viterbi decoding and is impossible for an infinite input sequence. A compromise between the optimal solution and hardware complexity is possible if the minimization of the quantization error is restricted to the current and the next 2 (here) future input samples.

$$S_0 = |x[n-2] + e[n-2] - v_0| \tag{4.7}$$

$$S_1 = |x[n-1] + e[n-1] - v_1| \tag{4.8}$$

$$S_2 = |x[n] + e[n] - v_2| \tag{4.9}$$

$$y[n-2] = \arg\min_{v_0 \in \pm 1} \left[ \min_{v_1, v_2 \in \pm 1} (S_0 + S_1 + S_2) \right] (5) \tag{4.10}$$

Thus the calculation of $y[n-2]$ requires the knowledge of $e[n-2], e[n-1], e[n]$. The latter can be expressed in function of previous values if in (4.4) the suitable timing shift is applied. In addition we must note that the outputs $y[n-2], y[n-1]$ are expressed as $v_0, v_1$ respectively since they have not be defined yet.

$$\begin{aligned} e[n-2] = {} & b_1(x[n-3] - y[n-3]) + b_2(x[n-4] - y[n-4]) \\ & + b_3(x[n-5] - y[n-5]) - a_1 e[n-3] - a_2 e[n-4] - a_3 e[n-5] \end{aligned} \tag{4.11}$$

$$\begin{aligned} e[n-1] = {} & b_1(x[n-2] - v_0) + b_2(x[n-3] - y[n-3]) \\ & + b_3(x[n-4] - y[n-4]) - a_1 e[n-2] - a_2 e[n-3] - a_3 e[n-4] \end{aligned} \tag{4.12}$$

$$\begin{aligned} e[n] = {} & b_1(x[n-1] - v_1) + b_2(x[n-2] - v_0) \\ & + b_3(x[n-3] - y[n-3]) - a_1 e[n-1] - a_2 e[n-2] - a_3 e[n-3] \end{aligned} \tag{4.13}$$

Extensive simulations have shown that keeping the last cost function term of the sum in 4.10 ($S_2$) does not typically impact performance. In fact, in many cases it offers superior stability than calculating the output using

all terms, while fewer terms lead to lower hardware complexity. Thus, the following simplification of the decision rule is proposed.

$$y[n] = \arg\min_{v_0 \in \pm 1} \left( \min_{v_1, v_2 \in \pm 1} S_2 \right) \tag{4.14}$$

Substituting the (4.13) into (4.9) we have :

$$\begin{aligned} S_2 = |x[n] + b_1 x[n-1] + b_2 x[n-2] + b_3 (x[n-3] - y[n-3]) \\ - a_1 e[n-1] - a_2 e[n-2] - a_3 e[n-3] - b_2 v_0 - b_1 v_1 - v_2| \end{aligned} \tag{4.15}$$

Our goal is to write $S_2$ as linear combination of $v_0, v_1, v_2$. So $e[n-1]$ from (4.12) has to be substituted in (4.15) since the first one contains the variable $v_0$. The (4.11) is useless since this does not contain some of minimized variable $v_0, v_1, v_2$.

$$\begin{aligned} S_2 = |x[n] + b_1 x[n-1] + (b_2 - a_1 b_1) x[n-2] + (b_3 - a_1 b_2) x[n-3] - a_1 b_3 x[n-4] \\ - (b_3 - a_1 b_2) y[n-3] + a_1 b_3 y[n-4] \\ + (a_1 a_1 - a_2) e[n-2] + (a_1 a_2 - a_3) e[n-3] + a_1 a_3 e[n-4] \\ - (b_2 - a_1 b_1) v_0 - b_1 v_1 - v_2| \end{aligned} \tag{4.16}$$

Defining the signal u[n] as

$$\begin{aligned} u[n] = x[n] + b_1 x[n-1] + (b_2 - a_1 b_1) x[n-2] + (b_3 - a_1 b_2) x[n-3] - a_1 b_3 x[n-4] \\ - (b_3 - a_1 b_2) y[n-3] + a_1 b_3 y[n-4] \\ + (a_1 a_1 - a_2) e[n-2] + (a_1 a_2 - a_3) e[n-3] + a_1 a_3 e[n-4] \end{aligned} \tag{4.17}$$

the (4.16) can be written as

$$S_2 = |u[n] - (b_2 - a1 b_1) v_0 - b_1 v_1 - v_2| \tag{4.18}$$

Well now it looks similar to the cost function in (4.5) (namely the Error-Feedback modulator) with the minimization variables to be 3 instead of 1 i.e. $v_0, v_1, v_2$.

Taking the z-Transform of (4.17) it gives :

$$\begin{aligned} U(z) = (1 + b_1 z^{-1} + (b_2 - a_1 b_1) z^{-2} + (b_3 - a_1 b_2) z^{-3} - a_1 b_3 z^{-4}) X(z) \\ + (-(b_3 - a_1 b_2) z^{-3} + a_1 b_3 z^{-4}) Y(z) + ((a_1 a_1 - a_2) z^{-2} + (a_1 a_2 - a_3) z^{-3} + a_1 a_3 z^{-4}) E(z) \end{aligned} \tag{4.19}$$

Substituting the output filter, E(z) from (4.3) into the (4.19) we have

$$U(z) = (1 + b_1 z^{-1} + (b_2 - a_1 b_1) z^{-2} + (b_3 - a_1 b_2) z^{-3} - a_1 b_3 z^{-4}) X(z) + (-(b_3 - a_1 b_2) z^{-3} + a_1 b_3 z^{-4}) Y(z) + ((a_1 a_1 - a_2) z^{-2} + (a_1 a_2 - a_3) z^{-3} + a_1 a_3 z^{-4}) \frac{b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} (X(z) - Y(z)) \tag{4.20}$$

Setting

$$A = ((a_1 a_1 - a_2) z^{-2} + (a_1 a_2 - a_3) z^{-3} + a_1 a_3 z^{-4}) \frac{b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} \tag{4.21}$$

we have :

$$U(z) = (1 + b_1 z^{-1} + (b_2 - a_1 b_1) z^{-2} + (b_3 - a_1 b_2) z^{-3} - a_1 b_3 z^{-4} + A) X(z) + (-(b_3 - a_1 b_2) z^{-3} + a_1 b_3 z^{-4} - A) Y(z) \tag{4.22}$$

After the trivial algebraic calculations the final equation is simplified. Our result is about the signal y[n-2], thus replacing $Y(z)$ with $z^2 Y(z)$ the final result related to signal y[n] is obtained :

$$U(z) = \frac{1 + (a_1 + b_1) z^{-1} + (a_2 + b_2) z^{-2} + (a_3 + b_2 a_1 + b_3 - a_2 b_1) z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} X(z) - \frac{(a_1^2 b_1 - a_2 b_1 - a_1 b_2 + b_3) z^{-1} + (a_1 a_2 b_1 - a_2 b_2 - a_3 b_1) z^{-2} + (a_1 a_3 b_1 - a_3 b_2) z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} Y(z) \tag{4.23}$$

## 4.2 Computation of our Examined 2-SLA Modulator

Choosing as NTF

$$NTF = (1 - z^{-1})^3 \tag{4.24}$$

G(z) is readily calculated :

$$G(z) = \frac{1 - NTF}{NTF} = \frac{3z^{-1} - 3z^{-2} + z^{-3}}{1 - 3z^{-1} + 3z^{-2} - z^{-3}} \tag{4.25}$$

Hence comparing with the general form of G(z) in (4.1) the factors are obtained

$$b_1 = 3, b_2 = -3, b_3 = 1 \tag{4.26}$$

$$a_1 = -3, a_2 = 3, a_3 = -1 \tag{4.27}$$

Substituting these values into the general form of 2-SLA in (4.23) our examined 2-SLA $\Delta\Sigma$ Modulator is obtained :

$$L_0(z) = \frac{1}{(1 - z^{-1})^3} \tag{4.28}$$

$$L_1(z) = -\frac{10z^{-1} - 15z^{-2} + 6z^{-3}}{(1 - z^{-1})^3} \tag{4.29}$$

and

$$STF' = \frac{z^{-3}}{1 + 7z^{-1} - 12z^{-2} + 5z^{-3}} \tag{4.30}$$

$$NTF' = \frac{(1 - z^{-1})^3}{1 + 7z^{-1} - 12z^{-2} + 5z^{-3}} \tag{4.31}$$

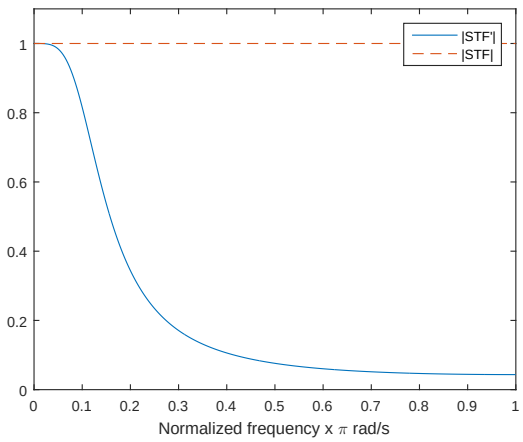
In fig.4.2 is plotted the absolute values of our $STF'$ and $NTF'$ in comparison with the familiar class of $STF = z^{-k}$ where k is depend on the order of the filter $L_0$, and $NTF = (1 - z^{-1})^3$. We observe a better behaviour in the very low frequencies for $|NTF'|$ compared to that of $|NTF|$ since the former is closer to 0.

## 4.3 The Effect of Look-Ahead Steps in Stability

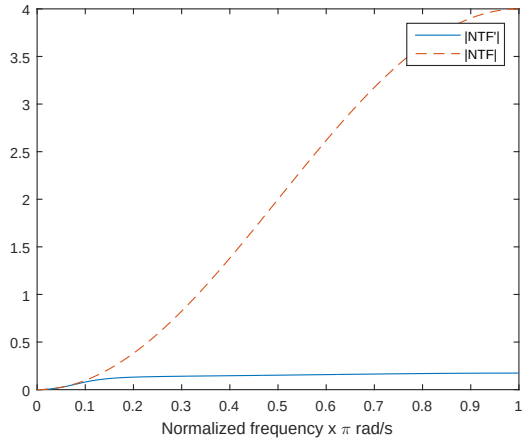
In (4.10) it was found the decision rule for 2-SLA modulator. For MSLA with the number of steps to be k the decision rule is generalized :

$$y[n] = \arg\min_{v_0 \in \pm 1} \left[ \min_{v_1, v_2, \ldots, v_k \in \pm 1} \sum_{j=0}^{k} S_j \right] \tag{4.32}$$

The performance of MSLA modulators has been verified using extensive MATLAB simulations. It has been observed that in many cases a better




(a) Comparison of Signal Transfer Functions




(b) Comparison of Noise Transfer Functions

Figure 4.2: Transfer Functions

output dynamic range may be obtained when the last or the last two metrics $S_{k,n}$ and $S_{k-1,n}$ in (4.32) are taken into account for the computation of the output, thus greatly simplifying the hardware complexity at the same time. Several different loop filters and associated NTFs have been simulated and results have been obtained, which demonstrate the improvement of the stability characteristics of the modulator as the number of look-ahead steps $k$ is increasing. The maximum sinusoidal input amplitude resulting in stable operation of the modulator is chosen as the stability measure.

NTFs which are stable when employed in a conventional $\Delta\Sigma$ modulator, remain stable for higher input amplitudes with higher values of $k$, while unstable NTFs may offer stable operation when $k$ is increased.

The attained results are shown in Fig. 4.3, while the corresponding filters are shown in Table 4.1. The stability was determined by MATLAB simulations of $2 \cdot 10^6$ input samples. For the low-pass filters 1-4 the signal $x_n = A \sin(2\pi \cdot 0.0041482n)$ was used as the input, whereas for the band-pass filter 5 the input signal was in the form $x_n = A \sin(2\pi \cdot 0.365n)$. The effect of the look-ahead steps $k$ in the increase of the maximum stable sinusoidal input amplitude is evident. Conclusively, the MSLA modulator offers higher input dynamic range than a conventional $\Delta\Sigma$ modulator with the same NTF.

The main consideration for the selection of the NTF poles are the stability requirements of the conventional $\Delta\Sigma$ modulator. The NTFs 4-5 do not have

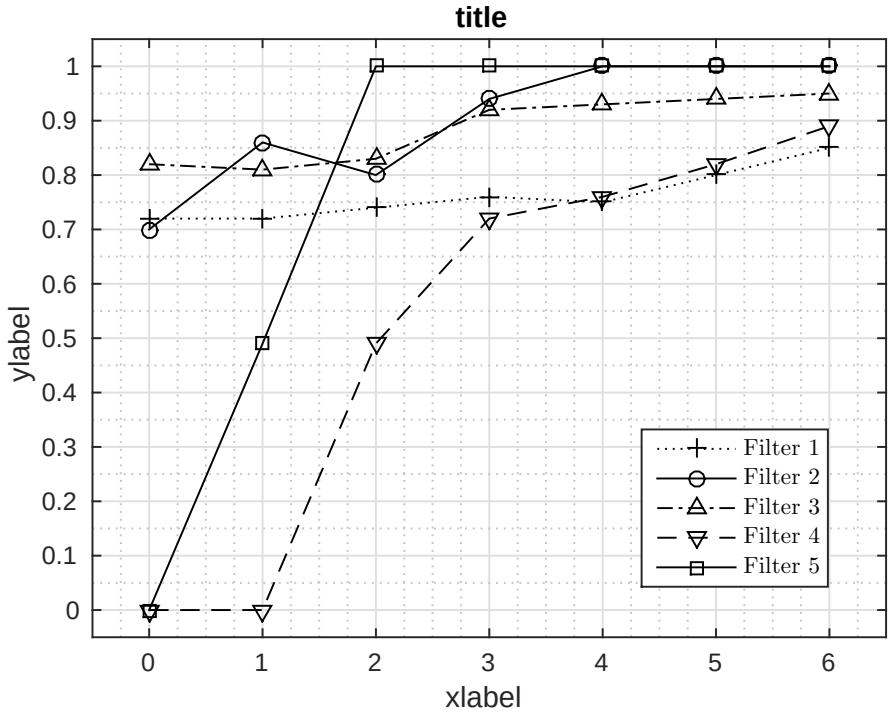



Figure 4.3: Maximum stable sinusoidal input amplitude for different look-ahead steps $k$ and NTFs (shown in Table 4.1)

any poles for the reduction of the out-of-band gain of the corresponding loop filters and are unstable when used in a conventional $\Delta\Sigma$ modulator. The obtained simulation results demonstrate that such NTFs may be used in a MSLA modulator and offer stable operation for a wide range of input amplitudes when the value of $k$ is sufficiently large ($\geq 2$). Thus, the design space of the NTF is greatly increased allowing for the selection of the zeros and poles of the NTF based on other design criteria apart from stability.

Table 4.1: Filter transfer functions of Fig. 4.3

| | **Transfer function** |
|---|---|
| **NTF 1** | $\frac{1-3.996z^{-1}+5.993z^{-2}-3.996z^{-3}+z^{-4}}{1-3.131z^{-1}+3.724z^{-2}-1.99z^{-3}+0.4029z^{-4}+3.5\cdot10^{-5}z^{-5}}$ |
| **NTF 2** | $\frac{(1-z^{-1})(1-1.994z^{-1}+z^{-2})}{(1-0.5995z^{-1})(1-1.384z^{-1}+0.5892z^{-2})}$ |
| **NTF 3** | $\frac{(1-z^{-1})(1-2z^{-1}+z^{-2})}{(1-0.6694z^{-1})(1-1.531z^{-1}+0.6639z^{-2})}$ |
| **NTF 4** | $(1-z^{-1})^3$ |
| **NTF 5** | $(1-\cos(2\pi\cdot0.365)z^{-1}+z^{-2})^2$ |

# Chapter 5

# Implementation Analysis of 2-SLA Modulator

## 5.1 Cascaded Integrator with Distributed Feedback

In this section we present the structure of our 2-SLA modulator . It is constructed by cascaded integrators with distributed feedback (CIDF). The schematic diagram is in fig. 5.1.

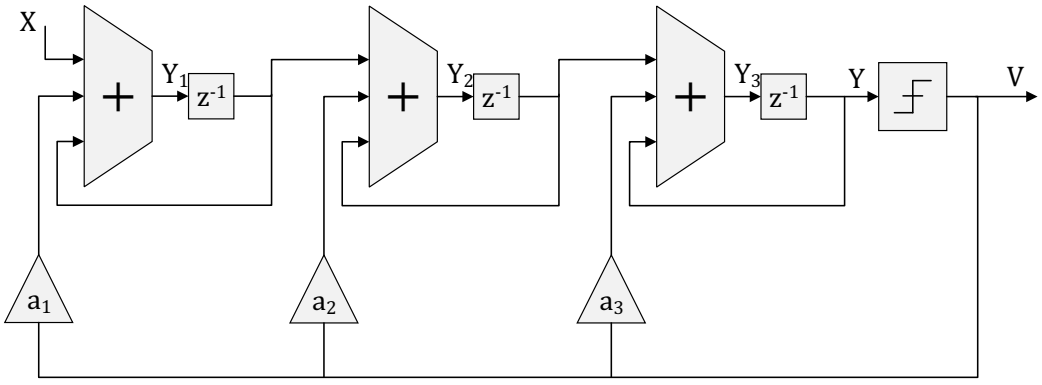



Figure 5.1: CIDF architecture for the construction of our 2-SLA Modulator

This architecture offers a lot of positive characteristics. As it is shown every integrator is related with just prior values and not older than them.

In other words each of them needs only one register for memory purposes. Furthermore no multiplications are required. Provided that $V = \pm 1$, the factors $a_1$,$a_2$,$a_3$ will be either a positive or a negative constant. That is an "if" statement for choosing the suitable value can be used.

By writing the equations it will be verified that filters with input X or V and output Y equal (4.28) and (4.29) respectively calculating at the same time the factors $a_1$,$a_2$,$a_3$.
Assuming V=0 and X the only one input the ratio $Y/V$ will be found.

$$Y_1 = X + Y_1 z^{-1} \tag{5.1}$$
$$Y_2 = Y_1 z^{-1} + Y_2 z^{-1} \tag{5.2}$$
$$Y_3 = Y_2 z^{-1} + Y_3 z^{-1} \tag{5.3}$$
$$Y = Y_3 z^{-1} \tag{5.4}$$

Solving each equation for $Y_1, Y_2, Y_3$ it arises :

$$Y_1 = \frac{X}{1 - z^{-1}} \tag{5.5}$$
$$Y_2 = \frac{Y_1 z^{-1}}{1 - z^{-1}} \tag{5.6}$$
$$Y_3 = \frac{Y_2 z^{-1}}{1 - z^{-1}} \tag{5.7}$$
$$Y = Y_3 z^{-1} \tag{5.8}$$

With successive substitutions from (5.8) toward (5.5) it arises :

$$Y = \frac{Y_2 z^{-2}}{1 - z^{-1}} = \frac{Y_1 z^{-3}}{(1 - z^{-1})^{-2}} = \frac{X z^{-3}}{(1 - z^{-1})^{-3}} \tag{5.9}$$

Therefore

$$\frac{Y}{X} = \frac{z^{-3}}{(1 - z^{-1})^{-3}} \tag{5.10}$$

As it is seen the the above filter equals $L_0(z)$ filter of (4.28). The factor $z^{-3}$ is just three delays and its existence does impact the filter behaviour.

Assuming now X=0 and V the only one input the ratio $Y/V$ will be found.

$$Y_1 = a_1 V + Y_1 z^{-1} \tag{5.11}$$
$$Y_2 = a_2 V + Y_1 z^{-1} + Y_2 z^{-1} \tag{5.12}$$
$$Y_3 = a_3 V + Y_2 z^{-1} + Y_3 z^{-1} \tag{5.13}$$
$$Y = Y_3 z^{-1} \tag{5.14}$$

Solving each equation for $Y_1(1-z^{-1}), Y_2(1-z^{-1}), Y_3(1-z^{-1})$ it arises :

$$Y_1(1-z^{-1}) = a_1 V \tag{5.15}$$
$$Y_2(1-z^{-1}) = a_2 V + Y_1 z^{-1} \tag{5.16}$$
$$Y_3(1-z^{-1}) = a_3 V + Y_2 z^{-1} \tag{5.17}$$
$$Y = Y_3 z^{-1} \tag{5.18}$$

Multiplying (5.18) with $(1-z^{-1})$ and combining this with (5.17) we have :

$$Y(1-z^{-1}) = a_3 V z^{-1} + Y_2 z^{-2} \tag{5.19}$$

Multiplying the above equation with $(1-z^{-1})$ and combining this with (5.16) we have :

$$Y(1-z^{-1})^2 = a_3 V z^{-1}(1-z^{-1}) + a_2 V z^{-2} + Y_1 z^{-3} \tag{5.20}$$

Finally we multiply the above equation with $(1-z^{-1})$ and combine this with (5.15) :

$$\begin{aligned} Y(1-z^{-1})^3 &= a_3 V z^{-1}(1-z^{-1})^2 + a_2 V z^{-2}(1-z^{-1}) + a_1 V z^{-3} \\ \frac{Y}{V} &= \frac{a_3 z^{-1}(1-z^{-1})^2 + a_2 z^{-2}(1-z^{-1}) + a_1 z^{-3}}{(1-z^{-1})^3} \\ \frac{Y}{V} &= \frac{a_3 z^{-1} + (a_2 - 2a_3) z^{-2} + (a_1 - a_2 + a_3) z^{-3}}{(1-z^{-1})^3} \end{aligned} \tag{5.21}$$

Comparing with $L_1(z)$ in (4.29), the factors $a_1$,$a_2$,$a_3$ are calculated :

$$\begin{aligned} a_3 &= -10 \\ a_2 - 2a_3 = 15 &\implies a_2 = -5 \\ a_1 - a_2 + a_3 = -6 &\implies a_1 = -1 \end{aligned} \tag{5.22}$$

## 5.2 Scaling Factor for the Fixed Point Representation

For the representation of signals the Fixed Point Representation (FPR) is used. Every signal in this representation is considered to be a fraction where a scale factor implicitly accompany it. If we multiply the fraction number with

the scale factor the actual value of the signal is obtained. In an implemented system now every signal is a stream of bits which represent the fraction number. The scale factor (which is almost always power of two) defines the range of signal while the number of bits of stream defines the precision for that scale factor. Caution must be given in the choice of the scale factor so as the whole range of a signal can be represented as well as not having a high value which results in lower precision. Let's represent now the 2-SLA of fig. 5.1 in FPR. A generalized solution would be to examine the maximum value of all signals, next choose the minimum scale factor which can represent that value and eventually applying in each and every signal that scale factor. The maximum value can be found inserting as input the Maximum Stable Amplitude input for a given frequency (MSA) i.e. the maximum input for which stability is guaranteed. However, a more efficient solution is to apply in each of three integrators a different scale factor. Finding the maximum value for every one the least scale factor is applied to the corresponding integrator. The widest range for every one arises from theirs outputs $y_1[n]$,$y_2[n]$,$y_3[n]$ (moreover the range of $y_1$ is smaller than $y_2$ and $y_2$ smaller than $y_3$). By observing the fig. 5.2a-5.2c, in the first integrator a scale factor $SF = 2^4$ is chosen, in the second integrator a $SF = 2^5$ is chosen while in the last one a $SF = 2^6$ is chosen.

Let's assume now an input to the modulator within the range $-1 < x < 1$ with a scale factor of $2^0$. Shifting it right 4 bits a scale factor of $s_1 = 2^4$ is obtained for all signals in modulator. Since the input to second stage has a scale factor of $2^4$, shifting it right 1 bit a scale factor of $s_2 = 2^5$ is obtained for all signals in second and third stage. Finally similarly shifting one bit the input to third stage a scale factor of $s_3 = 2^6$ is obtained for the third stage. In the fig. 5.3 the CIDF structure with shift operations is given. This solution is better because in every integrator the minimum scale factor is applied allowing in that way a higher precision. Notice that in feedback path the factors are affected (symbolised as $a_1^{'}, a_2^{'}, a_3^{'}$) due to shifts operations $s_1, s_2, s_3$ (namely gains) but it will be discussed in section 5.4.

## 5.3 Quantization of the Signals

No discussion made for the precision of signals. Lets quantize our system. In fig. 5.4 the blocks $Q_1, Q_2, Q_3$ are each of them a Quantizer which truncates the signal keeping only a certain number of bits for every one. As showed

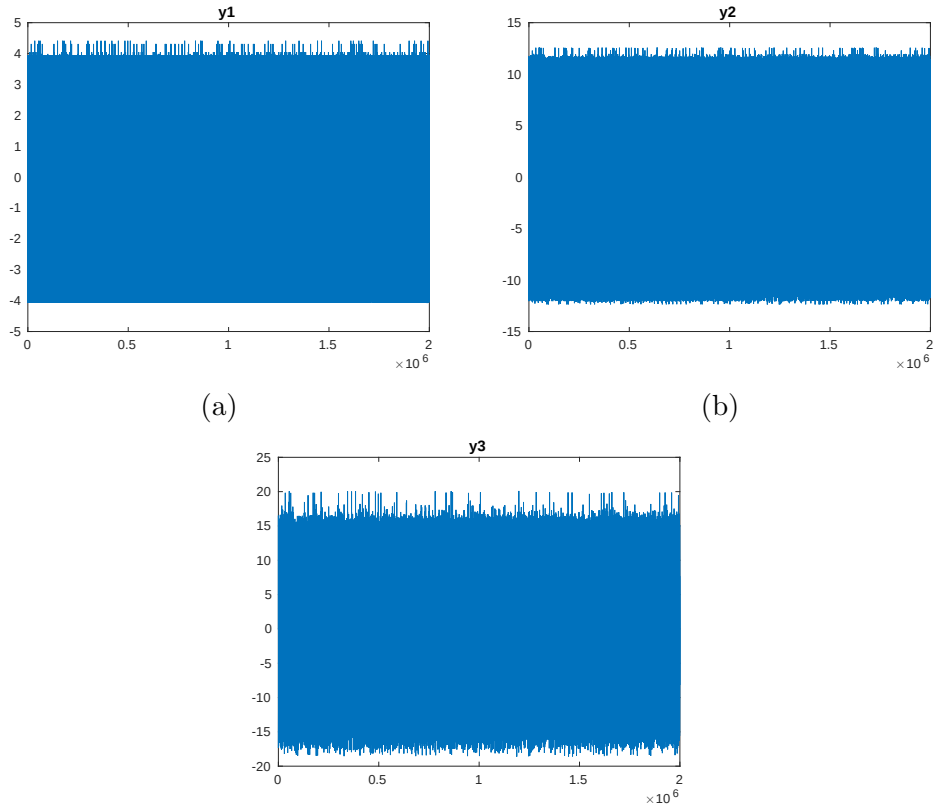



Figure 5.2: Maximum Amplitude of each integrator

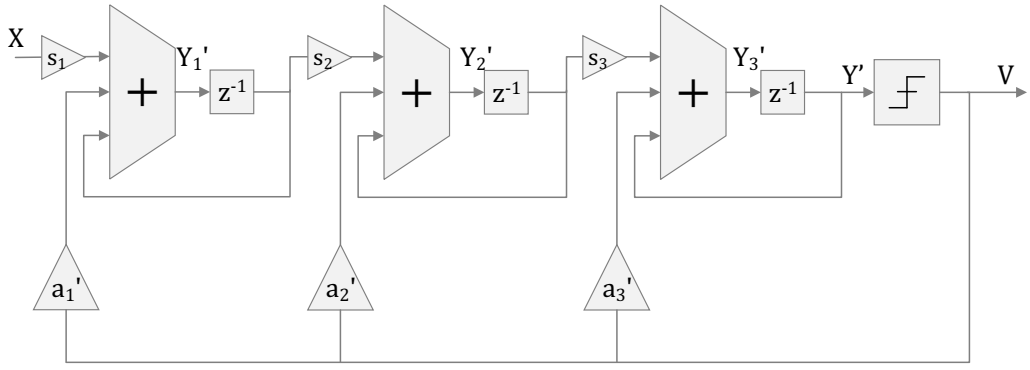



Figure 5.3: CIDF architecture with different scale factor for each integrator

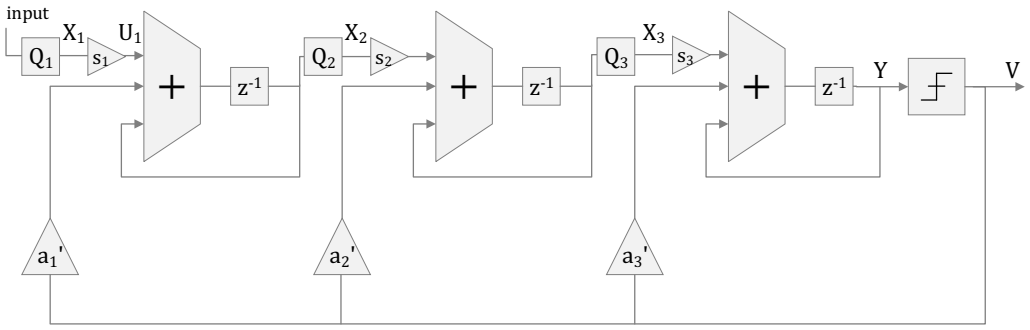



Figure 5.4: CIDF architecture with Quantizers (real system)

in section 3.1 a multi-bit quantizer can be represented by a gain factor plus an error source (fig. 5.5). Assuming that $q_1, q_2, q_3$ are the gains of quantizers $Q_1, Q_2, Q_3$ respectively then :

$$\begin{aligned} p &= q_1 2^{-4} \\ m &= q_2 2^{-1} \\ n &= q_3 2^{-1} \end{aligned} \tag{5.23}$$

where $p, m, n$ are illustrated in fig. 5.5.

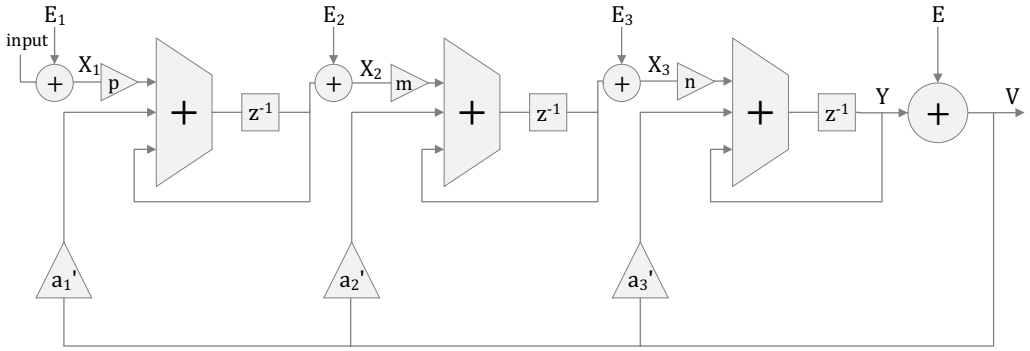



Figure 5.5: CIDF structure. Quantizers replaced by theirs linear model

In a first place we are interested in examining how the introduction of gains change the range i.e the weight factor of signals in FPR affecting hence the balance in the whole system and in which way the same analogy between them could be recovered resulting thus in an equivalent system. In order a comparison between fig. 5.1 and fig. 5.5 to be achievable, the error sources

have to be zeroed. Fortunately the described model of the system is linear, thus the theorem of superposition can be applied. Firstly error sources are zeroed, next searching for an equivalent system to that of fig. 5.1 (namely specifying factors $a_1^{'}, a_2^{'}, a_3^{'}$ of fig. 5.5) and finally adding the error sources back.

## 5.4 Equivalent System after the Introduction of Gains

The following matrix describe the system of fig. 5.1

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \frac{1}{1-z^{-1}} \left( \begin{bmatrix} 0 \\ Y_1 \cdot z^{-1} \\ Y_2 \cdot z^{-1} \end{bmatrix} + \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \cdot V + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} X \right)$$

while the next matrix describes the fig.5.3 (with now $s_1 \rightarrow p, s_2 \rightarrow m, s_3 \rightarrow n$) i.e. takes into consideration the inserted gains from the determination of scale factor for each integrator as well as from the quantization of signals. Every signal as well as the gain of amplifiers $a_1$,$a_2$,$a_3$ are changing value. Symbol ' after a variable shows that it is a new variable. To remind that our goal is to calculate the factors $a_1^{'}$,$a_2^{'}$,$a_3^{'}$ in order to acquire an equivalent architecture with the initial one (fig. 5.1).

$$\begin{bmatrix} Y_1^{'} \\ Y_2^{'} \\ Y_3^{'} \end{bmatrix} = \frac{1}{1-z^{-1}} \left( \begin{bmatrix} 0 \\ mY_1^{'} \cdot z^{-1} \\ nY_2^{'} \cdot z^{-1} \end{bmatrix} + \begin{bmatrix} a_1^{'} \\ a_2^{'} \\ a_3^{'} \end{bmatrix} \cdot V + \begin{bmatrix} p \\ 0 \\ 0 \end{bmatrix} X \right)$$

We know that $Q(Y(z)) = Q(rY(z)) \iff Q(Y_3(z-1)) = Q(rY_3(z-1))$ where r is a positive integer since the sign remains the same. So our goal is to show $Y_3^{'} = rY_3$. For $Y_3^{'} = rY_3$ comparing the third row of first and second matrix we find that $Y_2^{'}$ have to equal $Y_2^{'} = r/n \cdot Y_2$ and $a_3^{'} = r \cdot a_3$. Substituting these values in the second matrix we have :

$$\begin{bmatrix} Y_1^{'} \\ (r/n)Y_2 \\ rY_3 \end{bmatrix} = \frac{1}{1-z^{-1}} \left( \begin{bmatrix} 0 \\ mY_1^{'} \cdot z^{-1} \\ rY_2 \cdot z^{-1} \end{bmatrix} + \begin{bmatrix} a_1^{'} \\ a_2^{'} \\ ra_3 \end{bmatrix} \cdot V + \begin{bmatrix} p \\ 0 \\ 0 \end{bmatrix} X \right)$$

Now the values of $Y_1^{'}$ and $a_2^{'}$ have to be equal to $Y_1^{'} = r/(nm)Y_1$ and $a_2^{'} = r/n \cdot a_2$. Substituting these values in the previous matrix we have :

$$\begin{bmatrix} (r/nm)Y_1 \\ (r/n)Y_2 \\ rY_3 \end{bmatrix} = \frac{1}{1-z^{-1}} \left( \begin{bmatrix} 0 \\ (r/n)Y_1 \cdot z^{-1} \\ rY_2 \cdot z^{-1} \end{bmatrix} + \begin{bmatrix} a_1^{'} \\ (r/n)a_2 \\ ra_3 \end{bmatrix} \cdot V + \begin{bmatrix} p \\ 0 \\ 0 \end{bmatrix} X \right)$$

Now the values of $p$ and $a_1^{'}$ have to be equal to $p = r/(nm)$ and $a_1^{'} = r/(nm) \cdot a_1$. Substituting these values in the previous matrix we have :

$$\begin{bmatrix} r/(nm)Y_1 \\ (r/n)Y_2 \\ rY_3 \end{bmatrix} = \frac{1}{1-z^{-1}} \left( \begin{bmatrix} 0 \\ (r/n)Y_1 \cdot z^{-1} \\ rY_2 \cdot z^{-1} \end{bmatrix} + \begin{bmatrix} r/(nm) \cdot a_1 \\ (r/n)a_2 \\ ra_3 \end{bmatrix} \cdot V + \begin{bmatrix} r/(nm) \\ 0 \\ 0 \end{bmatrix} X \right)$$

That was the proof of equivalence of two systems. Variable r is cancelled and it implies that $Y_3^{'}$ equals $Y_3$. Next, multiplication with $knm$ is made so as to return in the initial matrix. Therefore :

$$\begin{bmatrix} kY_1 \\ kmY_2 \\ kmnY_3 \end{bmatrix} = \frac{1}{1-z^{-1}} \left( \begin{bmatrix} 0 \\ kmY_1 \cdot z^{-1} \\ kmnY_2 \cdot z^{-1} \end{bmatrix} + \begin{bmatrix} ka_1 \\ kma_2 \\ kmna_3 \end{bmatrix} \cdot V + \begin{bmatrix} k \\ 0 \\ 0 \end{bmatrix} X \right)$$

Eventually renaming the variables it arises :

$$\begin{bmatrix} Y_1^{'} \\ Y_2^{'} \\ Y_3^{'} \end{bmatrix} = \frac{1}{1-z^{-1}} \left( \begin{bmatrix} 0 \\ mY_1^{'} \cdot z^{-1} \\ nY_2^{'} \cdot z^{-1} \end{bmatrix} + \begin{bmatrix} k \cdot a_1 \\ km \cdot a_2 \\ kmn \cdot a_3 \end{bmatrix} \cdot V + \begin{bmatrix} k \\ 0 \\ 0 \end{bmatrix} X \right)$$

Therefore the values of $a_1^{'}$,$a_2^{'}$,$a_3^{'}$ in function of $a_1$,$a_2$,$a_3$ and gains k,m,n are obtained analytically.

$$\begin{aligned} a_1^{'} &= k \cdot a1 \\ a_2^{'} &= km \cdot a2 \\ a_3^{'} &= kmn \cdot a3 \end{aligned} \tag{5.24}$$

## 5.5 Defining the Number of Bits for each Signal

Now the affection of each quantizer $Q_1, Q_2, Q_3$ to the output of modulator is studied. As it has been said a quantizer can be replaced by an error source (see fig. 5.5). Our goal is the calculation of transfer functions for each error source . In that way we are informed to what extent each source affects the output of $\Delta\Sigma$ modulator. Applying the principle of superposition inputs X,V and two error sources are setting every time to zero retaining active only one error source. This task is repeated for every error source. Fig. 5.6a-5.6c depicts it.

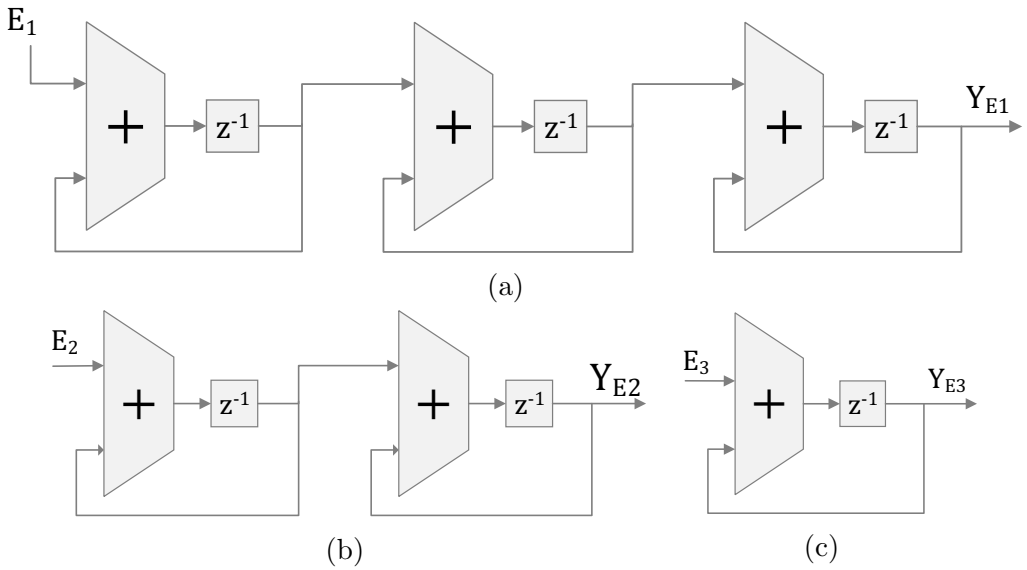



Figure 5.6: Error sources

Easily can be obtained

$$H_1 = \frac{Y_{E_1}}{E_1} = \frac{z^{-3}}{(1-z^{-1})^3} \tag{5.25}$$

$$H_2 = \frac{Y_{E_2}}{E_2} = \frac{z^{-2}}{(1-z^{-1})^2} \tag{5.26}$$

$$H_3 = \frac{Y_{E_3}}{E_3} = \frac{z^{-1}}{1-z^{-1}} \tag{5.27}$$

where $Y_{E_1}$,$Y_{E_2}$,$Y_{E_3}$ is the output due to each error signal $E_1, E_2, E_3$ respectively. Comparing to (2.9) the output now is calculated by involving also the $Y_{E_1}$,$Y_{E_2}$,$Y_{E_3}$ components in the calculation of Y :

$$Y = L_0 \cdot X + L_1 \cdot V + Y_{E_1} + Y_{E_2} + Y_{E_3} \tag{5.28}$$

$$Y = L_0 \cdot X + L_1 \cdot V + H_1 \cdot E_1 + H_2 \cdot E_2 + H_3 \cdot E_3 \tag{5.29}$$

Provided that $V = Y + E$ from (5.29) we have :

$$V = L_0 \cdot X + L_1 \cdot V + H_1 \cdot E_1 + H_2 \cdot E_2 + H_3 \cdot E_3 + E \tag{5.30}$$

$$V = \frac{L_0}{1+L1}X + \frac{H_1}{1+L1}E_1 + \frac{H_2}{1+L1}E_2 + \frac{H_3}{1+L1}E_3 + \frac{1}{1+L1}E \tag{5.31}$$

where $L_0(z)$,$L_1(z)$ are derived from (4.28) and (4.29). As we see the transfer functions (TF) of $E_1(z)$,$E_2(z)$,$E_3(z)$ are respectively :

$$TFE_1 = \frac{H_1}{1+L_1} = \frac{z^{-3}}{1+7z^{-1}-12z^{-2}+5z^{-3}} \tag{5.32}$$

$$TFE_2 = \frac{H_2}{1+L_1} = \frac{z^{-2}(1-z^{-1})}{1+7z^{-1}-12z^{-2}+5z^{-3}} \tag{5.33}$$

$$TFE_3 = \frac{H_3}{1+L_1} = \frac{z^{-1}(1-z^{-1})^2}{1+7z^{-1}-12z^{-2}+5z^{-3}} \tag{5.34}$$

These Transfer Functions (TF) give a very useful information. For frequencies $\Omega$ near 0 the TF of error $E_1$ equals approximately 1 while TFs of errors $E_2, E_3$ equal roughly 0. However $TFE_2$ has only one zero whilst $TFE_3$ has two zeroes in DC frequency. Consequently $TFE_1$ amplifies more than $TFE_2$ and $TFE_2$ amplifies more than $TFE_3$ their errors. Hence, the significance of the errors $E_1$,$E_2$,$E_3$ for an efficient operation of the Modulator is in the order that are written. Based in this fact, the highest to lowest priority in determination of bit-widths is $X_1$,$X_2$,$X_3$(see fig.5.4). The method that we followed is to dedicate the same number of bits for each signal $X_1$,$X_2$,$X_3$ and calculate the Signal to Quantization Noise Ratio (SQNR) in function of bits. The latter is given by :

$$SQNR = 10log_{10}\left[\frac{v_{ip}^2}{(v_{ip}-v_{fp})^2}\right] \tag{5.35}$$

where $v_{ip}$ is the output of 2-SLA Modulator with infinite precision (double precision of Matlab) while $v_{fp}$ is the finite precision corresponding to given bits each time. This measure expresses how good is the implemented system according to the ideal. From this information we are able to choose the bit-width of the signal with the highest priority i.e. $X_1$. It is plotted in fig.5.7a. For more than 15 bits we observe a hard converge. Hence, 15 bits is an adequately good and safe choice. Subsequently keeping the signal $X_1$ in 15 bits and changing the number of bits for signals $X_2$,$X_3$ (same number for both each time) ,and after examining again the SQNR, the determination of bit-width for $X_2$ is possible (second in order of priority).It is plotted in fig.5.7b and 12 bits seems perfect. Eventually since the definition of signals $X_1$ and $X_2$ is done, the SQNR is plotted in function of the number of bits for the signal $X_3$ which has the lowest priority fig.5.7c. The calculation task of bit-widths is completed after the final bit-width choice for signal $X_3$ where 7 bits are good enough.

The final CIDF architecture (ready for implementation on a board!) is presented in fig. 5.8 including the number of bits for signals as well as the gains $(s_1, s_2, s_3)$ in every integrator as calculated in section 5.2 . For the computation of gains $(a_1^{'}, a_2^{'}, a_3^{'})$ the (5.24) is used. The gains of Quantizers $Q_1, Q_2, Q_3$ equal $q_1 = 2^{15}, q_2 = 2^{-3}, q_3 = 2^{-5}$ where the exponential number 15 is given from the bit-width of $X_1$ signal while the exponential numbers -3 and -5 are obtained by subtracting the bit-width of the output from the input for $Q2, Q3$ quantizers respectively (see fig. 5.4 and 5.8). Hence from (5.23) we have that $p = 2^{11}, m = 2^{-4}, n = 2^{-6}$. Eventually from (5.24) :

$$\begin{aligned} a_1^{'} &= 2^{11} \cdot (-1) = -2048 \\ a_2^{'} &= 2^{11}2^{-4} \cdot (-5) = -640 \\ a_3^{'} &= 2^{11}2^{-4}2^{-6} \cdot (-10) = -20 \end{aligned} \tag{5.36}$$

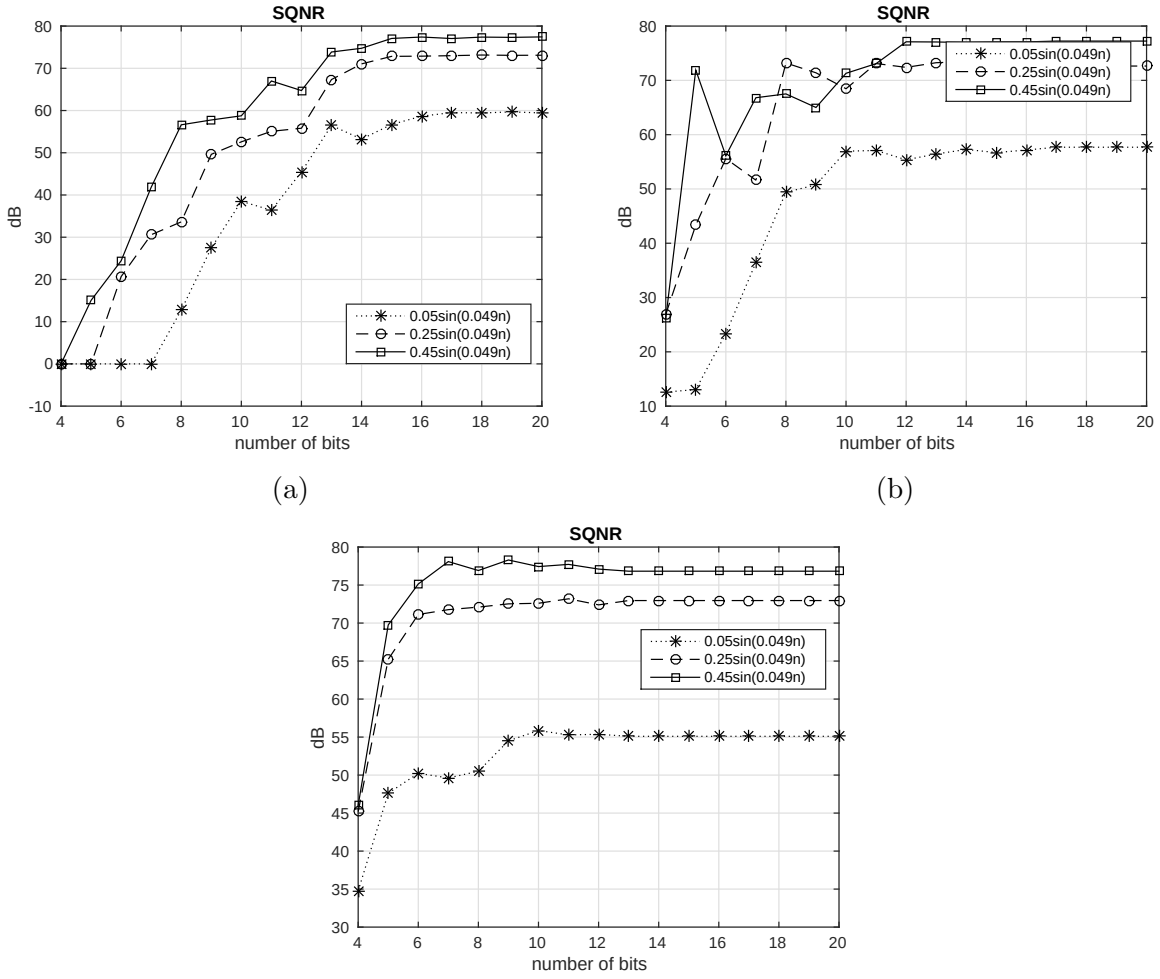



Figure 5.7: Signal to Quantization Noise in function of bits

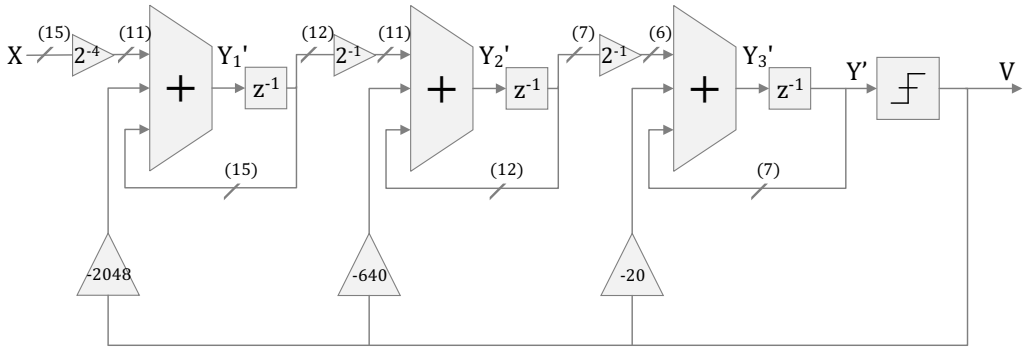



Figure 5.8: CIDF architecture for the construction of our 2-SLA Modulator

# Chapter 6

# FPGA Implementation of 2-SLA Modulator

## 6.1 Verilog Hardware Language

Verilog Hardware Language was written in PLANAHEAD program for the programming of our 2-SLA Modulator on FPGA board. The FPGA model is Xilinx Spartan xc3s500e. The verilog code is :

```
‘timescale 1ns / 1ps

module 2SLA (i_data_in,clock4,reset,out);

localparam bits_y1 = 15;
localparam bits_y2 = 12;
localparam bits_y3 = 7;

input reset,clock4;
input [bits_y1-1:0] i_data_in;
output out;

reg signed [3:0] a1 ;
reg signed [4:0] a2 ;
reg signed [5:0] a3 ;
reg out;
```

```
wire compl ;
wire signed [(bits_y1-1):0] y1_D, y1_D_part ,y1_Q , y1_Q_shift,i_data_in_Q;
wire signed [(bits_y2-1):0] y2_D , y2_Q , y2_Q_shift ,y2_D_part , y1_Q_shift_q ;
wire signed [(bits_y3-1):0] y3_D , y3_D_part, y3_Q , y2_Q_shift_q;

assign compl = ~i_data_in_Q[bits_y1-1];
assign y1_D_part = {a1[3:1], compl ,i_data_in_Q[(bits_y1-1-4):0]};
assign y1_D = y1_D_part + y1_Q ;

assign y2_D_part = {a2 + y1_Q_shift_q[bits_y2-1:bits_y2-5], y1_Q_shift_q[bits_y2-5-1:0]} ;
assign y2_D = y2_D_part  + y2_Q;

assign y3_D_part = {a3 + y2_Q_shift_q[bits_y3-1:bits_y3-6] , y2_Q_shift_q[bits_y3-6-1:0]};
assign y3_D = y3_D_part + y3_Q;

Dff  #(bits_y1)  FFi_data_in (.Q(i_data_in_Q), .D(i_data_in), .reset(reset), .clk(clock4));
Dff  #(bits_y1)  FFy1 (.Q(y1_Q), .D(y1_D), .reset(reset), .clk(clock4));
Dff  #(bits_y2)  FFy2 (.Q(y2_Q), .D(y2_D), .reset(reset), .clk(clock4));
Dff  #(bits_y3)  FFy3 (.Q(y3_Q), .D(y3_D), .reset(reset), .clk(clock4));

assign    y1_Q_shift = y1_Q >>>1;
assign    y1_Q_shift_q = y1_Q_shift[bits_y1-1:bits_y1-bits_y2];

assign    y2_Q_shift = y2_Q >>>1;
assign    y2_Q_shift_q = y2_Q_shift[bits_y2-1:bits_y2-bits_y3];

always @ (y3_Q)
begin
out = y3_Q[bits_y3-1];
if (y3_Q[bits_y3-1])
begin
a1 = 4'b0001;
a2 = 5'b00101;
a3 = 6'b001010;
end
else
begin
```

```
a1 = 4'b1111;
a2 = 5'b11011;
a3 = 6'b110110;
end
end

endmodule
```

and the verilog code for D flip flop module for the synchronization with the clock is :

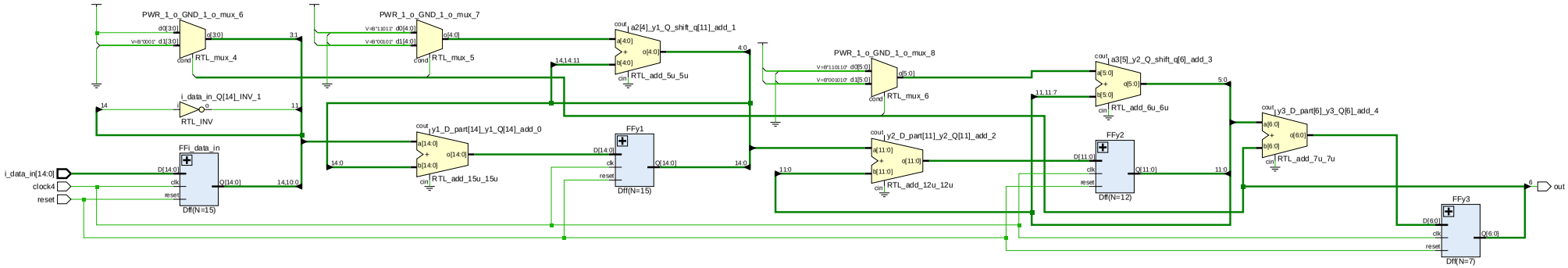
```
`timescale 1 ns/ 1 ps

module Dff #(parameter N=8)
(
output  reg  [N-1 : 0]  Q,
input  [N-1 : 0]  D,
input reset, clk
);

always @ (posedge clk)
    begin
        if(reset == 1'b1)
            Q <= {N-1 {1'b0}};
        else
            Q <= D;
    end

endmodule
```

The RTL schematic is given below :


PWR_1_o_GND_1_o_mux_6
d0[3:0]
V='B'1001' d1[3:0]
o[3:0]
cond
RTL_mux_4
3:1
PWR_1_o_GND_1_o_mux_7
V='B'1011' d0[4:0]
V='B'1010' d1[4:0]
o[4:0]
cond
RTL_mux_5
a2[4]_y1_Q_shift_q[11]_add_1
cout
a[4:0]
b[4:0]
o[4:0]
cin
RTL_add_5u_5u
4:0
14,14:11
i_data_in_Q[14]_INV_1
14
o
RTL_INV
11
y1_D_part[14]_y1_Q[14]_add_0
cout
a[14:0]
b[14:0]
o[14:0]
cin
RTL_add_15u_15u
14:0
FFi_data_in
D[14:0]
clk
reset
Q[14:0]
14,10:0
Dff(N=15)
FFy1
D[14:0]
clk
reset
Q[14:0]
Dff(N=15)
14:0
PWR_1_o_GND_1_o_mux_8
V='B'110110' d0[5:0]
V='B'001010' d1[5:0]
o[5:0]
cond
RTL_mux_6
y2_D_part[11]_y2_Q[11]_add_2
cout
a[11:0]
b[11:0]
o[11:0]
cin
RTL_add_12u_12u
11:0
a3[5]_y2_Q_shift_q[6]_add_3
cout
a[5:0]
b[5:0]
o[5:0]
cin
RTL_add_6u_6u
11,11:7
5:0
FFy2
D[11:0]
clk
reset
Q[11:0]
Dff(N=12)
11:0
y3_D_part[6]_y3_Q[6]_add_4
cout
a[6:0]
b[6:0]
o[6:0]
cin
RTL_add_7u_7u
FFy3
D[6:0]
clk
reset
Q[6:0]
Dff(N=7)
6
out
i_data_in[14:0]
clock4
reset

## 6.2 Output Spectrum

In fig. 6.2a-6.2d are depicted four spectres of our 2-SLA $\Delta\Sigma$ modulator. The four inputs derived from :

$$x = 0.45sin\left(\frac{\pi}{OSR}n\right) \tag{6.1}$$

where OSR takes four different values (are given below). The machine that displays the spectrum is called Spectrum Analyzer (fig 6.1). The Resolution bandwidth of the output spectrum is $3kHz$. If $f_b$ is the frequency of the input signal then easily it can be found from (2.23) :

$$f_b = \frac{f_s}{2 \cdot OSR} \tag{6.2}$$

Hence for $OSR = 64, 128, 512, 2048$ and for $f_s = 50MHz$ the input frequency $f_b$ is $390.6, 195.3, 48.8, 12.2$ kHz respectively. These values agree with the frequency of the displayed diracs (Z Transformation of sinusoidal signals) in the fig. 6.2a-6.2d.

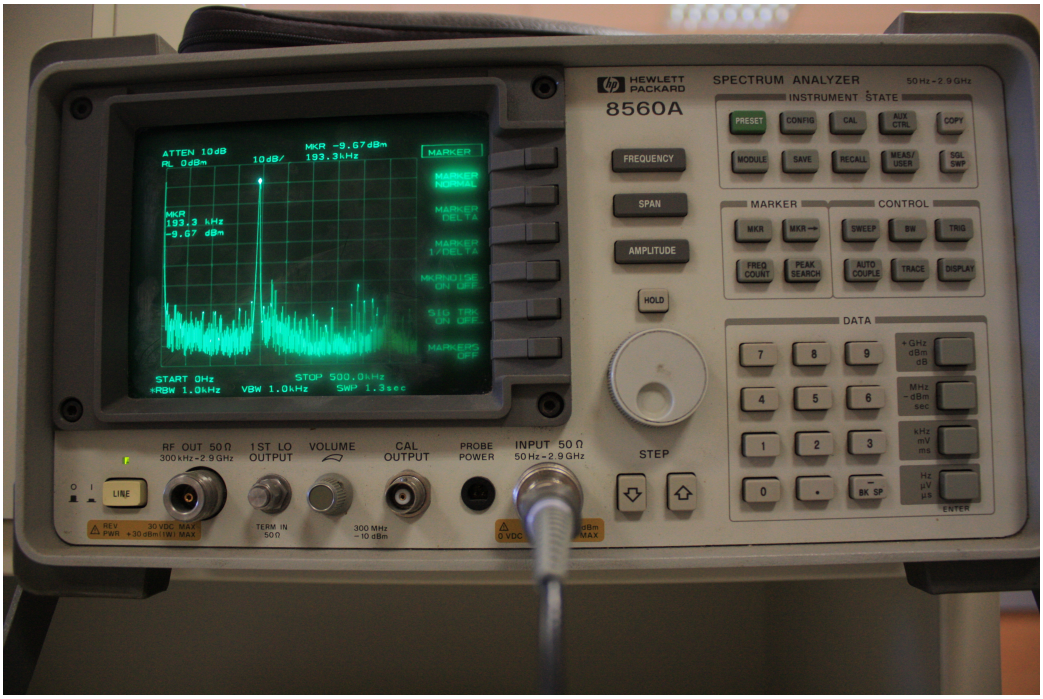
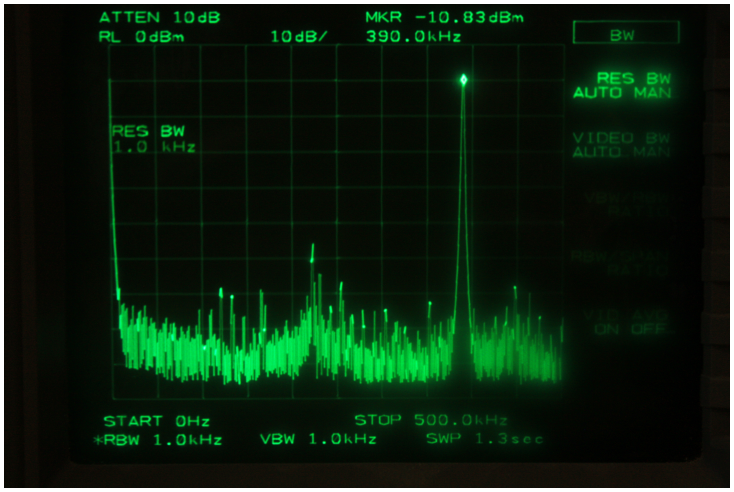
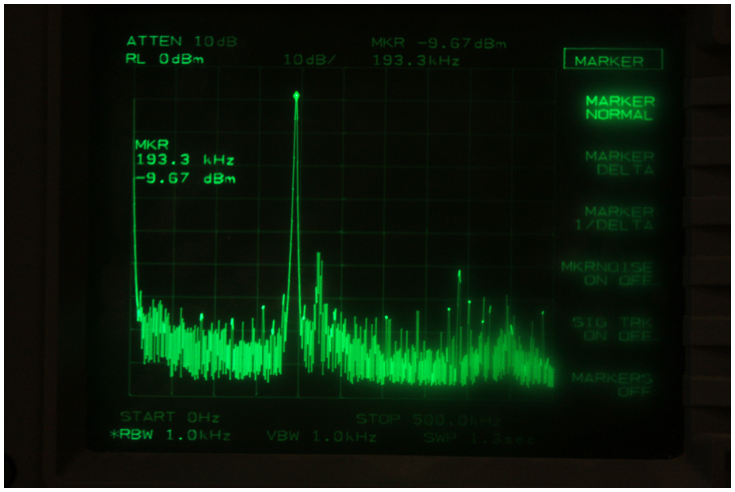


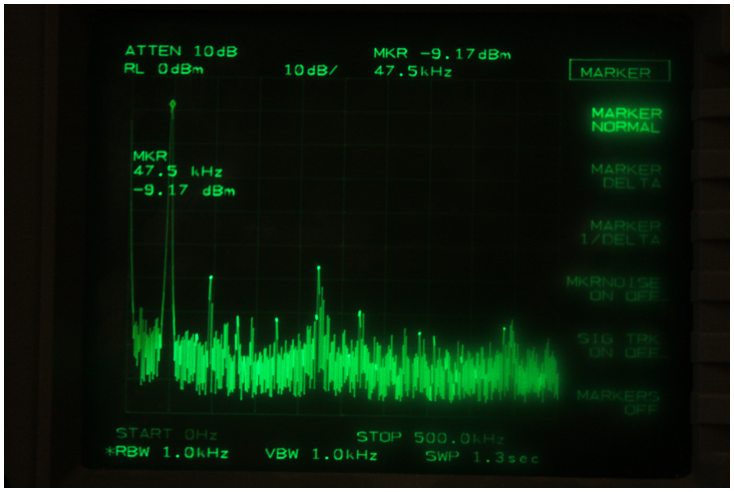

Figure 6.1: Spectrum Analyzer Machine



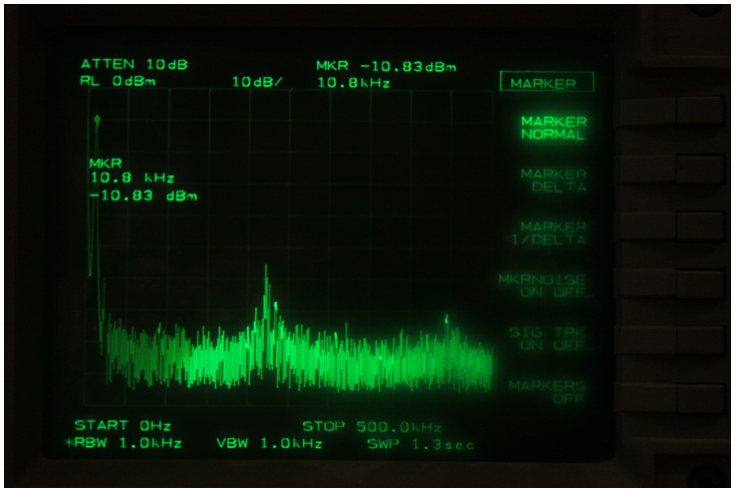

(a) Spectrum for OSR=64




(b) Spectrum for OSR=128




(c) Spectrum for OSR=512




(d) Spectrum for OSR=2048

# Bibliography


[1] Basetas C. , Kanteres A. , Sotiriadis P.P., "Hardware implementation aspects of Multi-Step Look-Ahead $\Sigma$-$\Delta$ modulation-like architectures for all-digital frequency synthesis applications," in *Proc. IEEE Int. Freq. Control Symp. & the European Freq. and Time Forum (FCS)*, 2015.

[2] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*, S. V. Kartalopoulos, Ed. Hoboken, New Jersey: John Wiley & Sons, Inc., 2005.

[3] Jurgen van Engelen,*Stability Analysis and Design of Bandpass Sigma Delta Modulators*, 1999.

[4] G. I. Bourdopoulos, A. Pnevmatikakis, V. Anastassopoulos, and T. L. Deliyannis, *Delta-Sigma Modulators: Modeling, Design and Applications*. 57 Shelton Street, Covent Garden, London WC2H 9HE, UK: Imperial College Press, 2003.

[5] S. H. Ardalan and J. J. Paulos, "An analysis of nonlinear behavior in Delta-Sigma modulators," *IEEE Trans. Circuits Syst.*, vol. 34, no. 6, pp. 593–603, Jun. 1987.

[6] E. Janssen and A. van Roermund, *Look-Ahead Based Sigma-Delta Modulation*. Springer, 2011.

[7] R. Huang, N. Lotze, and Y. Manoli, "On design a high speed sigma delta DAC modulator for a digital communication transceiver on chip," in *Proc. IEEE 11th Euromicro Conference on Digital System Design Architectures, Methods and Tools*, 2008, pp. 53–60.

[8] K. Galanopoulos, C. Basetas, and P. P. Sotiriadis, "Delta-sigma modulation techniques to reduce noise and spurs in all-digital RF transmitters," in *Proc. IEEE Int. Freq. Control Symp.*, Taipei, May 19–22 2014.

[9] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 2nd ed. Upper Saddle River, New Jersey 07458: Prentice-Hall, 1999.