A High Speed DAC Architecture for Continuous Time Pipeline ADC

Delft University of Technology

Mohammed Abo Alainein

TU Delft

A High Speed DAC Architecture for Continuous Time Pipeline ADC

by

Mohammed Abo Alainein

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Tuesday March 24, 2023 at 10:00 AM.

Student number:	4570081	
Project duration:	September 1, 2022 – March 1, 2023	
Thesis committee:	Dr. ir. M. Bolatkale,	TU Delft, NXP Semiconductor, supervisor
	Dr. S. M. Alavi,	TU Delft
	Ir. S. Bajoria,	NXP Semiconductors, supervisor
	Ir. Q. Liu,	NXP Semiconductors, supervisor

This thesis is confidential and cannot be made public until January 01, 2026.

TU Delft

Acknowledgments

First and foremost, I would like to thank NXP Semiconductors for providing me with the opportunity to carry out my master thesis project and for providing me with the necessary resources required for this work. This opportunity was crucial in helping me gain experience and develop my skills as an engineer. I would like to express my sincere gratitude to my thesis supervisor, Muhammed Bolatkale, for his support and guidance throughout my thesis project. His advice and feedback have been instrumental in shaping my understanding and in helping me complete this thesis. He always challenged me to think critically and deeply about my research, and his insightful questions and feedback changed my way of thinking and opened up new avenues of research. I would also like to thank Shagun Bajoria for pushing me forward and believing in me every step of the way. His encouragement and constructive feedback were crucial in keeping me motivated and focused throughout the thesis project. Furthermore, I am deeply indebted to Qilong Liu for his tireless assistance and guidance. His daily support and assistance were invaluable in helping me navigate and overcome challenges throughout the course of this thesis. Lastly, I would like to thank my family and friends for their support and love. Their encouragement and understanding kept me motivated and inspired throughout this journey.

Abstract

This thesis provides an investigation of the architecture and the design of the coarse DACs in continuous time pipeline (CTP) ADC to achieve $SFDR < -104dB$ within a bandwidth of $400MHz$ at sampling frequency of $4.8GHz$ in TSMC 28nm technology.
Mismatch errors of the coarse DACs in CTP ADC are very critical as they introduce distortion and leak the quantization noise of the coarse stages to the output. Conventional calibration techniques such as dynamic element matching (DEM) linearize the DACs by converting the DAC distortion to white noise. However, after the linearization, the residual gain errors of the DACs remain. As a result, the quantization noise of the coarse quantizers leak to the output and degrade the performance of the CTP. Therefore, the residual gain errors of the DACs need to be estimated and calibrated. A dual return to open resistive DAC is proposed in the first stage of the CTP. The proposed architecture employs conventional DEM technique and is verified within the first stage of the CTP with ideal digital back-end calibration. Simulation results shows an $SFDR < -104\ dBFS$ and a worst case $SNDR$ of $75\ dB$ for 60 Monte Carlo runs.

Furthermore, two new innovative techniques are presented in this thesis. The first technique, advanced dynamic element matching (ADEM), translates both the distortion and the gain errors due to element mismatch of the DACs in multi-stage CTP ADC into white noise. The second technique, advanced data weighted averaging (ADWA), noise shapes both the the distortion and the gain errors of the DACs. Therefore, the presented techniques do not require additional digital calibration for element mismatch errors. Finally, a DAC architecture is presented that allows a feasible implementation of the presented techniques. The techniques are verified using simulations in MATLAB and Cadence. However, The presented techniques require the CTP stages to have equal impedances.

Contents

1 Introduction

Frequency Modulated Continuous Wave (FMCW) radar, is a technology that is increasingly being used in autonomous driving vehicles to help them detect and navigate their environment. FMCW radar uses a continuous wave signal that is frequency modulated, which allows it to measure the range, velocity, and angle of objects in its field of view. One of the key parameters that determine the performance of FMCW radar is its bandwidth, which is the range of frequencies over which the radar signal is modulated. The bandwidth of an FMCW radar determines its range resolution, which is the ability of the radar to distinguish between objects that are close together. A wider bandwidth allows for a higher range resolution, which means that the radar can detect and track multiple objects that are close together with higher accuracy [1–3]. This is particularly important for autonomous vehicles, as it allows them to detect and avoid multiple obstacles that may be in close proximity to each other on the road. The bandwidth also determines the velocity resolution of an FMCW radar, which is the ability of the radar to measure the velocity of an object [3]. A wider bandwidth allows for a higher velocity resolution, which means that the radar can measure the velocity of an object with greater accuracy. This is important for autonomous vehicles, as it allows them to predict the motion of other vehicles, pedestrians, and other objects on the road, and make decisions based on that information. Other important parameter in FMCW radar sensors is the noise floor. A lower noise floor allows the radar to detect weaker signals with higher sensitivity [4].
Analog-to-Digital Converters (ADCs) play a critical role in the operation of FMCW radar systems. These converters are responsible for converting the analog radar signals that are received by the antenna into a digital format that can be processed by the radar's signal processing system. High bandwidth, linearity, and high resolution are the key requirements for ADCs in FMCW radar systems, as they directly affect the radar's measurements [4].

Discrete time pipeline (DTP) ADC architectures are known for their ability to deliver high resolutions across a wide signal bandwidth. However, DTP ADCs typically utilize an anti aliasing filter (AAF) preceding the ADC to mitigate signal aliasing artifacts. The AAF is usually followed by an integrated buffer to drive the ADC. Both the AAF and the buffer are very problematic as they significantly increase the system power consumption and system noise [5]. These issues are solved in continuous time sigma delta ($CT\Sigma\Delta$) ADCs since these ADCs have inherent anti aliasing filtering and they are easy to drive due to their resistive input impedance [6]. To achieve adequate noise shaping in $CT\Sigma\Delta$ ADCs, high oversampling ratio is needed. Furthermore, $CT\Sigma\Delta$ uses negative feedback. This restricts maximum sampling rate and hence limits the maximum achievable bandwidth while ensuring the power efficiency [5, 7]. These issues are solved in continuous time pipeline (CTP) ADCs. CTP architectures provide inherent anti alias filtering and are able to achieve high bandwidth in low Oversampling Ratios (OSR) while ensuring the power efficiency. Furthermore, CTP ADCs are easy to drive due to their resistive input impedance [5–8].

The main objective of the thesis is to investigate the implementation and the design of the coarse DACs within the CTP ADC in TSMC 28nm technology. This is done as follows. Firstly, in Chapter 2, an overview of the CTP ADC is given with its design specification. Hereafter, the impact of the non-idealities of the coarse DACs within the CTP ADC is investigated and requirements for the coarse DACs are derived based on MATLAB simulations. In Chapter 3, two new innovative calibration techniques for the coarse DACs are presented and verified by simulations in MATLAB. Thereafter, in Chapter 4, a DAC architecture for a multi-stage CTP ADC is proposed. The DAC architecture allows a feasible implementation of the proposed calibration techniques and is verified by simulations in Cadence. However, the techniques require the CTP stages to have equal impedances. Therefore, due to impedance scaling imposed on the CTP stages, the presented techniques are not implemented and investigated any further. In light of this, an alternative DAC architecture is investigated with impedance scaling taken into account. The implementation of the DAC architecture is presented and verified in the first stage of the CTP.

2

Continuous Time Pipeline ADC

2.1. Architecture Overview

A general block diagram of the CTP architecture is shown in Figure 2.1. It consists of $k-1$ identical coarse or front-end stages, a back-end stage and $k-1$ digital compensation filters $H_i(z)$.

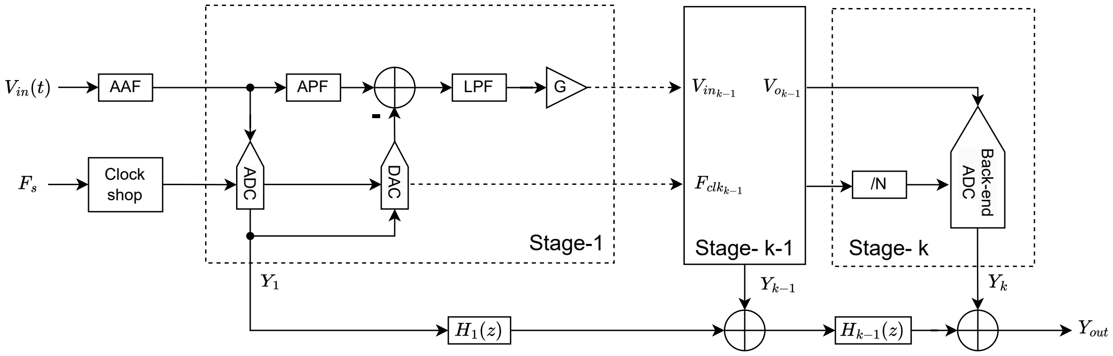


Figure 2.1: General block diagram of CTP architecture

The input to the CTP ADC passes first through an anti aliasing filter (AAF) to attenuate out of band blockers that alias back in the signal-band and degrade the dynamic range of the converter or cause saturation to internal nodes. Then the signal is fed into the first coarse stage where it is sampled by a low resolution coarse ADC. The coarse DAC generates the analog output based on the digital input generated by the preceding ADC. The output of the coarse DAC is subtracted from a delayed version of the input signal $V_{in}(t)$ to generate a residue signal. The analog input signal is delayed by means of continuous time all- pass filter (APF), where the delay provided by the filter matches the delay in the ADC-DAC path. If the delay of the APF does not match the delay of the ADC-DAC path, the residue signal increases and might cause saturation. The residue signal (ADC quantization noise, non-linearity and sampling images) is further filtered by a low pass filter (LPF) and amplified with a inter-stage gain factor (G) before it is applied to the subsequent stages, where the same operation is repeated. The LPF must ensure that a desired amount of gain G can be applied to the residue signal without causing saturation. Once the input signal is processed through all front-end stages, it is digitized by the back-end ADC. The final output Y_{out} is obtained by combining the digital output of each stage in digital compensation filters $H_i(z)$. Ideally, the errors related to the front end ADCs are cancelled. The digital filter $H_i(z)$ of each coarse stage must mimic the transfer function of the continuous time path DAC-LPF-G of each stage. Otherwise, the quantization noise errors of the coarse ADCs will leak to the output and degrade the signal to quantization noise ratio (SQNR) of the converter.
Ideally, The SQNR of CTP ADC for a nyquist back-end ADC is estimated in Equation 2.1.

$$SQNR = 6.02N + 1.76 + 10log\left(\frac{fs_{BE}}{2 \cdot BW}\right) + \sum_{i=1}^{k-1} G_i \tag{2.1}$$

Where, N is the number of bits of the back-end ADC, fs_{BE} is the sampling frequency of the back-end ADC, BW is the bandwidth and G_i is the inter-stage gain of the coarse stage i.

To achieve the requirements and specifications needed for the targeted application, it has been proposed to develop a CTP ADC in 28 nm CMOS technology with 3 front end stages, each stage has a resolution of 3 bits and sampling frequency of $4.8\ GHz$. The back end stage is a 6 bits Successive Approximation Register (SAR) ADC sampled at $1.6\ GHz$. The targeted bandwidth is $400\ MHz$ with Dynamic Range (DR) of $70\ dB$. The specifications for the DR are listed in Table 2.1.

Table 2.1: DR Specifications

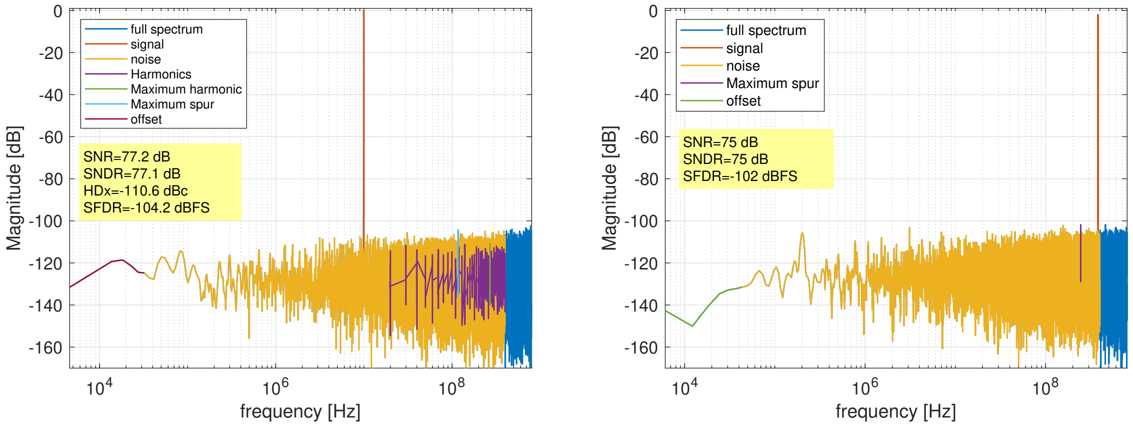
Noise Source	Value (dB)
Quantization	75
Jitter	75
Digital Mismatch	75
Thermal	71
DR	**67.6**

Furthermore, CTP ADC has to achieve harmonic distortion $HDx < -80\ dBc$ and spurious free dynamic range $SFDR < -104\ dBFS$. The SFDR excludes HDx but includes aliases of it. A summary of the system specifications is given in Table 2.2.

Table 2.2: System Specifications

Parameter	Spec	Unit	Parameter	Spec	Unit
Coarse resolution	3	$bits$	HDx	-80	dBc
Back-end resolution	6	$bits$	SFDR	-104	$dBFS$
fs_{coarse}	4.8	GHz	DR	67.6	dB
fs_{BE}	1.6	GHz	Noise density	10.3	$nV/\sqrt{Hz}$
Gain G	12	dB	Supply	0.9	V
Bandwidth BW	400	MHz	Process Technology	28	nm

Figure 2.2 shows a MATLAB simulation of the output spectrum of the CTP ADC. In this simulation all system blocks are considered to be ideal. The APF used in the MATLAB model is a second order filter with the required latency and a unity magnitude response. The LPF is a second order filter. The DAC is an ideal Non-return-to-zero (NRZ) DAC. Finally, The digital compensation filters $H_i(z)$ have transfer functions that exactly replicate the corresponding continuous time path of each coarse stage.



(a) $F_{in} = 10\ MHz$ (b) $F_{in} = 380\ MHz$

Figure 2.2: Output spectrum of the ideal CTP with low and high frequency input tone. FFT points $= 2^{16}$

2.2. Coarse DAC in Continuous Time Pipeline ADC

In this section, the coarse DACs are analyzed. Based on the analysis, requirements for the coarse DACs are derived. The results introduced in this section are mainly based on MATLAB simulations. All blocks, except the coarse DACs, are considered to be ideal.
The mathematical analysis and equations presented in this chapter, are based on 4 coarse stages, each stage has a resolution of 2-bits. The analysis can easily be extended to N-bits and K (>1).

2.2.1. DAC Gain Errors

Mismatch in DAC elements cause the DAC output levels to deviate from their nominal values. This deviation introduce distortion and a gain error e_g. The DAC distortion due to element mismatch is discussed later in section 2.2.3. The gain errors of the coarse DACs leak the quantization noise of the coarse stages to the output. Assume that e_{gi} is the DAC gain error of the coarse stage i in the simplified block diagram of the pipeline shown in Figure 2.3. Furthermore, G_i is the inter-stage gain and Q_i is the quantization noise of the coarse stage i. Q_{BE} is the back-end quantization noise. V_{in} is the input to the pipeline. The output V_{out} of the four coarse stages can be estimated as follows

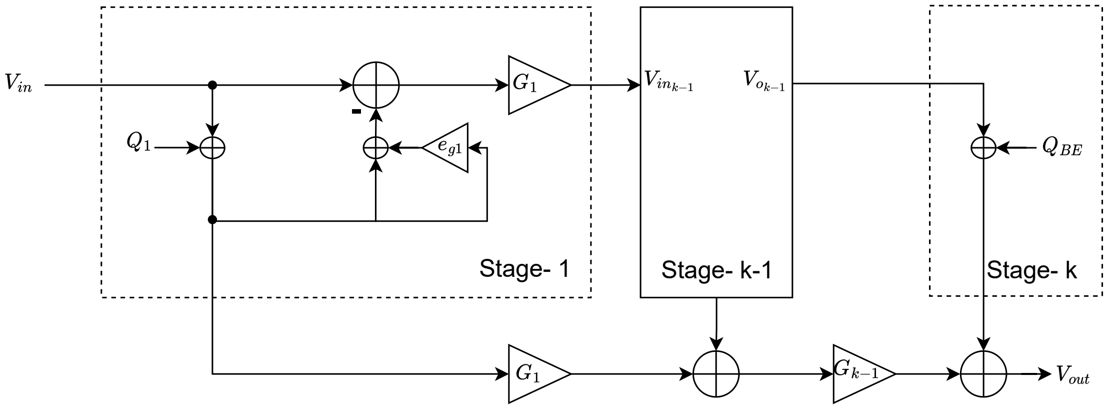
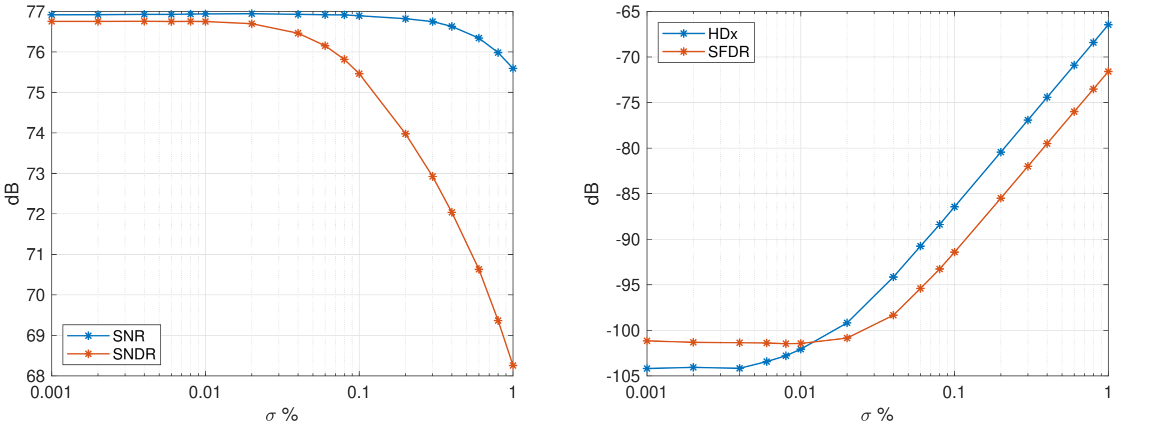


Figure 2.3: Simplified block diagram with error sources

$$\begin{aligned} V_{out} = {} & G^4\, V_{in}\left(1 - e_{g1} + e_{g1}e_{g2} - e_{g1}e_{g2}e_{g3} + e_{g1}e_{g2}e_{g3}e_{g4}\right) \\ & + G^4 Q_1\left(-e_{g1} + e_{g2} + e_{g1}e_{g2} - e_{g2}e_{g3} - e_{g1}e_{g2}e_{g3} + e_{g2}e_{g3}e_{g4} + e_{g1}e_{g2}e_{g3}e_{g4}\right) \\ & + G^3 Q_2\left(-e_{g2} + e_{g3} + e_{g2}e_{g3} - e_{g3}e_{g4} - e_{g2}e_{g3}e_{g4}\right) \\ & + G^2 Q_3\left(-e_{g3} + e_{g4} + e_{g3}e_{g4}\right) \\ & + G Q_4\left(-e_{g4}\right) \\ & + Q_{BE} \end{aligned} \tag{2.2}$$

The quantization noise leakage may vary depending on the relative errors between the DACs. For example if e_{g1} has an opposite sign to e_{g2} then the leakage of Q_1 becomes more severe. This analysis is also confirmed by MATLAB simulations. Figure 2.4 show performance of the CTP ADC when the coarse DACs have different gain errors. The gain errors are drawn from the standard Gaussian distribution with standard deviation of σ.



(a) SNR and SNDR at $F_{in} = 10\ MHz$ (b) HDx at $F_{in} = 10\ MHz$ and SFDR at $f_{in} = 380\ MHz$

Figure 2.4: CTP performance when coarse DACs have different gain errors. Each simulation point is the average of 50 runs

The gain errors of the coarse DACs may be estimated and calibrated by the digital compensation filters $H_i(z)$. However, the filter implementation is outside the scope of this thesis. For further details the reader may refer to [9, 10]. Henceforth, it is assumed, unless mentioned otherwise, that the gain errors of the DACs can be estimated and calibrated by the digital filters. Note that in the next chapter, new innovative calibration techniques are proposed to calibrate the gain errors of the coarse DACs due to element mismatch without the need of estimating the values of the DAC gain errors by the digital filters. Since the DAC errors are generated by MATLAB, the values of DAC gain errors can be extracted and fed directly into the digital filters. This is shown in Figure 2.5. The gain blocks e_{gi} have the same values of the DAC gain errors. Note that this is an ideal cancellation of the DAC gain errors and it does not represent realistic digital calibration scheme since the values of the gain error blocks need to be estimated. Therefore, this calibration represent an ideal calibration for the DAC gain errors and it is only used as an ideal reference. Finally, dither can be applied to the coarse ADCs to relax the quantization noise leakage requirements of the digital filter. Dither reduces the tonal behaviour of the coarse ADC. If the coarse ADC is sufficiently linearized, then quantization noise resembles white noise. Then any leakage of the quantization noise will not degrade the SFDR and HDx performance but will reduce the SNR of the CTP. The ADC dither requirements fall outside the scope of this thesis. Nevertheless, at a later stage of the project, dither and noise shaping techniques were applied to the coarse ADC to achieve an effective linearity equivalent to an 8-bits ADC. However, the results presented, unless mentioned otherwise, assumes a coarse ADC of 3 bits without any linearization techniques.

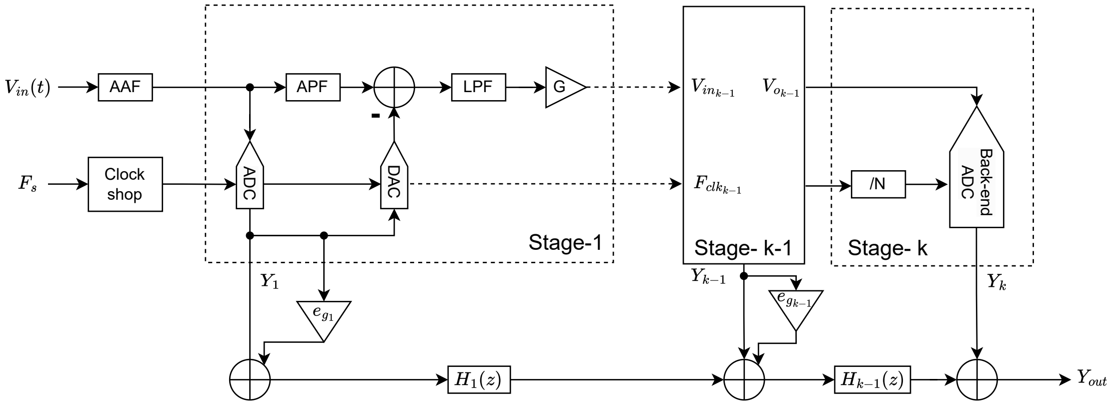


Figure 2.5: Modified block diagram of CTP architecture for calibrating the DAC gain errors

2.2.2. DAC Waveforms and Inter-symbol Interference

The purpose of the DAC is to convert a sequence of digital input values $x[n]$ into an analog waveform $y(t)$. The input of the DAC is updated at times nT where $n = 0, 1, 2, ..$ is the sample number and T is the sample period. For the sake of simplicity, consider a 2 bits model of the DAC as shown in Figure 2.6. The input of the DAC is a thermometer coded digital input C_i where C_i can be 0 or 1. The block $(2C - 1) \cdot p(t - NT)$ represent the output of one unit element of the DAC. If the input of the DAC unit element is $C_i[n] = 0$ then the output of the unit element is $-p(t - nT)$. Similarly, if $C_i[n] = 1$, the output is $+p(t - nT)$ where $p(t)$ represents the pulse shape of the unit element. The output of each unit element is summed and scaled by a Δ where $\Delta = \frac{Vsup}{2^N}$ denotes the DAC's minimum step size. N is the number of bits and $Vsup$ is the supply voltage. This model can be extended to N-bits by simply having $2^N - 1$ unit elements.

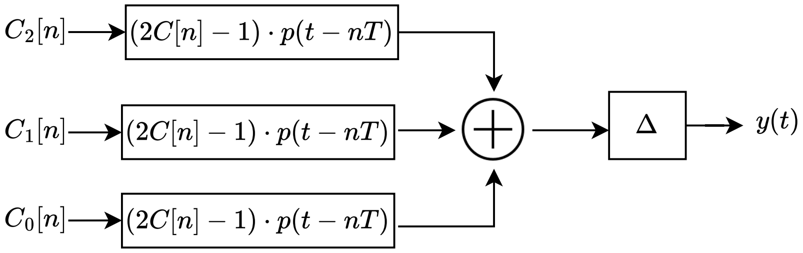

$C_2[n]$
$(2C[n] - 1) \cdot p(t - nT)$
$C_1[n]$
$(2C[n] - 1) \cdot p(t - nT)$
$C_0[n]$
$(2C[n] - 1) \cdot p(t - nT)$
Δ
$y(t)$

Figure 2.6: 2-bits model the DAC

During the nth sampling period, the output of the DAC can be written as follows:

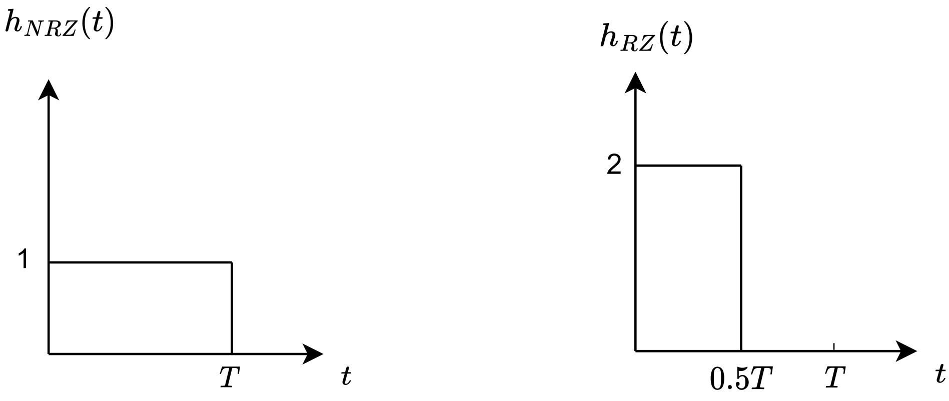
$$y(t) = \sum_{i=1}^{2^{N-1}} (2C_i[n] - 1) \cdot p_i(t - nT) \tag{2.3}$$

Common pulse shapes are Non-return-to-zero (NRZ) and Return-to-zero (RTZ) as illustrated in Figure 2.7. The NRZ DAC holds the value of the digital data for one clock periode T, where RZ DAC holds the data for $0.5T$. The transfer function of the DAC waveforms can be expressed as:

$$H_{NRZ}(s) = \frac{1 - e^{-sT}}{s} \tag{2.4}$$

$$H_{RZ}(s) = 2\frac{1 - e^{-sT/2}}{s} \tag{2.5}$$

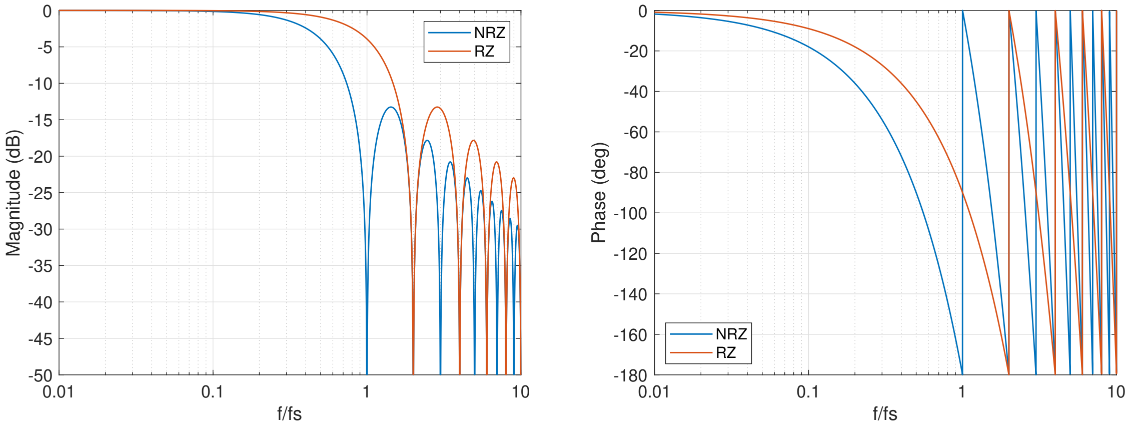
Figure 2.8 depicts the magnitude and phase responses of NRZ and RZ DACs.The RZ DAC results in less dropping for multiple Nyquist bands, which is not desirable in the CTP ADC as this might increase



(a) NRZ

(b) RZ

Figure 2.7: DAC impulse response



(a) Magnitude response

(b) Phase response

Figure 2.8: Frequency response of NRZ and RZ DAC

the residue signal and complicate the design of the APF. Furthermore, the NRZ DAC has an effective delay of $0.5T$, while the RZ DAC has a delay of $0.25T$.
The major problem with NRZ DACs is the Inter-symbol interference (ISI). This is an instance of dynamic non-linearity caused by the differences between the rise t_r and fall t_f times of the NRZ DAC. The root cause is the mismatch between the resistance and timing of the PMOS and NMOS switches in the DAC unit element. Figure 2.9 shows the NRZ waveform of a 1-bit DAC and the error due to asymmetry in rise and fall time. This error is data dependent and it results strong second order distortion and increased noise floor [11]. This is confirmed by the simulations shown in Figure 2.10, where $t_r = 10\,ps$ and $t_f = 11\,ps$. This mismatch of 10% between rise and fall time is introduced to the unit elements of the NRZ DAC in the first coarse stage. Each unit element is identical such that there is no mismatch between the unit elements.The mismatch between the unit elements is discussed later in Section 2.2.3.

The rise an fall time may be equalized by proper sizing of NMOS and PMOS devices. However, the rise and fall times will not match up across variations in process, voltage, and temperature. According to schematic simulations of a simple resistive NRZ DAC in 28n technology, the rise and fall times may have deviation up to 20% across different process corners. This will result in significant ISI.

Figures 2.11 and 2.12 show MATLAB simulations of the CTP ADC with mismatch in rise and fall times of the NRZ coarse DACs. The mismatch in rise and fall times is added to each stage separately. The

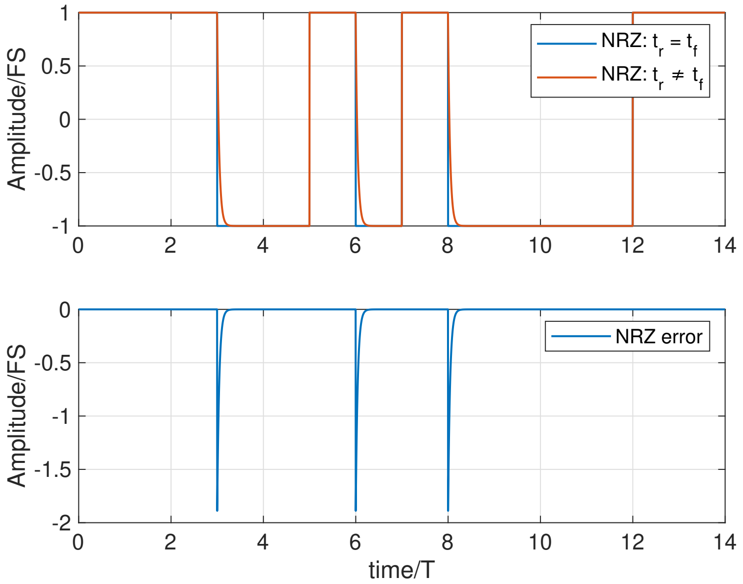
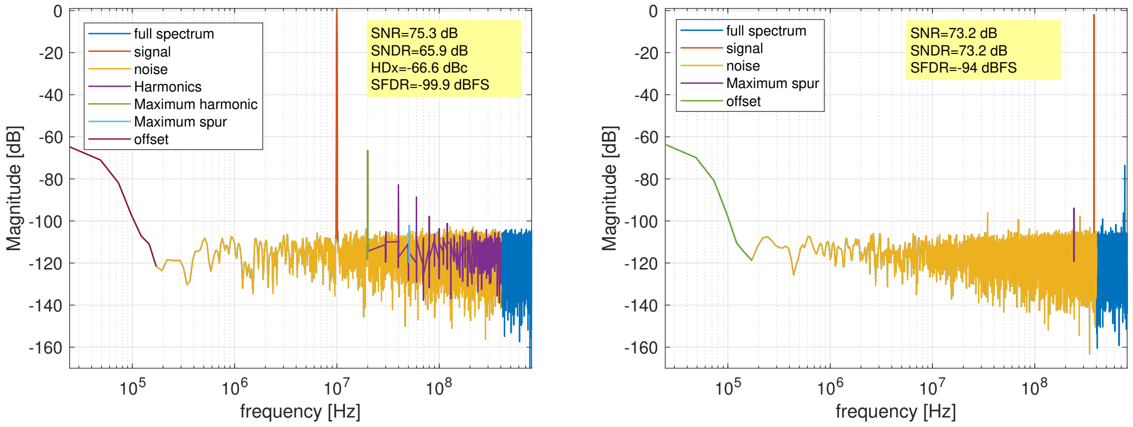


Figure 2.9: NRZ wavefrom of 1-bit DAC



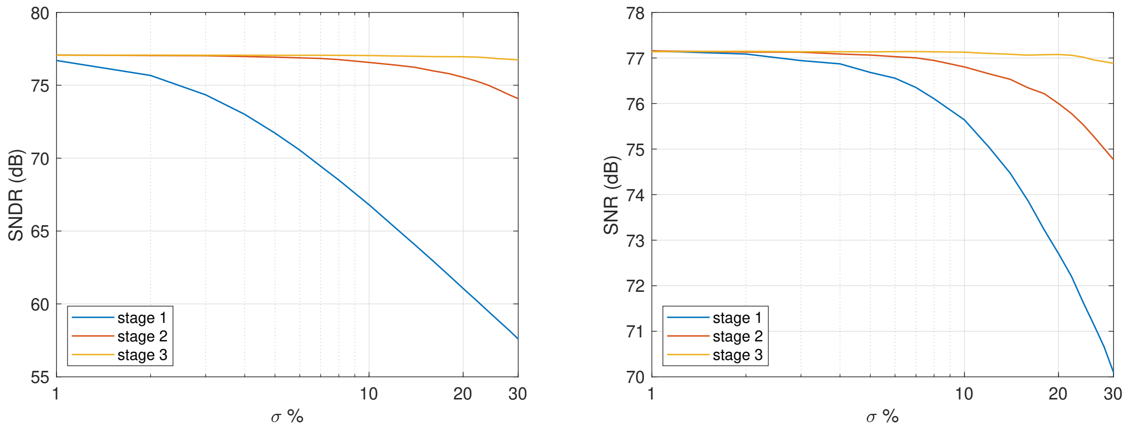
(a) $F_{in} = 10\,MHz$

(b) $F_{in} = 380\,MHz$

Figure 2.10: Output spectrum of the CTP with 10% asymmetry in rise and fall time of the first coarse DAC with NRZ waveform. $t_r = 10ps$ and $tf = 11ps$

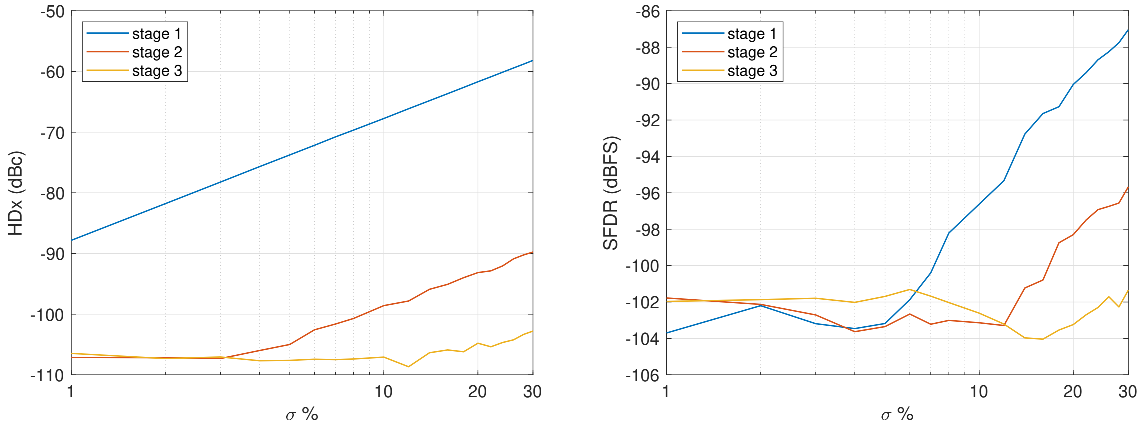
mismatch is successively considered in each stage while assuming the DACs in all other stages to be ideal, starting with the first coarse stage, then moving on to the second, and finally the last coarse stage. The errors originating form the first coarse stages are the most dominant since the errors occurs at later stages are suppressed by the accumulative gain of the preceding stages. Based on the simulation results, it is important to address the ISI issue for the first two stages.

Many techniques exist to mitigate the ISI [11–15]. One popular solution is to use RZ DAC waveform. Since an RZ waveform has a rising and falling edge in every clock cycle, unequal rise/fall times do not result in non-linearity. However, RZ waveform has higher sensitivity to jitter[8].
The major problem of the RZ DACs in CTP is the increased residue signal compared to NRZ DACs. To illustrate this, consider the the first stage of the CTP ADC shown in Figure 2.13. The output of the amplifier can be written in the frequency domain as follows:



(a) $F_{in} = 10\ MHz$ (b) $F_{in} = 10\ MHz$

Figure 2.11: Rise and fall time asymmetry in NRZ DACs of the coarse stages. The mean value is $10ps$



(a) $F_{in} = 10\ MHz$ (b) $F_{in} = 380\ MHz$

Figure 2.12: Rise and fall time asymmetry in NRZ DACs of the coarse stages. The mean value is $10ps$

$$\begin{aligned} Y(f) = U(f) \cdot \left[H_{APF}(f) - e^{-j2\pi f t_d} \cdot H_{dac}(f)\right] \cdot G \cdot H_{LPF}(f) \\ -G \cdot H_{LPF}(f) \cdot H_{dac}(f) \cdot e^{-j2\pi f t_d} \cdot \sum_{k\neq 0} U(f - kf_s) \\ -G \cdot H_{LPF}(f) \cdot H_{dac}(f) \cdot e^{-j2\pi f t_d} \cdot \sum_{k=0} Q_1(f - kf_s) \end{aligned} \tag{2.6}$$

Where, $t_d = T_s$ represent the excess delay in the ADC-DAC path. The first term of the equation is the leakage of the input signal due to ADC-DAC delay mismatch. The second term represents the input images and the last term represents the quantization noise of the coarse ADC. Furthermore consider a second order LPF and a first order approximation of the APF as follows:

$$H_{APF}(s) = \frac{1 - sT_{APF}/2}{1 + sT_{APF}/2} \tag{2.7}$$

Where T_{APF} is equal to $1.5T_s$ for NRZ and $1.25Ts$ for RZ DAC. Figure 2.14a shows the output of the amplifier for RZ and NRZ DAC. Clearly, RZ DAC increases the residue signal. This is problematic because it may cause the amplifier to clip and introduce distortion. Furthermore, small residue signals are

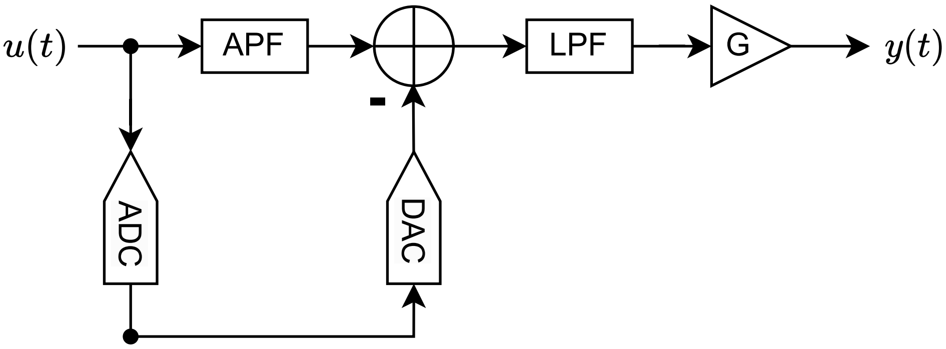
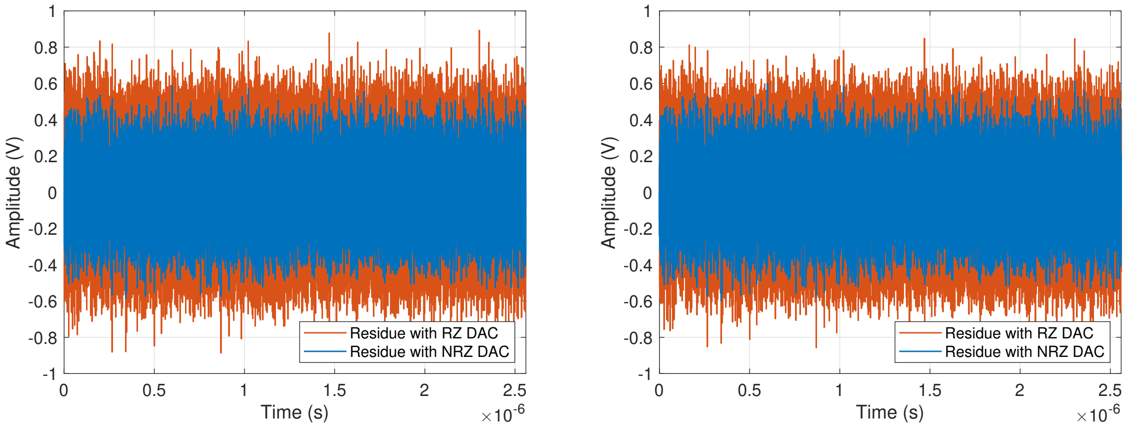


Figure 2.13: Stage-1 of CTP ADC

preferred to avoid overloading the subsequent stages. Having an APF with better delay compensation does not improve the residue signal as shown in Figure 2.14b, where the APF is implemented as an ideal delay element with unity magnitude response. Based on Equation 2.6, the residue can be minimized if the APF has attenuation that mimics the DAC transfer. If the APF and LPF can be combined to map the transfer of the RZ DAC, then the residue can be minimized.



(a) Amplifier output with APF implemented according to Equation 2.7

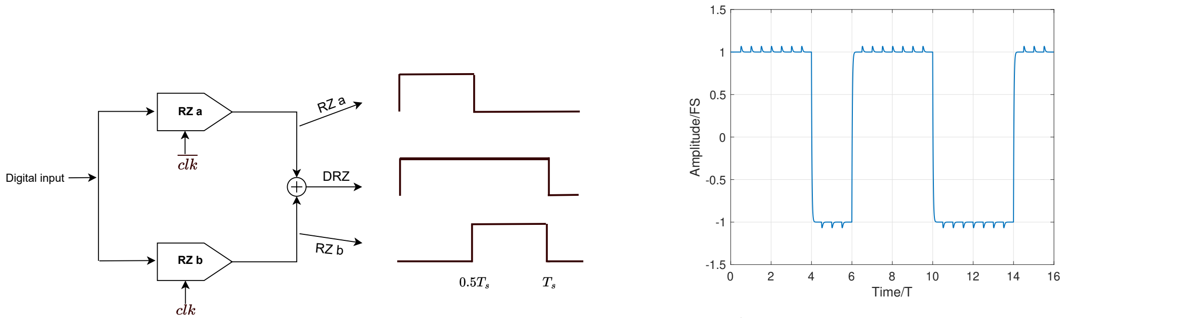
(b) Amplifier output with APF implemented as an ideal delay element

Figure 2.14: Amplifier output of the first stage

To address the ISI and the saturation issues, we can utilize a Dual Return-to-Zero (DRZ) DAC, which combines the benefits of both NRZ and RZ DACs [15, 16]. In DRZ, two independent return to zero signals are generated with a time offset of $0.5T_s$. These signals are then linearly added to form an output that is continuous over the full clock period. The DRZ scheme is shown in Figure 2.15a. The DRZ DAC has an effective transfer function that is equal to the NRZ DAC. Furthermore, when two RZ signals are free from ISI, their linear sum will also be ISI free [15].

Asymmetry in rise and fall time of the DAC unit element in DRZ scheme results in glitches. Figure 2.15b shows an example of the DRZ waveform with 20% mismatch between the rise and fall times in each RZ half. The glitches are data independent since they occur at the clock rate and therefore they do not degrade the linearity of the DAC. Similarly, any mismatch between the RZ halves, will not result any data dependent errors. However, any mismatch between the unit elements of the DAC will result in harmonic distortion. The mismatch between the unit elements is discussed in 2.2.3.

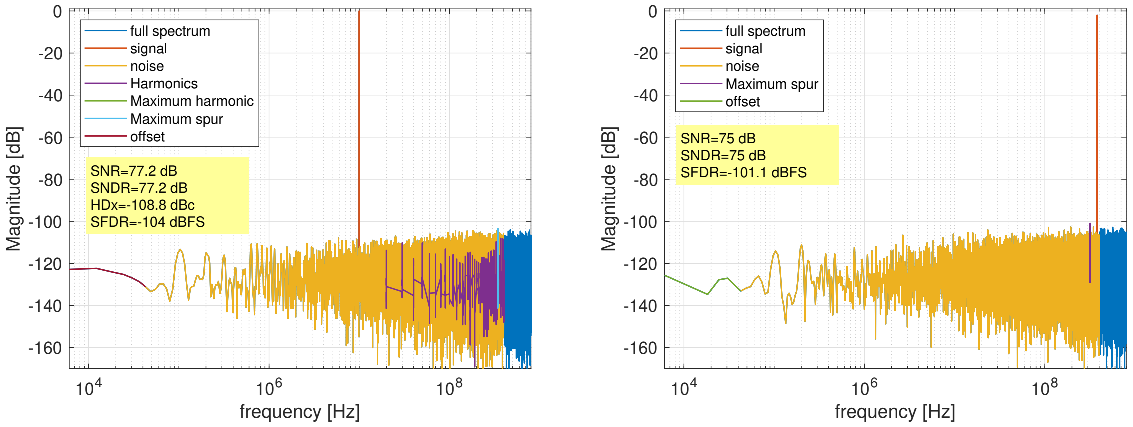
Figure 2.16 shows the output of the CTP. The coarse DACs have DRZ waveform with $t_r = 10\,ps$ and $tf = 12\,ps$ given to each RZ waveform. The RZ halves and the unit elements of the DACs are identical and all the coarse DACs are identical.



(a) DRZ scheme

(b) DRZ waveform with $t_r = 10ps$ and $t_f = 12ps$ in each RZ half

Figure 2.15: DRZ scheme and waveform of 1 bit DAC



(a) $F_{in} = 10\ MHz$

(b) $F_{in} = 380\ MHz$

Figure 2.16: Output spectrum of the CTP with DRZ coarse DACs. Each RZ half have $t_r = 10ps$ and $tf = 12ps$. DAC elements are identical

2.2.3. DAC Element Mismatch

DAC unit elements shown in the DAC model in Figure 2.6 are assumed to be identical (equal-valued resistors or current sources). However, during circuit fabrication, mismatch is introduced in the unit elements of the DAC. The mismatch errors cause the DAC output levels to deviate from their nominal values. Figure 2.17 shows the DAC model of including the errors introduced by unit element mismatch. The errors are denoted by e_i and they are assumed to follows a Gaussian distribution with a certain standard deviation σ.

Recall from Equation 2.8 that the output of the DAC can be written as:

$$y(t) = \sum_{i=1}^{2^{N-1}} (2C_i[n] - 1) \cdot p_i(t - nT) \tag{2.8}$$

The DAC errors due to element mismatch can be viewed as introducing constant gain error e_g, constant offset β and an additive error term $e_{dac}(t)$ that is a deterministic non-linear function of the DAC input [17]. The DAC output can be written as follows

$$y(t) = \alpha x[n] + \beta + e_{dac}(t) \tag{2.9}$$

where $\alpha = 1 + eg$. Mismatch errors introduce by DAC elements cause harmonic distortion and leak the quantization noise of the coarse ADCs to the output and therefore degrade the SQNR and the linearity of the CTP ADC. Figure 2.18 shows the output spectrum of the CTP when DAC element mismatch of

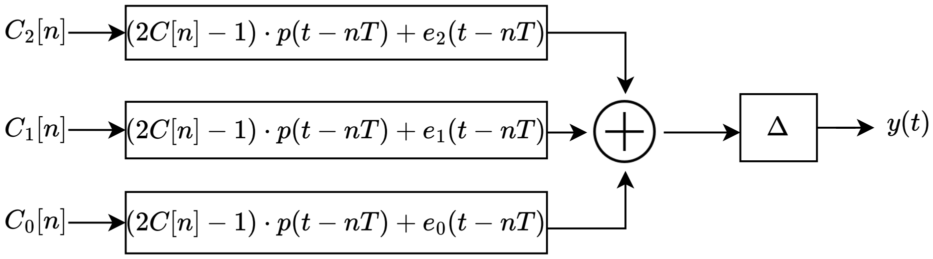
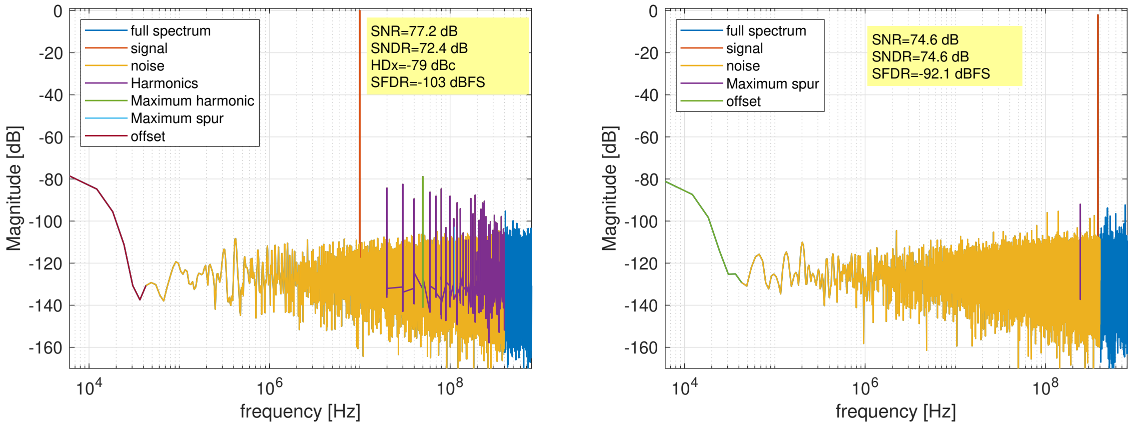

$C_2[n]$
$(2C[n]-1)\cdot p(t-nT)+e_2(t-nT)$
$C_1[n]$
$(2C[n]-1)\cdot p(t-nT)+e_1(t-nT)$
$C_0[n]$
$(2C[n]-1)\cdot p(t-nT)+e_0(t-nT)$
$+$
Δ
$y(t)$

Figure 2.17: 2-bits static model of the DAC including mismatch errors

$\sigma = 0.1\%$ is introduced to all coarse DACs (3-bits, NRZ). Due to mismatch errors, DR is degraded by $\sim 3\ dB$ and $SFDR$ by $\sim 10\ dB$.



(a) $F_{in} = 10\ MHz$

(b) $F_{in} = 380\ MHz$

Figure 2.18: Output spectrum of the CTP with DAC element mismatch of $\sigma = 0.1\%$.

To simplify the analysis and the equations, the DAC model is simplified to the model shown in Figure 2.19, where the DAC pulse is omitted. From unit element mismatch prospective, the models shown in Figure 2.17 and Figure 2.19 are considered to be equivalent [17].

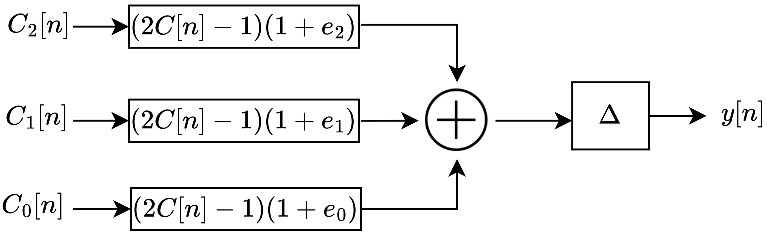


Figure 2.19: Simplified 2-bits static model of the DAC including mismatch errors

Consider $K[n]$ to be the digital input to the DAC, where $k[n] = C_0[n] + C_1[n] + C_2[n]$ for a 2-bits DAC. For example, if $k[n] = 1$, then $[C_0\ C_1\ C_2] = [1\ 0\ 0]$. Similarly, if $k[n] = 2$, then $[C_0\ C_1\ C_2] = [1\ 1\ 0]$. An interesting and important property of unit element DACs is the symmetry properties of the possible DAC outputs. For example, if $K[n] = 1$, this means that there are three different possibilities to produce the DAC output. One option is $[C_0\ C_1\ C_2] = [1\ 0\ 0]$. Another valid option is $[C_0\ C_1\ C_2] = [0\ 1\ 0]$. Finally, the last option is $[C_0\ C_1\ C_2] = [0\ 0\ 1]$. All possible outputs of a 2-bits DAC are illustrated in Figure 2.20. Note that for $k = 0$ and $k = 3$ the DAC output is unique and for all other inputs there are three possible values. As illustrated in Figure 2.20, If the same unit elements are always used to produce the output of the DAC for a certain input $K[n]$, the DAC output values will not lie on a straight line and hence the

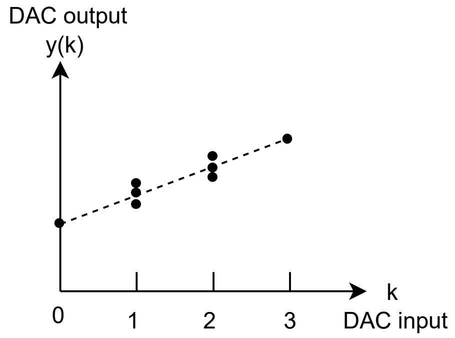


Figure 2.20: Possible outputs of 2-bits DAC represented by dots. The dashed line represent the average

DAC transfer function is non-linear.

Many techniques exist to linearize the transfer function of the DAC, such as dynamic element matching (DEM), in which the DAC elements are selected randomly for each input code to break the correlation between the DAC error and its input [18]. As a result, the average value of all possible values lie on the same linear line connecting the points $k = 0$ and $k = 3$ as illustrated by the dashed line in Figure 2.20. If the input $k = 1$, the possible DAC outputs can be written as

$$y(k = 1) = (-1 + e_0 - e_1 - e_2) \cdot \Delta \tag{2.10}$$

$$y(k = 1) = (-1 - e_0 + e_1 - e_2) \cdot \Delta \tag{2.11}$$

$$y(k = 1) = (-1 - e_0 - e_1 + e_2) \cdot \Delta \tag{2.12}$$

In DEM, the DAC elements are selected randomly. Therefore, the average output value for $k = 1$ is

$$\bar{y}(k = 1) = -\left(1 + \frac{e_0 + e_1 + e_2}{3}\right) \cdot \Delta \tag{2.13}$$

Similarly, for input $k = 2$, the possible DAC outputs are given as follows:

$$y(k = 2) = (+1 + e_0 + e_1 - e_2) \cdot \Delta \tag{2.14}$$

$$y(k = 2) = (+1 + e_0 - e_1 + e_2) \cdot \Delta \tag{2.15}$$

$$y(k = 2) = (+1 - e_0 + e_1 + e_2) \cdot \Delta \tag{2.16}$$

The average output value for $k = 2$ is

$$\bar{y}(k = 2) = +\left(1 + \frac{e_0 + e_1 + e_2}{3}\right) \cdot \Delta \tag{2.17}$$

The unique DAC output values for the inputs $k = 0$ and $k = 3$ are

$$\bar{y}(k = 0) = -3\left(1 + \frac{e_0 + e_1 + e_2}{3}\right) \cdot \Delta \tag{2.18}$$

$$\bar{y}(k = 3) = +3\left(1 + \frac{e_0 + e_1 + e_2}{3}\right) \cdot \Delta \tag{2.19}$$

Based on Equations 2.13, 2.17, 2.18 and 2.19, The average transfer function of the DAC is linear. The residual gain error can be expressed as:

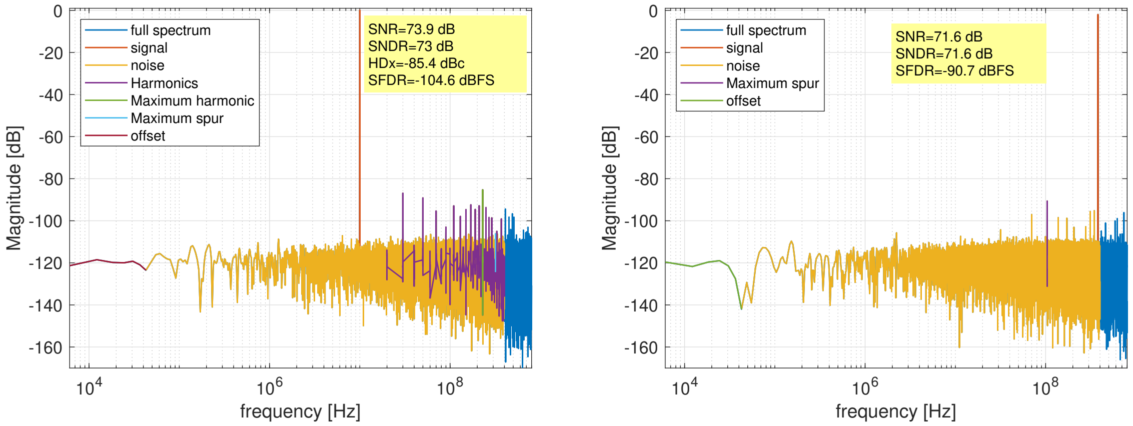
$$e_g = \frac{e_0 + e_1 + e_2}{3} \tag{2.20}$$

The DEM algorithm can also be described by representing the DAC unit elements in an array. The DAC unit elements of the first stage can be represented as follows:

$$DAC_1 = \begin{bmatrix} U_{1,1} & U_{1,2} & U_{1,3} \end{bmatrix} \tag{2.21}$$

Where $U_{1,1}$ represents the first unit element of the fist coarse DAC. Similarly, $U_{1,2}$ represents the second unit element of the first coarse DAC. Similar notation can be written for the DACs of the remaining stages.If the digital input of the first DAC is $k_1[n] = K_1$, then the output of the DAC is built up by selecting K_1 unit element randomly from the DAC array. For example, if $k_1[n] = 1$ then one of the unit elements is randomly selected and connected to the high bit C_0. Therefore, the remaining unit elements will be connected to the low bits C_1 $C2$. If $k_1[n] = 2$, then two randomly selected unit elements will be connected to the high bits C_0 C_1. the remaining element will be connected to the low bit C_2. If $k_1[n] = 0$ then all bit are low. This is denoted as non of the unit elements are selected. Finally, if $k_1[n] = 3$, then all unit elements are selected to connect with all high bits C_0 C_1 C_2. To summarize, the DEM algorithm can be represented by randomly shuffling the columns of the DAC array shown in Equation 2.21. DEM algorithm circuitry can be implemented by having a digital encoder that randomly shuffles the input bits of the DAC by means of random generator, multiplexers and other digital circuitry. More details regarding the implementation can be found in [17, 19].

As a result of the random selection of the unit elements, the harmonic distortion introduced by the DAC (e_{dac}) is transformed into white noise. Nonetheless, the error power lying within the signal band will increase the noise floor of the converter. Figure 2.21 shows the output spectrum of the CTP for $\sigma = 0.1\%$ and DEM applied to the coarse DACs (3-bits,NRZ) of all three stages. The DEM technique linearizes the DACs. However, the resulted gain errors of the coarse DACs leak (e_{g1}, e_{g2} and e_{g3} for coarse stage 1, 2 and 3 respectively) the quantization errors of the front-end ADCs to the output and therefore, reduces the linearity and dynamic range of the converter. Figure 2.22 shows the histogram of HDx and $SFDR$ for different cases. The first case is when no DEM is applied. The second case is when DEM is applied to the first stage only. The third case is when DEM is applied to the first and the second stages. The last case is when DEM is applied to all coarse stages. Based on the results, it can be concluded the residual gain errors of the coarse DACs are limiting the linearity performance of the CTP ADC. Furthermore, Figure 2.23 shows the $SNDR$ and SNR. Applying the DEM technique reduces SNR and $SNDR$ by $\approx 2.5dB$ for $\sigma = 0.1\%$.

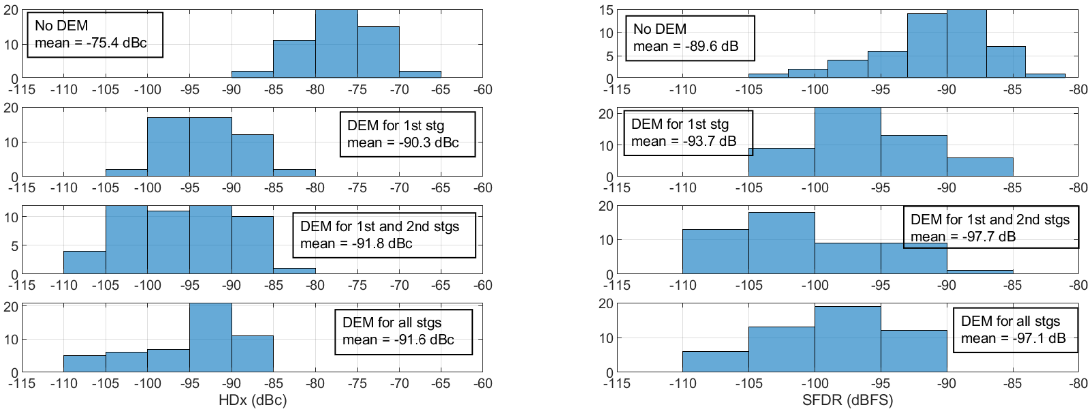


(a) $F_{in} = 10\,MHz$

(b) $F_{in} = 380\,MHz$

Figure 2.21: Output spectrum of the CTP with DAC element mismatch of $\sigma = 0.1\%$ and DEM applied to all DACs

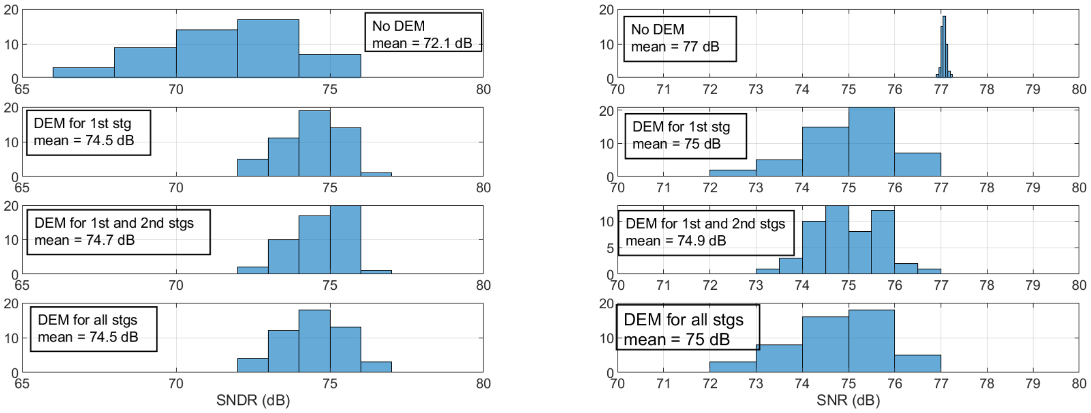
As discussed in section 2.2.1, the residual gain errors of the coarse DACs may be estimated and calibrated by the digital compensation filters $H_i(z)$. Since the DAC element mismatch errors are generated by MATLAB, the values of residual gain errors can be extracted and fed directly into the digital filters as shown in Figure 2.5. Note that this calibration is used as an ideal reference and it's results are compared with the proposed calibration presented in 3.



(a) $F_{in} = 10\ MHz$

(b) $F_{in} = 380\ MHz$

Figure 2.22: HDx and $SFDR$ of the CTP with DEM applied. DAC element mismatch of $\sigma = 0.1\%$ is introduced to all DACs

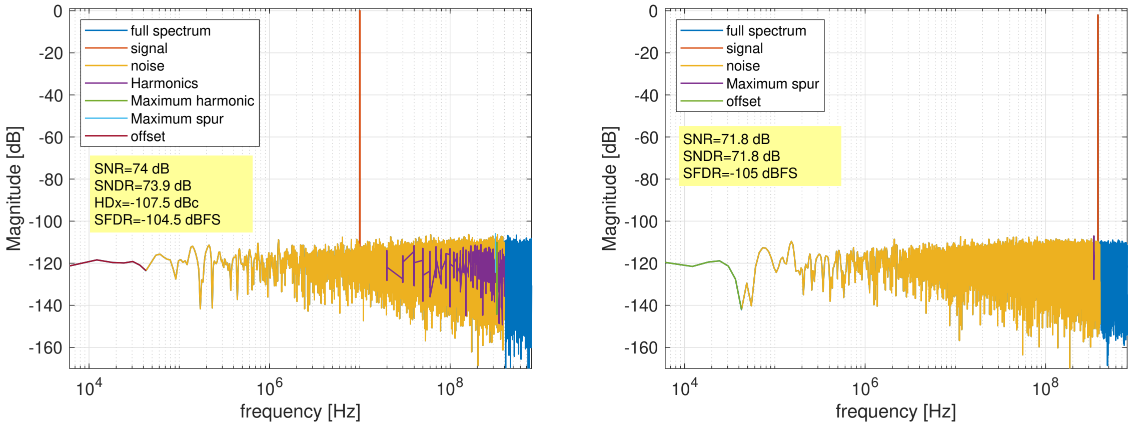


(a) $F_{in} = 10\ MHz$

(b) $F_{in} = 10\ MHz$

Figure 2.23: SNR and $SNDR$ of the CTP with DEM applied. DAC element mismatch of $\sigma = 0.1\%$ is introduced to all DACs

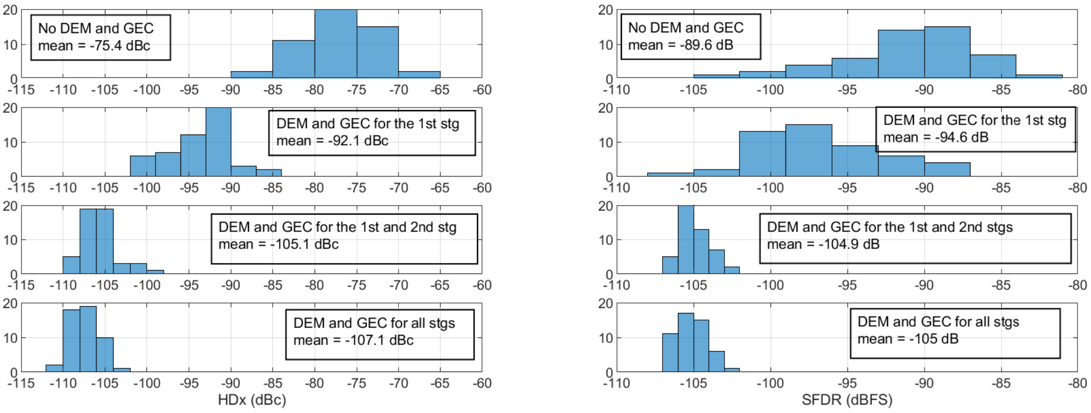
Figure 2.24 shows the output spectrum of the CTP with DEM and the ideal gain error calibration (GEC) applied to all coarse DACs. As expected no harmonic distortion or spurs are present at the output and the linearity of the converter is solely limited to the noise floor. Figure 2.25 shows the histogram of HDx ans $SFDR$ when DEM and GEC are applied. DAC element mismatch of $\sigma = 0.1\%$ is introduced to all DACs. Furthermore, Figure 2.26 shows the $SNDR$ and the SNR. Based on the results, it is necessary to apply DEM and calibrate the residual gain errors at least of the first two coarse stages in order to meet $SFDR$ specification. The linearity of last coarse stage is less important than the first two stages since its input resembles white noise. Since the DAC distortion is turned into white noise, the $SNDR$ and SNR drop by $\approx 2dB$ for $\sigma = 0.1\%$. This σ value is realistic for device matching that can be achieved in CMOS $28\ nm$ technology.



(a) $F_{in} = 10\,MHz$

(b) $F_{in} = 380\,MHz$

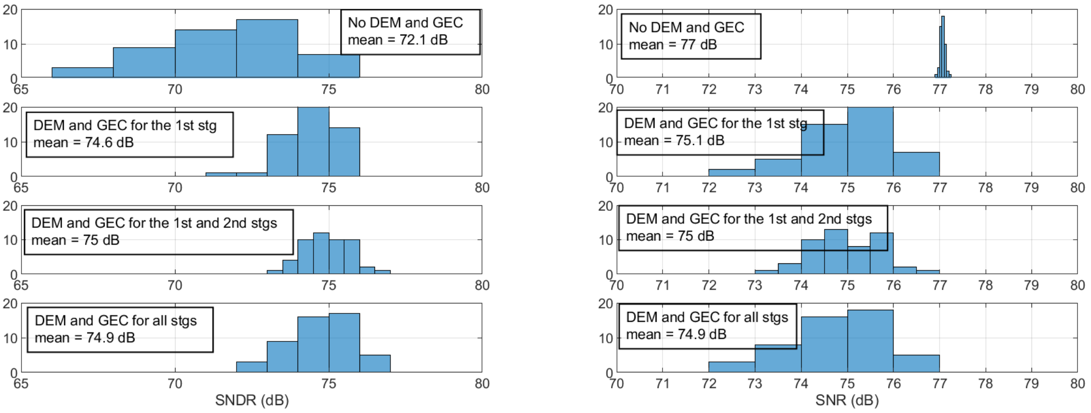
Figure 2.24: Output spectrum of the CTP with DEM and ideal GEC applied to all DACs. DAC element mismatch of $\sigma = 0.1\%$.



(a) $F_{in} = 10\,MHz$

(b) $F_{in} = 380\,MHz$

Figure 2.25: HDx and $SFDR$ of the CTP with DEM and ideal GEC. DAC element mismatch of $\sigma = 0.1\%$ is introduced to all DACs



(a) $F_{in} = 10\,MHz$

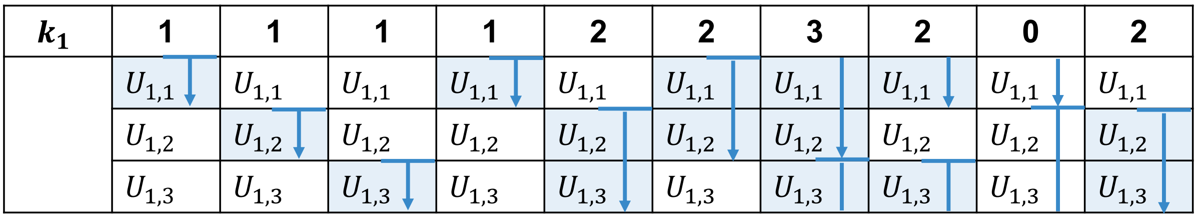
(b) $F_{in} = 10\,MHz$

Figure 2.26: $SNDR$ and SNR of the CTP with DEM and ideal GEC. DAC element mismatch of $\sigma = 0.1\%$ is introduced to all DACs

The SNR and $SNDR$ can be improved by noise shaping the DAC distortion. First order Data Weighted Averaging (DWA) aim to make the term $e_{dac}(t)$ in Equation 2.9 uncorrelated with the DAC input while ensuring that $e_{dac}(t)$ has a first order high pass shape. This means that the power spectral density (PSD) of $e_{dac}(t)$ should be zero at $\omega = 0rad/s$ and that the PSD is free of spurious tones and rises at $20\ dB$/decade as ω increases from zero [17]. It is important to note the general DWA algorithm or any variation of it, aims to linearize the DAC by noise shaping the integral non-linearity (INL) errors. However, after the linearization, the DAC will still exhibit the residual gain error e_g.
Many different methods exist to perform DWA algorithm. The general and direct method to realize DWA is described in [20], where the DAC elements are selected sequentially from the DAC array shown in Equation 2.21. This is represented by a pointer $p(n)$ that points to the next available unused DAC element and is described as follows:

$$p(n) = (p(n-1) + k(n-1))\ mod\left(2^N - 1\right) \tag{2.22}$$

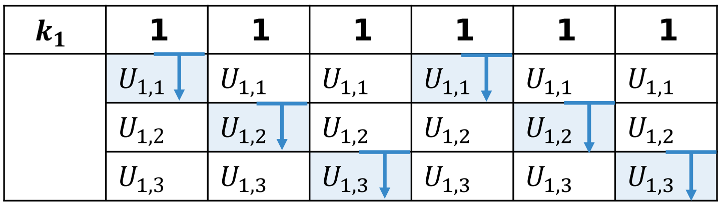
Figure 2.27 shows an example of the operation of DWA with an arbitrary DAC input k_1. The shaded cells indicate the used elements. The pointer is indicated in blue and its arrow points to the next elements to be used.



k_1	1	1	1	1	2	2	3	2	0	2
	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$
	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$
	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$

Figure 2.27: Example of DWA element selection for arbitrary input values

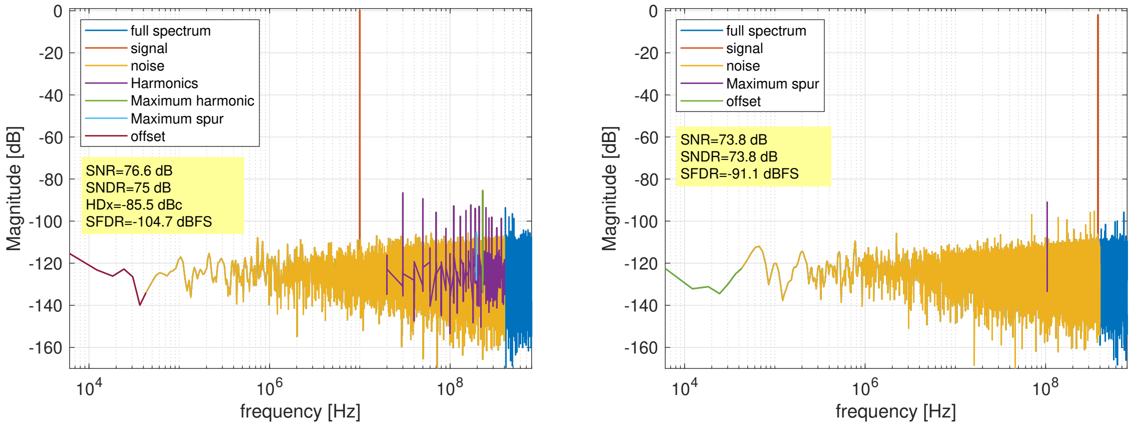
This cyclic selection of the DAC elements ensures that the DAC INL errors quickly sum to zero. Therefore, the DAC distortion moves to high frequencies. However, tone generation is a major problem in the DWA algorithm if the DAC input is not a busy signal but a low amplitude or low frequency tone.For example, if the DAC input is a DC digital signal with value of $k = 1$, then the elements used are shown in Figure 2.28



k_1	1	1	1	1	1	1
	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$	$U_{1,1}$
	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$	$U_{1,2}$
	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$	$U_{1,3}$

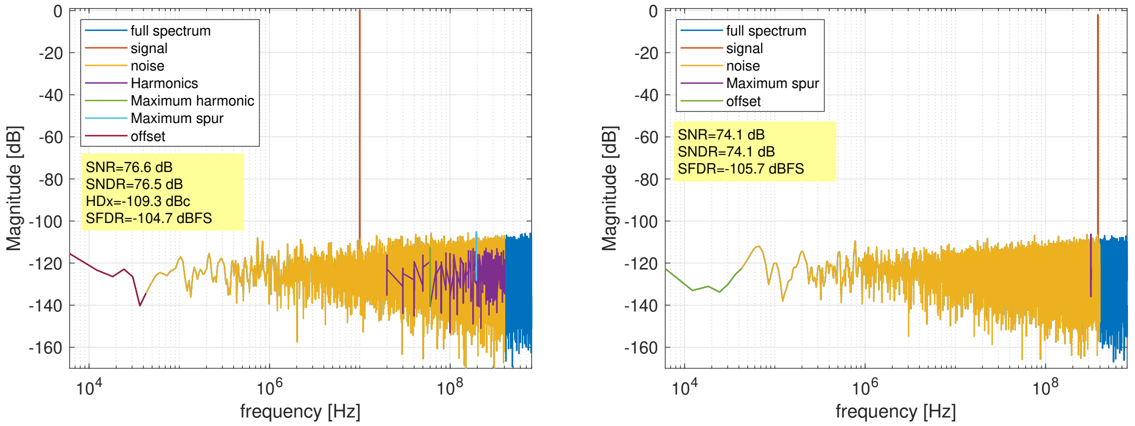
Figure 2.28: Example of DWA element selection for a DC digital input

this makes $e_{dac}[n]$ to be a periodic signal and not a high pass filtered noise. Therefore, the DAC output might show spurious tones and nonlinear artifacts for low input amplitudes and frequencies [20]. This tonal behaviour can be broken by introducing some random effect in the DWA algorithm. There are many publications offering different methods to solve this issue such as Random DWA (RnDWA) and Random Incrementing DWA (RIDWA). For details the reader may refer to [21–25]. It is important to note the general DWA algorithm or any variation of it, aims to linearize the DAC by noise shaping the INL errors However, after linearization, DAC will exhibit a linear gain error e_g.
Figure 2.29 shows the output spectrum of the CTP with DWA. Similar to the DEM, the HDx and $SFDR$ are limited by the DAC gain errors. Figure 2.30 shows the output spectrum when applying DWA and ideal GEC. Finally, Figure 2.31 shows the Monte Carlo simulations at $\sigma = 0.1\%$. In the worst case, $SNDR$ and SNR are improved by $\sim 4.5dB$ compared with the DEM technique.



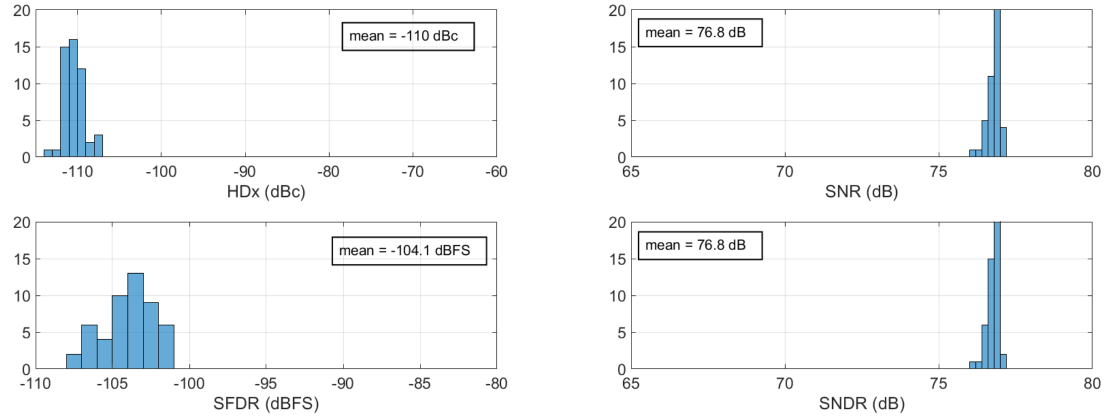
(a) $F_{in} = 10\ MHz$ (b) $F_{in} = 380\ MHz$

Figure 2.29: Output spectrum of the CTP with DAC element mismatch of $\sigma = 0.1\%$ and DWA applied to all DACs



(a) $F_{in} = 10\ MHz$ (b) $F_{in} = 380\ MHz$

Figure 2.30: Output spectrum of the CTP with DAC element mismatch of $\sigma = 0.1\%$ and DWA with ideal GEC applied to all DACs



(a) HDx at $F_{in} = 10MHz$ and $SFDR$ at $F_{in} = 380MHz$ (b) SNR and $SNDR$ at $F_{in} = 10MHz$

Figure 2.31: Monte Carlo simulations of the CTP with DWA and ideal GEC applied to all DACs. DAC element mismatch of $\sigma = 0.1\%$ is given to all DACs

2.3. Summary

In this chapter, the CTP ADC is introduced and the coarse DACs of the CTP are investigated. This is done by means of MATAB simulations. The non-idealities of the coarse DACs are modeled and integrated into the CTP MATLAB model.

Simple NRZ DACs are prone to inter-symbol interference (ISI). While return-to-zero (RZ) DACs can effectively mitigate ISI, they may saturate the ampifier or the subsequent stages. To address this issue, we can utilize a dual return-to-zero (DRZ) DAC, which combines the benefits of both NRZ and RZ DACs. Furthermore, mismatch errors in the DAC unit elements degrade the linearity and the dynamic range of the CTP. Based on the analysis, it is concluded that at least the first two coarse DACs in the CTP ADC need to be calibrated. This includes the INL and the residual gain errors due to element mismatch. Conventional calibration techniques such as DEM or DWA calibrate the INL errors. However, the residual gain errors still need to be calibrated.

3
DAC Calibration

Many techniques exist to linearize the transfer function of the DAC. Dynamic element matching (DEM), in which the distortion due mismatch in DAC unit elements is transformed into white noise. Nonetheless, the error power lying within the signal band will increase the noise floor of the converter. Data weighted averaging technique aim to reduce the inband noise by noise shaping the DAC mismatch distortion. DAC element selection algorithms linearize the DAC under the assumption that the resulted linear gain error is acceptable. This might be valid in many applications. However, in CTP ADC, the linear gain error of the coarse DACs leaks the quantization errors of the coarse ADCs to the output and therefore, degrades the linearity performance of the converter. Two new innovative calibration techniques are proposed in this chapter. The techniques aim to linearize the coarse DACs and minimize the quantization error leakage due to the residual gain errors of the DACs without the need for estimating the their values in the digital domain. The work presented in this chapter has resulted in two patent filings.

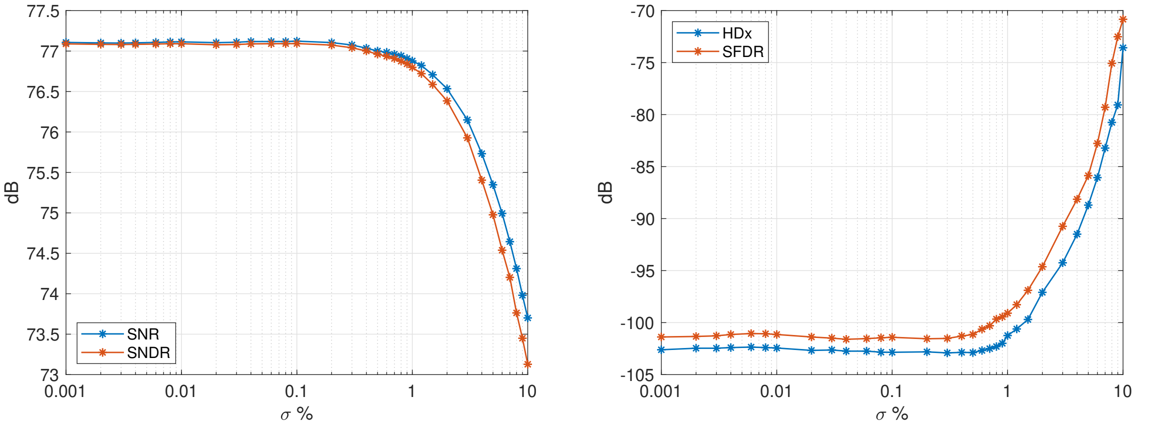
The mathematical analysis and equations presented in this chapter, are based on 4 coarse stages, each stage has a resolution of 2-bits. The analysis can easily be extended to N-bits and K (>1) coarse stages.
It is observed that the leakage of the quantization noise due to the residual DAC gain errors in Equation 2.2 can be minimized if the DAC gain errors of the front-end stages are all equal $e_g = e_{g1} = e_{g2} = e_{g3} = e_{g4}$. Then Equation 2.2 reduces to:

$$\begin{aligned} V_{out} = {} & G^4 V_{in}\left(1 - e_g + e_g^2 - e_g^3 + e_g^4\right) \\ & + G^4 Q_1 e_g^4 \\ & - G^3 Q_2 e_g^3 \\ & + G^2 Q_3 e_g^2 \\ & - G Q_4 e_g \\ & + Q_{BE} \end{aligned} \tag{3.1}$$

This analysis is also confirmed by MATLAB simulations. Figure 3.1 shows the case when the coarse DACs have the same gain error e_g. The gain errors are drawn from the standard Gaussian distribution with standard deviation of σ. Compared with the previous results shown in Figure 2.4, it can be concluded that if the coarse DACs share the same gain errors, the quantization error leakage is minimized. The performance of the CTP does not degrade up to $\sigma = 1\%$.

The proposed calibration techniques are based on the idea of making the linear gain errors of the coarse DACs equal on average, while also calibrating the harmonic distortion of the DACs. Unlike the conventional DEM, where only the harmonic distortion is converted into white noise, the proposed Advanced Dynamic Element Matching (ADEM) technique aims to translate both the harmonic distortion and the residual gain errors of the coarse DACs into white noise. The second proposed technique Advanced

Data Weighted Averaging (ADWA) aims to noise shape both the harmonic distortion and the residual gain errors of the coarse DACs. The techniques are described in the following sections.



(a) SNR and SNDR at $F_{in} = 10\ MHz$ (b) HDx at $F_{in} = 10\ MHz$ and SFDR at $f_{in} = 380\ MHz$

Figure 3.1: CTP performance when coarse DACs have the same gain error. Each simulation point is the average of 50 runs

3.1. Advanced Dynamic Element Matching

As discussed in 2.2.3, due to DAC element mismatch, the DAC will exhibit a gain error e_g that can not be eliminated by the DEM technique. Consider a CTP ADC with 4 coarse stages where each stage has a resolution of 2-bits. Assume that e_{gi} is the DAC gain error of the coarse stage i. The DAC unit elements of the coarse stages are combined in an array as follows

$$DAC_{array} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} U_{1,1} & U_{1,2} & U_{1,3} \\ U_{2,1} & U_{2,2} & U_{2,3} \\ U_{3,1} & U_{3,2} & U_{3,3} \\ U_{4,1} & U_{4,2} & U_{4,3} \end{bmatrix} \tag{3.2}$$

Where, the first row represent the elements of the first coarse DAC and the second row represent the elements of the second DAC, etc. In the conventional DEM, if the DAC input of the first stage is $k_1[n] = K_1$, then the DAC output of the first coarse stage is built up by randomly selecting K_1 unit elements from the first row in DAC_{array}. Similarly, if the DAC input of the second coarse stage is $k_2[n] = K_2$, then the DAC output of the second stage is built up by using the first K_2 unit elements from the second row in DAC_{array}. Similar statements hold for stage 3 and 4. This operation can be viewed as randomly shuffling the columns of the DAC array. Figure 3.2 shows a 2-stage CTP with conventional DEM. The conventional DEM block shuffles the output of the coarse ADC randomly. This can be done by using multiplexers and a random generator. Then the shuffled output is applied to the DAC. This is equivalent to shuffling the columns of the DAC array.

However, in the proposed ADEM technique, the output of the first coarse DAC is built up by randomly selecting K_1 unit elements from the DAC array. This means that the output of the first coarse DAC can be built up by using any unit elements in the DAC array without any restriction of using the first row only. Similarly, the output of other coarse DACs can by constructed by using any unit elements from the DAC array.

One way of performing the proposed DEM algorithm is to randomly shuffle all unit elements in the DAC_{array} and then built up the DAC output of stage i by using the elements that appear in row i after the randomization. This will ensure that the INL and gain errors of the DACs are converted into white noise. Another simpler way to implement the proposed DEM is to use the conventional DEM to swap the columns of the DAC array. Then, the rows of the DAC array are swapped by means of a random

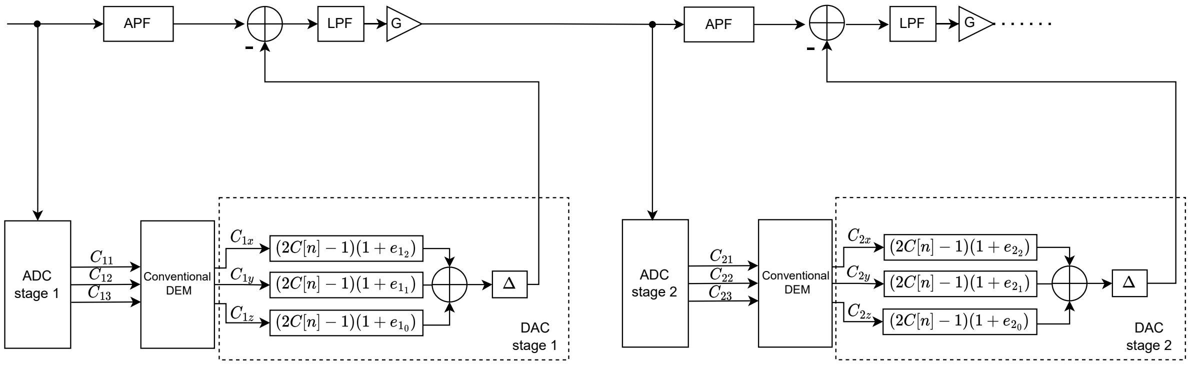


Figure 3.2: Simplified diagram of 2-stages CTP with conventional DEM

signal Rnd and switches as illustrated in Figure 3.3.

By randomly swapping the columns in the DAC array, the INL errors of each corresponding row are turned into white noise. This will result a gain error e_{gi} for each row i. Therefore, after randomizing the columns, each row i of the DAC array can be represented by unit elements that have a gain error e_{gi}. All possible outputs of a DAC i with gain error e_{gi} can be written as

$$y(ki=0) = -3\left(1+e_{gi}\right)\cdot\Delta \tag{3.3}$$

$$y(ki=1) = -1\left(1+e_{gi}\right)\cdot\Delta \tag{3.4}$$

$$y(ki=2) = +1\left(1+e_{gi}\right)\cdot\Delta \tag{3.5}$$

$$y(ki=3) = +3\left(1+e_{gi}\right)\cdot\Delta \tag{3.6}$$

Note that if the DAC has only gain error, there is only one possible output for each digital input. When swapping the rows of the DAC array in a random fashion, the average output of each row in the DAC array can be written as

$$\bar{y}(ki=0) = -3\left(1+\frac{e_{g1}+e_{g2}+e_{g3}+e_{g4}}{4}\right)\cdot\Delta \tag{3.7}$$

$$\bar{y}(ki=1) = -1\left(1+\frac{e_{g1}+e_{g2}+e_{g3}+e_{g4}}{4}\right)\cdot\Delta \tag{3.8}$$

$$\bar{y}(ki=2) = +1\left(1+\frac{e_{g1}+e_{g2}+e_{g3}+e_{g4}}{4}\right)\cdot\Delta \tag{3.9}$$

$$\bar{y}(ki=3) = +3\left(1+\frac{e_{g1}+e_{g2}+e_{g3}+e_{g4}}{4}\right)\cdot\Delta \tag{3.10}$$

This means that the average outputs of all coarse DACs lie on the same line and therefore, the DAC transfer of each coarse stage is linear with a gain error e_g that is equal to

$$e_g = \frac{e_{g1}+e_{g2}+e_{g3}+e_{g4}}{4} \tag{3.11}$$

Since swapping the rows is done randomly, the gain errors e_{gi} are also turned into white noise. More precisely, the errors $e_{gi}-e_g$ are converted to white noise. Finally, the proposed ADEM algorithm can be easily extended to N-bits and x coarse stages ($x>1$). Furthermore, depending on the system behaviour and linearity specifications, the proposed ADEM technique can be applied to the first rows only instead of all rows in the DAC array.

Figure 3.4 shows the output spectrum of the CTP when the proposed ADEM technique is applied to all DACs. As expected, no harmonic distortion or spurs appear at the output and HDx and $SFDR$ are limited to the noise floor.

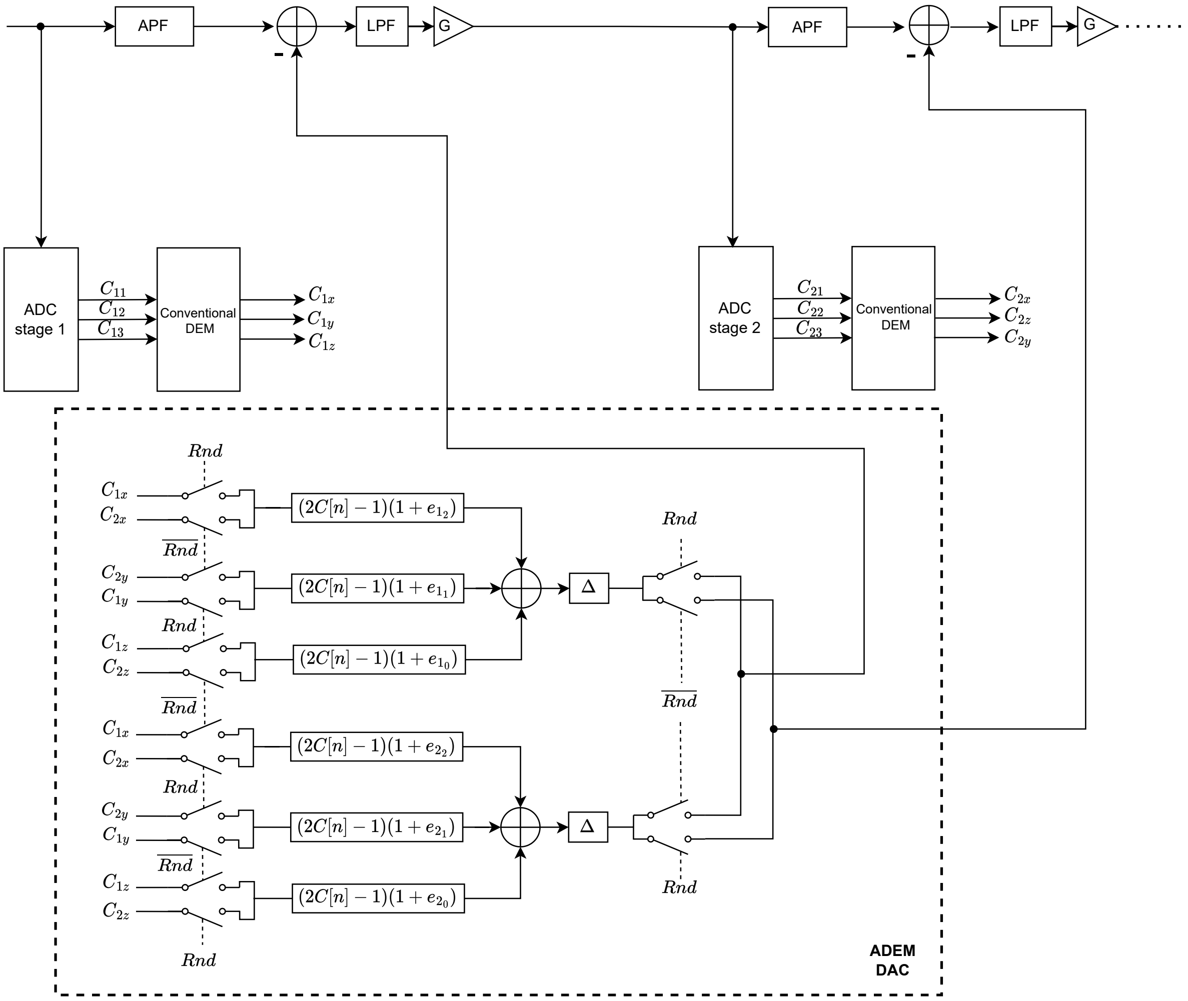
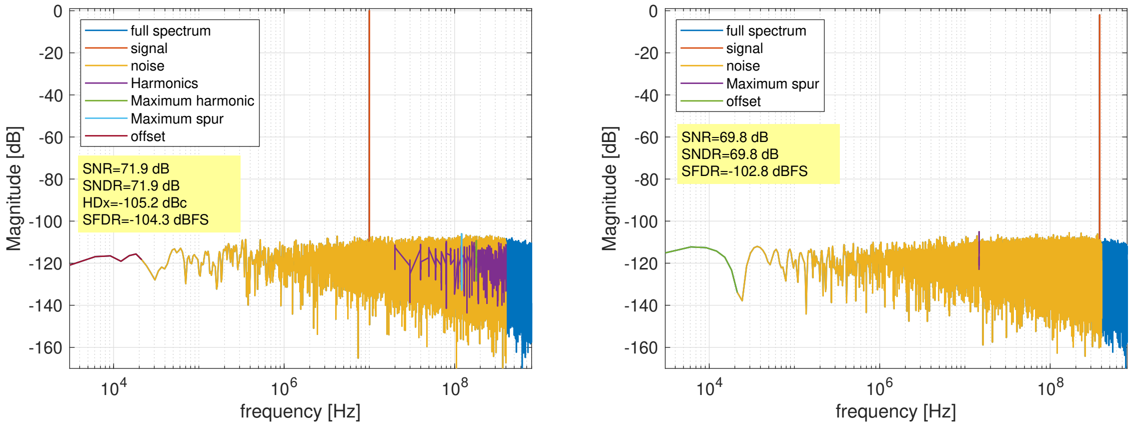


Figure 3.3: Simplified diagram of 2-stages CTP with advanced DEM

Figure 3.5 and 3.6 show the Monte Carlo simulation of the proposed DEM in comparison with the conventional DEM and the conventional DEM with ideal GEC. Element mismatch σ is introduced to all DACs. The calibration schemes are applied to all coarse stages. The proposed DEM technique clearly improves the HDx and $SFDR$ performance compared with the conventional DEM. Furthermore, the HDx and $SFDR$ of the proposed DEM are comparable with those of the DEM with ideal GEC. The $SNDR$ and SNR of the proposed DEM are slightly lower than the SNR and $SNDR$ of the conventional DEM. This is expected since both the INL and gain errors of the coarse DACs are converted into white noise. However, in the conventional DEM with ideal GEC, only the INL errors are converted into white noise while the gain errors are canceled. Table 3.1 shows a comparasion of the results for $\sigma = 0.1\%$.

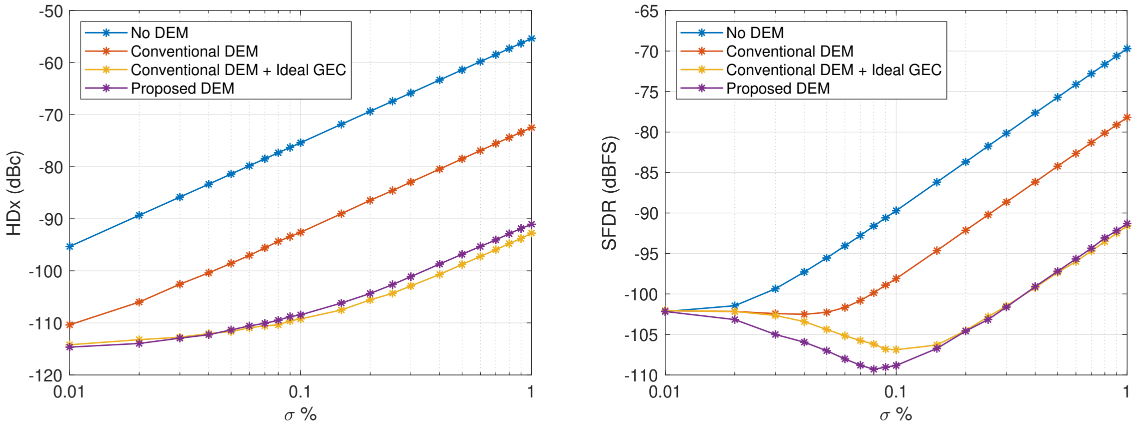
Table 3.1: Monte Carlo results for $\sigma = 0.1\%$

	SNR (dB)	SNDR (dB)	HDx (dBc)	SFDR (dBFS)
No DEM	77.1	72.1	-75.4	-89.7
Conventional DEM	74.6	74.4	-92.5	-98.1
Conventional DEM + ideal GEC	74.4	74.6	-109.3	-106.9
Proposed DEM	73.5	73.5	-108.5	-108.8



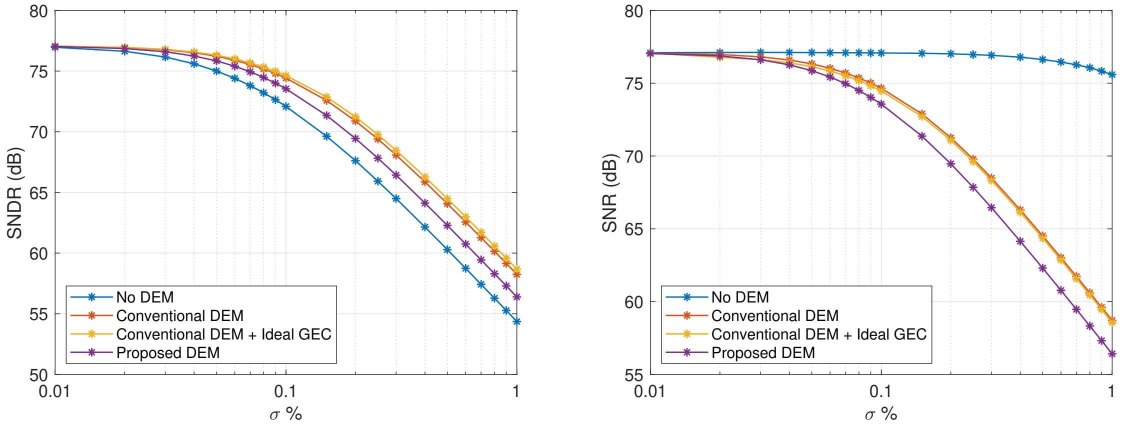
(a) $F_{in} = 10\ MHz$ (b) $F_{in} = 380\ MHz$

Figure 3.4: Output spectrum of the CTP when the proposed ADEM technique is applied to all DACs. Element mismatch of $\sigma = 0.1\%$ is introduced to all DACs.



(a) HD_x at $F_{in} = 10\ MHz$ (b) $SFDR$ at $F_{in} = 380\ MHz$

Figure 3.5: HDx and $SFDR$ Monte Carlo simulations of the CTP. Each simulation point is the average of 50 runs



(a) $SNDR$ at $F_{in} = 10\ MHz$ (b) SNR at $F_{in} = 10\ MHz$

Figure 3.6: $SNDR$ and SNR Monte Carlo simulations of the CTP. Each simulation point is the average of 50 runs

3.2. Advanced Data Weighted Averaging

The proposed ADWA technique aims to noise shape both the INL and gain errors of the DACs in order to improve the $SNR/SNDR$. Consider the DAC array shown in Equation 3.2. The columns of the DAC array are swapped in a specific manner to achieve first order noise shaping of the INL errors. Furthermore, the rows of the DAC array are swapped in a specific manner to achieve first order noise shaping of the gain errors. The DAC output of the first stage is built up by using the elements that appear in the first row after swapping the rows and the columns. Similarly, the DAC output of the second stage is built up by using the elements that appear in the second row in the DAC array, etc. In order to simplify the analysis, firstly the noise shaping of the INL errors is discussed assuming that the rows of the DAC array are swapped. Thereafter, the noise shaping of the gain errors is discussed assuming that the columns are swapped.

For the 2-bits DAC shown in Figure 2.19, the possible INL errors for in input $k = 1$ can be written as

$$INL(k=1) = \frac{1}{3}(+4e_0 - 2e_1 - 2e_2) \cdot \Delta = \mathbf{x_1} \tag{3.12}$$

$$INL(k=1) = \frac{1}{3}(-2e_0 + 4e_1 - 2e_2) \cdot \Delta = \mathbf{x_2} \tag{3.13}$$

$$INL(k=1) = \frac{1}{3}(-2e_0 - 2e_1 + 4e_2) \cdot \Delta = \mathbf{x_3} \tag{3.14}$$

Similarly, for an input $k = 2$, the possible INL errors are

$$INL(k=2) = \frac{1}{3}(+2e_0 + 2e_1 - 4e_2) \cdot \Delta = -\mathbf{x_1} \tag{3.15}$$

$$INL(k=2) = \frac{1}{3}(+2e_0 - 4e_1 + 2e_2) \cdot \Delta = -\mathbf{x_2} \tag{3.16}$$

$$INL(k=2) = \frac{1}{3}(-4e_0 + 2e_1 + 2e_2) \cdot \Delta = -\mathbf{x_3} \tag{3.17}$$

for the inputs $k = 0$ and $k = 3$ the INL errors are zero

$$INL(k=0) = 0 \tag{3.18}$$

$$INL(k=3) = 0 \tag{3.19}$$

Note that

$$\mathbf{x_1} + \mathbf{x_2} + \mathbf{x_3} = \mathbf{0} \tag{3.20}$$

In order to achieve first order noise shaping, the columns of the DAC array are swapped such that the sum of the INL errors of each row is equal to zero as quickly as possible. This will move the INL distortion to higher frequencies. In order to maintain the noise shaping of the INL errors while the rows are swapped, we have to put these errors in a hold state before they can be summed to zero. One possible realization is to employ S pointers for each stage, where S denotes the number of the stages. For example, if we have 2 coarse stages and each stage has 2-bits resolution and assume that the rows of the DAC array are swapped randomly, then the element selection of the columns of both stages, for arbitrary inputs, is illustrated in Figures 3.7 and 3.8. The pointers are denoted in blue and green. The general concept is that we have to keep track of all the INL errors that appear in each stage. A more general representation of the switching scheme of the DAC array columns is shown in Figures 3.9 and 3.10. This figure shows the Finite State Machine (FSM) that represent the switching scheme of the pointers used in the first stage for a 2 stages CTP and each stage has 2-bits resolution. Similar FSM can be derived for the second stage. The switching scheme aim to make the sum of the INL errors equal to zero as quickly as possible. Note that one of the pointers must always be in a hold state while the other pointer is active such that the two stages can not use the same elements at the same clock cycle. This switching scheme can be easily extended to S stages ($S > 1$) and N bits. This noise shaping scheme ensures that the INL errors of each row is noise shaped at all stages. This results,

residual gain errors e_{gi} for each stage. Note that it is possible to apply this scheme for certain rows only and not all rows.
Since it is possible to noise shape the INL errors of the columns regardless of how the rows are swapped, we can distinguish many possibilities to swap the rows. Consider the CTP ADC with 4 coarse stages where each stage has a resolution of 2-bits. Then, there are 5 different options to swap the rows as shown in Table 3.2. In all options, the INL errors of all coarse stages are noise shaped.

Table 3.2: Gain errors NS options for 4 coarse stages CTP

	Gain errors NS order	**Remark**
Option 1	0	Gain errors of all stages is converted into white noise.
Option 2	1	Noise shaping of gain errors at the first coarse stage. Gain errors of other stages are converted into white noise
Option 3	2	Noise shaping of gain errors at the first and second stages. Gain errors of other stages is converted into white noise.
Option 4	3	Noise shaping of gain errors at the first three coarse stages. Gain error of the fourth stage is converted into white noise.
Option 5	4	Noise shaping of the gain errors at all coarse stages.

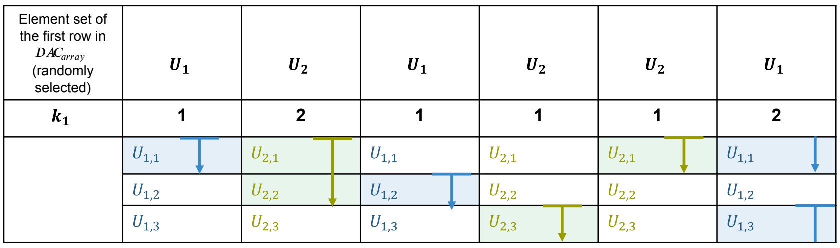


Figure 3.7: Example of element selection for the first stage according to the proposed DWA

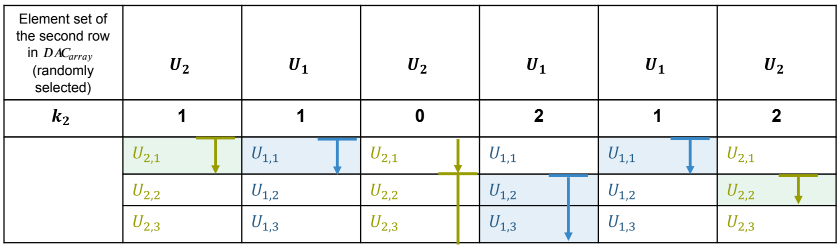
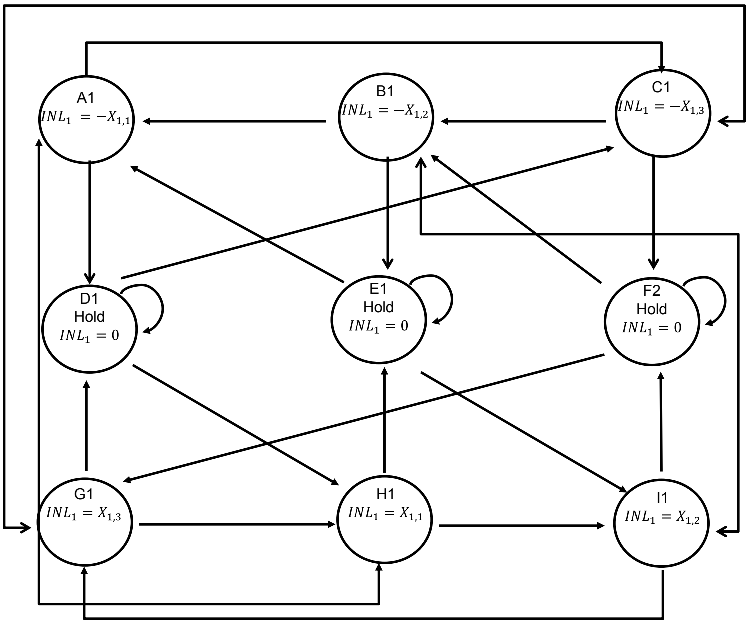


Figure 3.8: Example of element selection for the second stage according to the proposed DWA

Option 1:
In this option the rows of the DAC array are swapped randomly. This is similar to the proposed DEM technique in the previous section 3.1. The difference is that the columns are swapped according to the INL noise shaping scheme shown in Figures 3.9 and 3.10. This options is expected to have better SNR than the proposed DEM. Compared with the different options, this option is expected to have lower SNR.

Option 2:
In this option, the objective is to noise shape the gain errors at the first stage and converting the gain errors at other stages into white noise. More precisely, we want to make the average gain error at all stages equal to the desired value e_g given in Equation 3.11 while noise shaping the errors that deviate from this value. When the elements set U_1 is selected for any row, the error to be noise shaped at that row is $GE = e_{g1} - e_g$. Similarly, if the element set is U_2 is selected, the error is $GE = e_{g2} - e_g$. Based on the selected set of elements and the DAC input, the gain errors that need to be noise shaped can be written as follows:



Element set of the first row in $\mathbf{DAC_{array}}$	**Input $\mathbf{k_1}$**	**State**
U_1	1	G1,H1,I1
U_1	2	A1,B1,C1
U_1	0 or 3	D1,E1,F1
U_2	-	D1,E1,F1

Figure 3.9: FSM of the first pointer used in stage 1. The left FSM is for the first pointer. Similar FSM can be drived for other stages

For input $k = 0$

$$GE(k=0) = 3 \cdot \left(\frac{1}{4} \left(-3e_{g1} + e_{g2} + e_{g3} + e_{g4} \right) \cdot \right) \Delta = -3\mathbf{E_1} \tag{3.21}$$

$$GE(k=0) = 3 \cdot \left(\frac{1}{4} \left(+e_{g1} - 3e_{g2} + e_{g3} + e_{g4} \right) \cdot \right) \Delta = -3\mathbf{E_2} \tag{3.22}$$

$$GE(k=0) = 3 \cdot \left(\frac{1}{4} \left(+e_{g1} + e_{g2} - 3e_{g3} + e_{g4} \right) \cdot \right) \Delta = -3\mathbf{E_3} \tag{3.23}$$

$$GE(k=0) = 3 \cdot \left(\frac{1}{4} \left(+e_{g1} + e_{g2} + e_{g3} - 3e_{g4} \right) \cdot \right) \Delta = -3\mathbf{E_4} \tag{3.24}$$

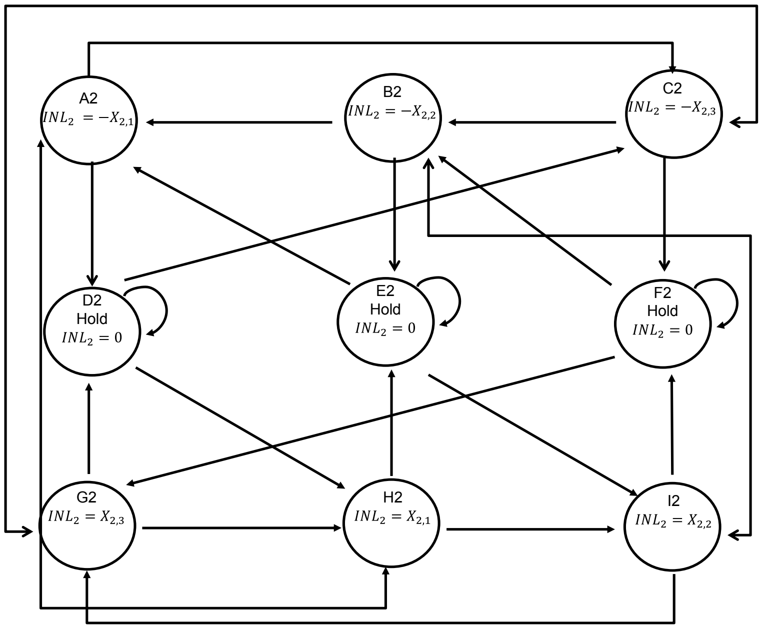
For input $k = 1$

$$GE(k=1) = \left(\frac{1}{4} \left(-3e_{g1} + e_{g2} + e_{g3} + e_{g4} \right) \cdot \right) \Delta = -\mathbf{E_1} \tag{3.25}$$

$$GE(k=1) = \left(\frac{1}{4} \left(+e_{g1} - 3e_{g2} + e_{g3} + e_{g4} \right) \cdot \right) \Delta = -\mathbf{E_2} \tag{3.26}$$

$$GE(k=1) = \left(\frac{1}{4} \left(+e_{g1} + e_{g2} - 3e_{g3} + e_{g4} \right) \cdot \right) \Delta = -\mathbf{E_3} \tag{3.27}$$

$$GE(k=1) = \left(\frac{1}{4} \left(+e_{g1} + e_{g2} + e_{g3} - 3e_{g4} \right) \cdot \right) \Delta = -\mathbf{E_4} \tag{3.28}$$



Element set of the first row in $\mathbf{DAC_{array}}$	Input $\mathbf{k_1}$	State
U_2	1	G2,H2,I2
U_2	2	A2,B2,C2
U_2	0 or 3	D2,E2,F2
U_1	-	D2,E2,F2

Figure 3.10: FSM of the second pointer used in stage 1. Similar FSM can be drived for other stages

For input $k = 2$

$$GE(k=2) = \left(\frac{1}{4}\left(+3e_{g1} - e_{g2} - e_{g3} - e_{g4}\right)\cdot\right)\Delta = +\mathbf{E_1} \tag{3.29}$$

$$GE(k=2) = \left(\frac{1}{4}\left(-e_{g1} + 3e_{g2} - e_{g3} - e_{g4}\right)\cdot\right)\Delta = +\mathbf{E_2} \tag{3.30}$$

$$GE(k=2) = \left(\frac{1}{4}\left(-e_{g1} - e_{g2} + 3e_{g3} - e_{g4}\right)\cdot\right)\Delta = +\mathbf{E_3} \tag{3.31}$$

$$GE(k=2) = \left(\frac{1}{4}\left(-e_{g1} - e_{g2} - e_{g3} + 3e_{g4}\right)\cdot\right)\Delta = +\mathbf{E_4} \tag{3.32}$$

For input $k = 3$

$$GE(k=3) = 3\cdot\left(\frac{1}{4}\left(+3e_{g1} - e_{g2} - e_{g3} - e_{g4}\right)\cdot\right)\Delta = +3\mathbf{E_1} \tag{3.33}$$

$$GE(k=3) = 3\cdot\left(\frac{1}{4}\left(-e_{g1} + 3e_{g2} - e_{g3} - e_{g4}\right)\cdot\right)\Delta = +3\mathbf{E_2} \tag{3.34}$$

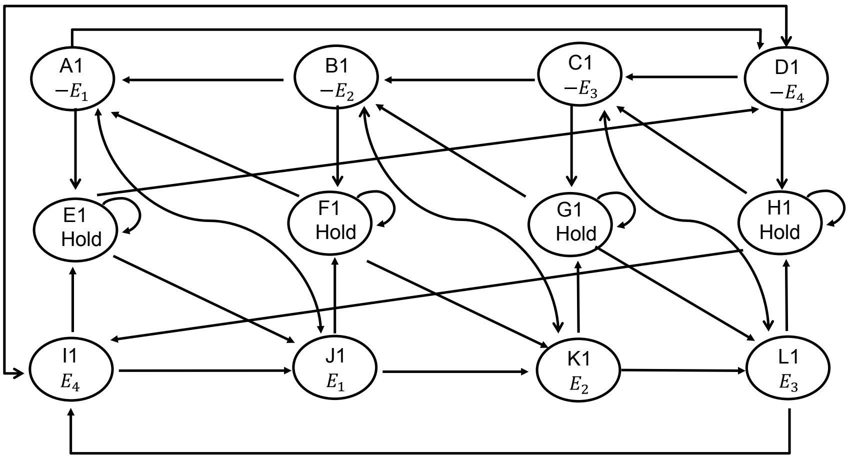
$$GE(k=3) = 3\cdot\left(\frac{1}{4}\left(-e_{g1} - e_{g2} + 3e_{g3} - e_{g4}\right)\cdot\right)\Delta = +3\mathbf{E_3} \tag{3.35}$$

$$GE(k=3) = 3\cdot\left(\frac{1}{4}\left(-e_{g1} - e_{g2} - e_{g3} + 3e_{g4}\right)\cdot\right)\Delta = +3\mathbf{E_4} \tag{3.36}$$

E_1 represent the error if the selected set of elements is U_1 and E_2 is the error if U_2 is selected, etc. For example, if the input of the first stage DAC is $k_1 = 1$ and based on the row swapping technique the element set U_3 is selected for the first DAC then the error that appear at the first stage $-\mathbf{E_3}$. Similarly, assume the the element set U_2 is selected for the fourth stage, and an arbitrary input of the fourth stage is $k_4 = 3$, then the error at the fourth stage is $+3\mathbf{E_2}$. Note that

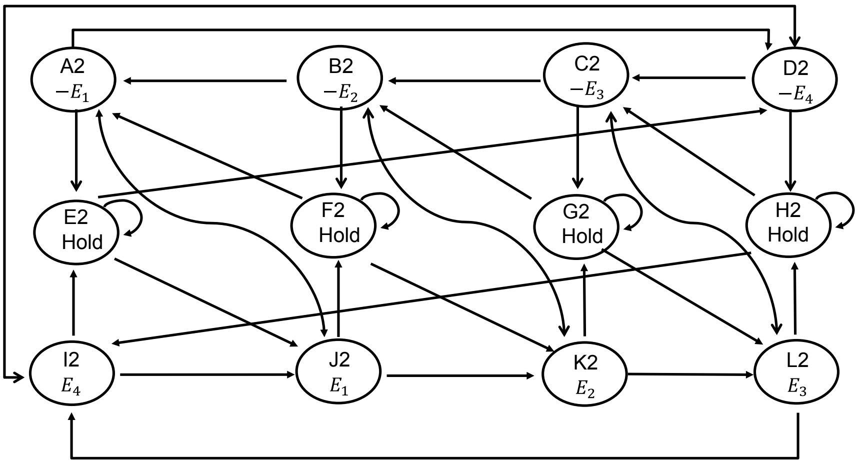
$$\mathbf{E_1} + \mathbf{E_2} + \mathbf{E_3} + \mathbf{E_4} = \mathbf{0} \tag{3.37}$$

We can achieve first order noise shaping of the gain errors at the first stage by selecting the sets of elements to be used at the first stage based on the input of the first stage such that the sum of the errors equal zero as fast as possible ($E_1 + E_2 + E_3 + E_4 = 0$). In other words, we want to swap the rows of the DAC array based only on the digital input of the first stage such that Equation 3.37 is satisfied. This can be achieved by employing two pointers. The first pointer is used when the input of the first stage is $k_1 = 1$ or $k_1 = 2$. The second pointer is used when $k_1 = 0$ or $k_1 = 3$. . The pointers decide which row should be used to produce the output of the first stage, i.e., which set of elements will appear at the first row in the DAC_{array}. the remaining other rows will be swapped randomly to convert their errors into white noise at the remaining stages. The working principle of the pointers is described by the FSMs shown in Figures 3.11 and3.12.



State	input
A1, B1, C1 D1	1
I1, J1, K1, L1	2
E1, F1, G1, H1	0 or 3

Figure 3.11: FSM for the first pointer in option 2



State	input
A2, B2, C2, D2	0
I2, J2, K2, L2	3
E2, F2, G2, H2	1 or 2

Figure 3.12: FSM for the second pointer in option 2

When the input of the first stage is $k_1 = 1$ or $k1 = 2$, then the first pointer is activated, while the second pointer must be in one of the “hold” states. Similarly, if the input is $k_1 = 0$ or $k_1 = 3$ then the second pointer leaves the hold state and the first pointer goes to a hold state. Table 3.3 shows an example of the elements set selection of the first stage based on the FSM diagrams and arbitrary input values. The unit elements sets of the remaining rows in the DAC_{array} will be assigned randomly.

Option 3:

In option 1,the first row in DAC_{array} is chosen based on the value of the first stage input. The other rows are swapped randomly. This results first order noise shaping of the gain errors at the first stage. However, at other stages the gain errors are turned into white noise. In this option, the objective is to noise shape the gain errors of the first and second stages, while converting the gain errors at the third

Table 3.3: Example of the pointers in option 2 of the proposed DWA

First stage input k_1	2	2	3	2	3	2
First pointer state	J1	K1	G1 (hold)	L1	H1 (hold)	I1
Second pointer state	E2 (hold)	E2 (hold)	J2	F2 (hold)	K2	G2 (hold)
First row elements set	U_1	U_2	U_1	U_3	U_2	U_4

and fourth stages into white noise. In order to noise shape the errors at the second stage, we have to select the set of elements that appears at the second row in DAC_{array} based on the input of the second stage. Applying the same switching scheme of option 1 to the second stage is not possible because it might happen that the same set of elements are chosen for the first and the second rows, which is not feasible.

In option 1, the purpose is to add the errors to zero as soon as possible. However, one is not restricted to choose the set of elements according to the finite stage machines in Figures 3.11 and 3.12 only. The pointer represented by the finite state machine is a simple way to achieve noise shaping, but is not the only way. The pointer represented by the finite state machine is a simple way to achieve noise shaping, but is not the only way.

For example, consider the FSM shown in Figure 3.11 and assume the state J1 is initially chosen when the input of the DAC is 2. If the next input is 2, we are not restricted to choose state K1. We can choose any state from the remaining states (K1, L1, I1). Assume, state L1 has been chosen randomly and the next input is 2, then we have to choose one state from the remaining states (K1, I1). We can continue this until all states are chosen and the error sums to zero. Once this happens, we can start again and choose one state from all available states.

Consider another example where the DAC input is 1,1,2,2. Assume the state J1 is initially chosen, the next state could be chosen from the remaining states (K1, L1,I1). Assume I1 is chosen. Then the next state could be chosen from states A1 or D1. If A1 is chosen, then the next state must be D1.

This concept can be used in order to noise shape the gain errors at the first and second stages based on their inputs. Firstly, based on the input of the first stage, we generate all possible states that can be used at the first stage, i.e., all possible set of elements that can be used in the first row in the DAC_{array}. Also, based on the input of the second stage, we generate all possible set of elements that can be used in the second row in the DAC_{array}. Then, we combine all possible options and generate a unique set of elements for the first and the second stages. The remaining rows will swapped randomly. For illustration, a high-level block diagram is given in Figure 3.13

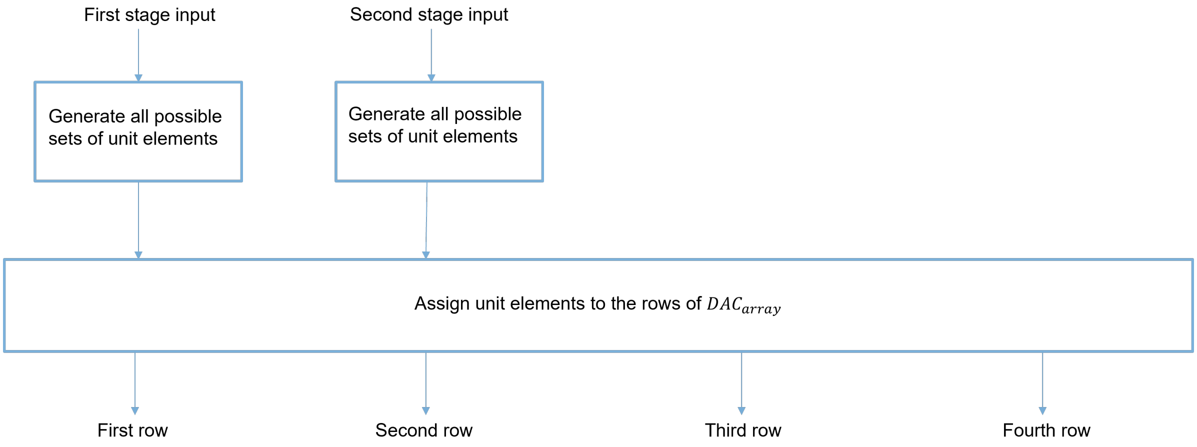
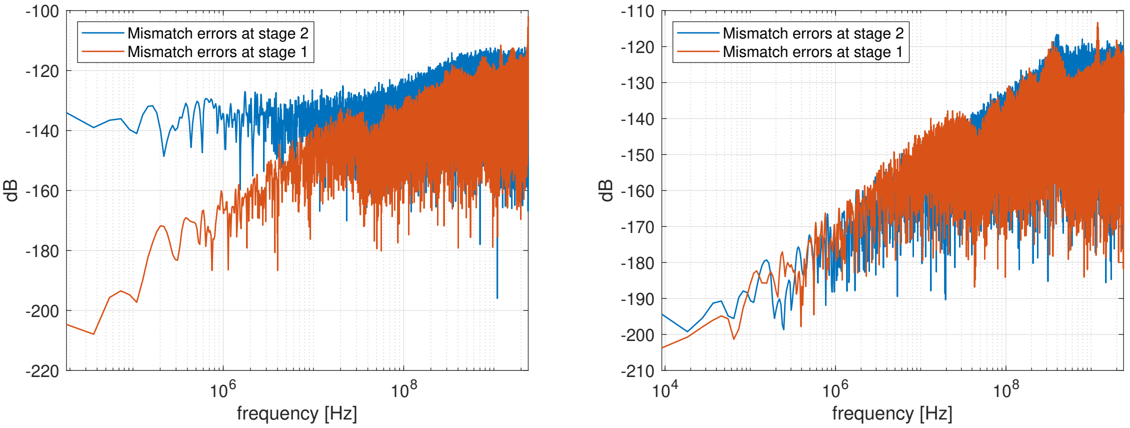

First stage input
Second stage input
Generate all possible sets of unit elements
Generate all possible sets of unit elements
Assign unit elements to the rows of DAC_{array}
First row
Second row
Third row
Fourth row

Figure 3.13: High-level block diagram for option 3 of the proposed DWA

For example, assume that unit elements set U_2 or U_4 can be used in the first row of DAC_{array} based on the first stage input. Also, assume that only the elements set U_2 can be used in the second row based on the second stage input. Then, we choose elements set U_4 for the first row and elements set U_2 for the second row. Furthermore, the elements in the third and fourth rows will be chosen randomly from

elements sets U_1 and U_3.
However, it is possible that there is only one unique set of unit elements that needs to be used in the first and the second stage. For example, the set U_1 is the only possible solution that need to be used in the first and second rows to maintain the noise shaping for both stages. In this case, we choose this set for the first stage and we choose random unit element sets for the all other stages. In other words, we always give priority to the first stage as it is more important than the second stage. This will affect the noise shaping at the second stage and the noise floor increases. Figure 3.14 shows an example of the mismatch errors spectrum at stage 1 and 2 for options 2 and 3.



(a) Option 2

(b) Option 3

Figure 3.14: Example of the mismatch errors spectrum at stages 1 and 2

Option 4 and 5:

These options can be realized by extending the same concept discussed in option 2 to include the third and fourth stages. This is illustrated in the high-level block diagram shown in Figure 3.15. The best results can be obtained if we give priority to the first stages in a similar manner as described in option 3.

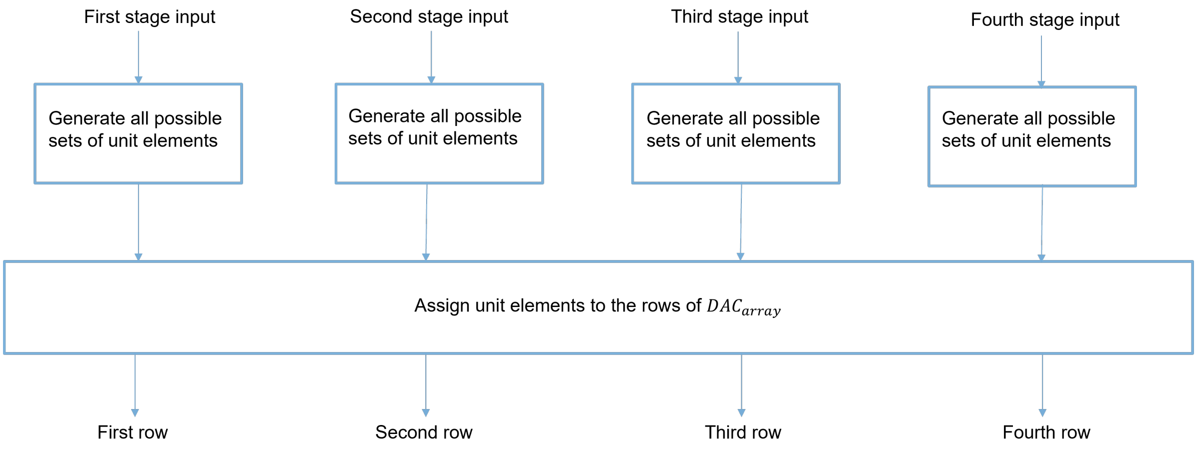

First stage input
Second stage input
Third stage input
Fourth stage input
Generate all possible sets of unit elements
Generate all possible sets of unit elements
Generate all possible sets of unit elements
Generate all possible sets of unit elements
Assign unit elements to the rows of DAC_{array}
First row
Second row
Third row
Fourth row

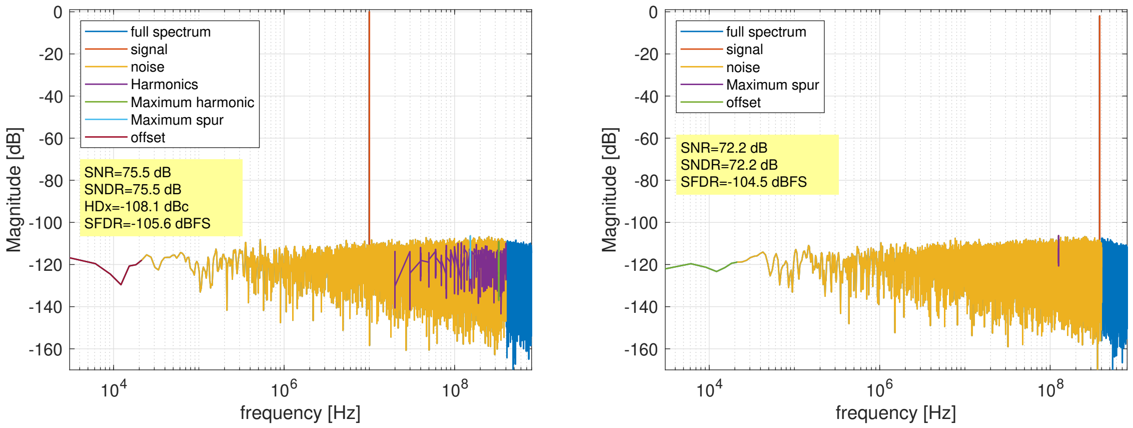
Figure 3.15: High-level block diagram for options 4 and 5 of the proposed DWA

Next, simulations of a 3 coarse stages CTP are presented. Each coarse stage has resolution of 3-bits. This results 4 different options. In option 1, gain errors at all coarse stages are converted into white noise. In option 2, the gain errors at the first stage is noise shaped while the gain errors at the other stages are translated into white noise. Similarly, in option 3 the gain errors at the first and the second stages are noise shaped while the gain error at the last coarse stage is translated into white noise. Finally, in option 4, the gain errors at all coarse stages are noise shaped. Note that in all the options, the INL errors at all coarse stages are noise shaped.

Figure 3.16 shows the output spectrum when the proposed DWA technique option 2 is applied to all DACs. Figures 3.18 and 3.17 show the Monte Carlo simulations. The results are summarized in Table 3.4 for $\sigma = 0.1\%$. Comparing the different options, option 2 seems to offer the optimal results as options 3 and 4 do not noticeably improve beyond option 2.

Table 3.4: Summary of Monte Carlo results for $\sigma = 0.1\%$

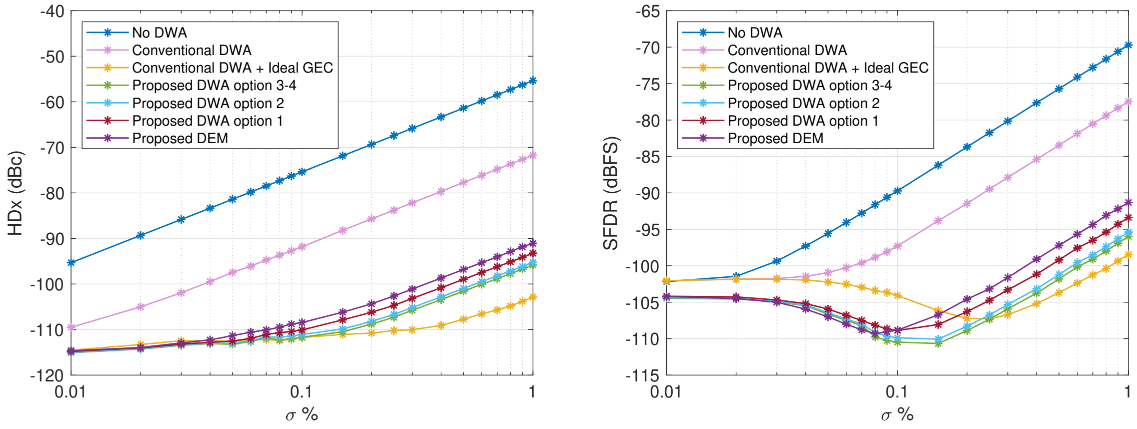
	SNR (dB)	SNDR (dB)	HDx (dBc)	SFDR (dBFS)
No DWA/DEM	77.1	77.2	-75.4	-89.7
Conventional DEM	74.6	74.4	-92.5	-98.1
Conventional DEM + ideal GEC	74.4	74.6	-109.3	-106.9
Conventional DWA	76.7	76.3	-91.8	-97.3
Conventional DWA + ideal GEC	76.8	76.7	-111.7	-104.1
Proposed DEM	73.5	73.5	-108.5	-108.8
Proposed DWA option 1	74.8	74.8	-110.1	-108.8
Proposed DWA option 2	76.1	76.1	-110.3	-109.5
Proposed DWA option 3	76.3	76.3	-110.5	-110
Proposed DWA option 4	76.4	76.4	-110.5	-110



(a) $F_{in} = 10\ MHz$

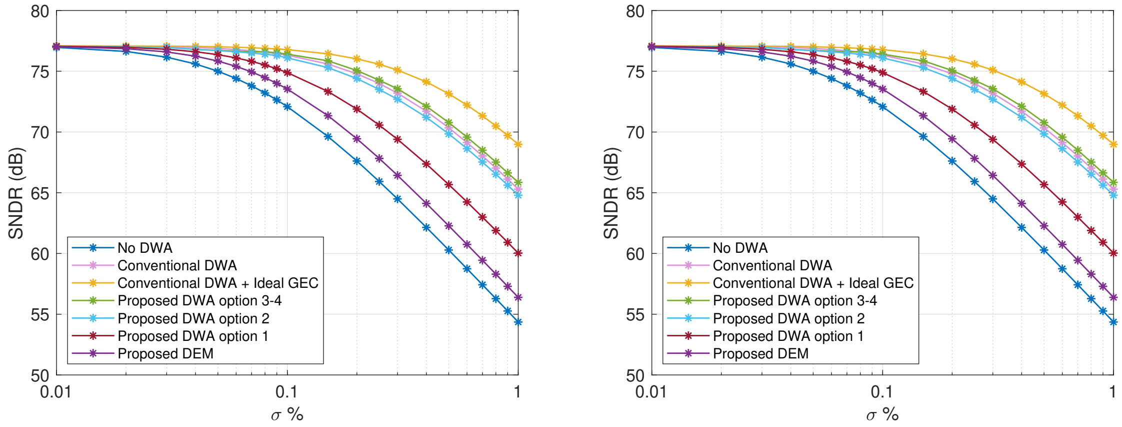
(b) $F_{in} = 380\ MHz$

Figure 3.16: Output spectrum of the CTP when the proposed DWA technique option 2 is applied. Element mismatch of $\sigma = 0.1\%$ is introduced to all DACs.



(a) HD_x at $F_{in} = 10\ MHz$ (b) $SFDR$ at $F_{in} = 380\ MHz$

Figure 3.17: HDx and $SFDR$ Monte Carlo simulations of the CTP with different calibration schemes. Element mismatch σ is introduced to all DACs. The calibration schemes are applied to all coarse stages. Each simulation point is the average of 50 runs



(a) $SNDR$ at $F_{in} = 10\ MHz$ (b) SNR at $F_{in} = 10\ MHz$

Figure 3.18: $SNDR$ and SNR Monte Carlo simulations of the CTP with different calibration schemes. Element mismatch σ is introduced to all DACs. The calibration schemes are applied to all coarse stages. Each simulation point is the average of 50 runs

3.3. Summary

In this chapter, two new innovative calibration techniques were presented. The first technique, advanced DEM, converts the DAC distortion and gain errors into white noise. This improves the linearity of the CTP ADC significantly compared with the conventional DEM. However, the improvement in the linearity resulted in degradation of the SNR and $SNDR$. In order to account for this degradation advanced DWA technique is introduced where the DAC distortion and gain errors are noise shaped. This improves the SNR and $SNDR$ at the cost of complexity.

4 DAC Implementation

A natural way to implement the DAC is to make use of the virtual ground provided by the amplifier by utilizing resistive DACs (R-DACs). R-DACs tend to add minimal thermal noise, which makes the converters that use them highly efficient in terms of power consumption. Additionally, the layout of these DACs is typically uncluttered as a result of the straightforward nature of the unit element [26]. However, R-DAC loads the virtual ground of the amplifier and therefore reduce the amplifier's loop gain. This might degrade the linearity of the amplifier.

A different approach instead of depending on the virtual ground is to utilize current steering DACs (CS-DACs). CS-DACs do not load the virtual ground node and therefore the amplifier loop gain is higher. This improves the linearity of the amplifier. However, CS-DACs have higher noise contribution than R-DACs when supplying the same current. The noise spectral density of CS-DACs is at least $3\ dB$ worse than R-DACs [26].

Other distinctions between a R-DACs and a CS-DACs pertain to the switches. In R-DACs, the switches are typically in triode region, while in a CS-DACs, the switches operate in saturation region. The drawback for the CS-DACs is that the voltage required to control the gate of the switches may need to be managed to keep the transistors in saturation over PVT variations. Often, additional circuitry is needed to set the gate-drive voltage appropriately [27]. Finally, the sensitivity of R-DACs and CS-DACs to ISI and jitter is identical [26].

To limit the scope of the thesis, it has been decided to investigate and implement R-DAC only. This is due to the low noise contribution of R-DACs. In addition, NXP has already developed a design of a 2-bits Dual-Return-to-Open (DRO) R-DAC in $TSMC28\ nm$ technology at $F_s = 6\ GHz$ and a bandwidth of $120MHz$ for a continuous time delta sigma converter [28]. This design is used as a reference.

As previously mentioned in Chapter 2, it is proposed to utilize DRZ DACs for the initial two stages of the CTP ADC. DRZ DAC is realized by combining two RZ DACs. However, RZ DACs have higher thermal noise spectral density than NRZ DACs. During the half of the clock period where the unit element output goes to zero, the DAC simply injects noise without contributing to the signal component. Therefore, the thermal noise spectral density of the RZ DAC is twice as high as that of its NRZ equivalent. This issue is solved by utilizing DRO R-DAC, where the DAC unit resistor is left floating. As a result, the output DAC current is zero and no noise can be transferred to the output since the unit element is floating.

Firstly, the implementation of the ADEM and ADWA is investigated. in section 4.1 DRZ and DRO R-DAC architectures are proposed. The architectures allows a feasible implementation of the proposed calibration techniques discussed in Chapter 3. However, the proposed DAC and the calibration techniques requires the impedance of the CTP stages to be identical. This reduces the power efficiency of the CTP. Therefore, due to the impedance scaling of the CTP stages, the proposed DAC and its calibration are not implemented and further investigated.

Alternatively, in section 4.2 a different DAC architecture is investigated. With impedance scaling taken into account, a dual return to open resistive DAC is implemented in the first stage of the CTP. The architecture employs conventional DEM technique and is verified within the first stage of the CTP with ideal digital back-end calibration. Moreover, dither and noise shaping were applied to the coarse ADC to linearize the coarse quantizer and to relax the requirements of the digital compensation filters. The 3-bits coarse ADC with noise shaping and dithering achieves linearity equivalent to an 8-bit ADC. As a result, the system can tolerate gain errors of up to 0.28% while still meeting the SFDR specifications.

The test bench used to verify the performance of the DAC for a 1 stage CTP is shown in Figure 4.1. The APF is implemented as an ideal delay element with a resistor $R_{APF} = 200\Omega$. The coarse ADC is implemented as an ideal ADC model in Verilog-A with the required noise shaping and dithering. The LPF of the first stage is integrated into the amplifier. The amplifier provides the virtual ground node and it has open loop gain around $49\ dB$. The closed loop gain is $12\ dB$ and its $3\ dB$ bandwidth is around $940MHz$. Furthermore, $Fs = 4.8\ GHz$ and $Vdd = 0.9\ V$ The amplifier is a real amplifier and is verified to achieve $SFDR < -104dBFS$ in the same test-bench but with ideal NRZ DAC with ideal resistors and ideal switches. Furthermore, for comparison purposes, the DAC is also investigated with a linear amplifier model. The linear amplifier is a small signal model of the real amplifier. However, it has the same input and output impedance and open-loop and closed-loop transfer as the real amplifier. Finally, the data is exported from Cadence and is processed in MATLAB. The coarse stages 2-3 are replaced by LPFs. The back-end ADC is replaced by an ideal sampler at $F_s/3$. The digital filter $H_1(z)$ is an ideal Finite Impulse Response (FIR) filter that estimates the CT transfer function seen from the DAC. The digital filter $H_1(z)$ realizes ideal digital calibration unless mentioned otherwise. The same test bench can be extended to test multiple coarse stages.

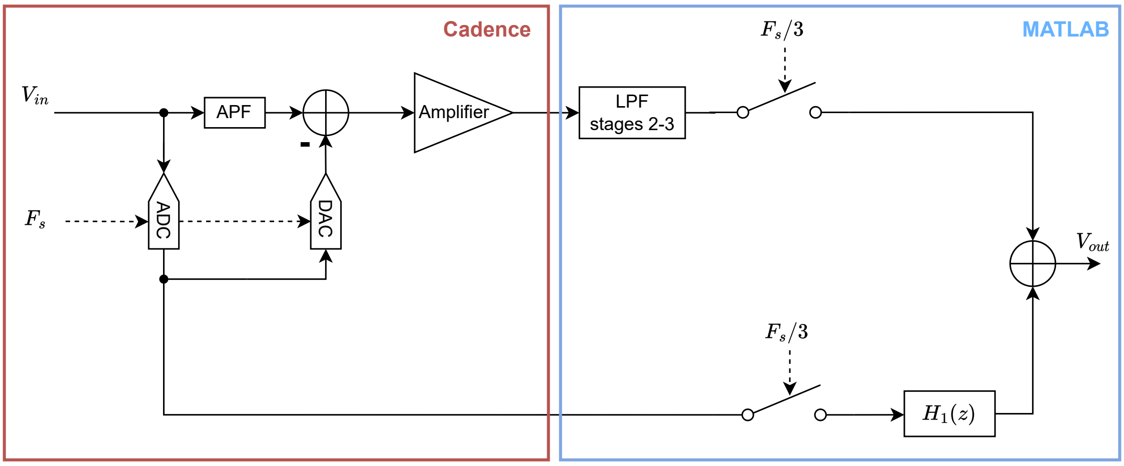


Figure 4.1: Test setup for the DAC in 1 stage CTP

In this test setup, the specifications of the DAC are summarized in

Table 4.1: DAC design specification

Parameter	**Value**	**Unit**
SNDR	75	dB
HDx	-80	dBc
SFDR	-104	dBFS
Thermal noise	2.47	$nV/\sqrt(Hz)$
Maximum input amplitude (Vin)	0.7	V
Supply voltage (Vdd)	0.9	V

4.1. Virtual-Ground-Switched DRZ and DRO R-DAC

The goal of this section is to develop a DAC architecture that allows a feasible implementation of the proposed ADEM and ADWA techniques. Firstly, DRZ and DRO R-DAC architectures are investigated and a DAC architecture is proposed. The proposed DAC can be used in the coarse stages without interchanging the DAC elements of the coarse stages and hence CTP impedance scaling can be used.

Furthermore, conventional DEM can be applied to each DAC separately. Thereafter, the same DAC architecture is extended to allow the implementation of the proposed ADEM and ADWA techniques in 2-stages CTP without impedance scaling.

Figure 4.2 shows a virtual-ground-switched NRZ R-DAC unit cell [29], where the resistors are connected to the supply. The node I_{op} and I_{on} connects to the virtual ground of the amplifier. The digital data is denoted by D and $\overline{D}$. As motivated in [29], this architecture addresses the issue of the supply parasitic resistors. Supply parasitic resistors result in data dependent current drawn from the supply and therefore, harmonic distortion. However, this is not an issue in this architecture since the nodes 1 and 2 between the resistors and switches do not change when there is data transition [29]. This is however only true with an ideal amplifier.

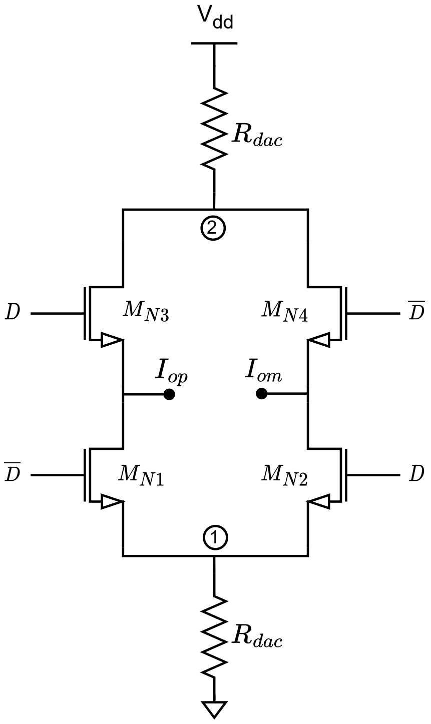


Figure 4.2: DAC unit cell [29]

The objective is to modify the DAC unit cell in Figure 4.2 to develop a DRZ/DRO DAC unit cell. The proposed DRZ unit cell is shown in Figure 4.3. The clock clk is high during half T_s. The switches M_{N9},M_{N10},M_{N11} and M_{N12} are connected to a reference voltage $V_{cm} = 0.5V_{dd}$. The switches utilize the DRZ DAC. If these switches are removed, the resistors are left floating during a half period of the clock cycle. Then the unit cell represents DRO DAC. To illustrate the differences between the DRZ and DRO DAC, consider node 1 in Figure 4.3. If the clock transitions from high to low, then the switches M_{1N}, M_{2N} are turned off. Therefore, without M_{N9}, the voltage at node 1 will go from $\approx 0.5V_{dd}V$ to $\approx 0\,V$. Once the clock is high again, node 1 needs to recharge again from $0\ V$ to $0.5V_{dd}V$. As a result, the speed benefit of this architecture is lost since the nodes 1,2,3 and 4 are supposed to stay charged. This causes slow on and off transitions and therefore large glitches. This makes the average DAC current heavily dependent on the glitches and therefore very sensitive to dynamic variations. However, in the DRZ DAC this is not an issue. The RZ switches ensure that the current is always flowing through the resistors R_{dac} and therefore the nodes 1,2,3 and 4 are fixed. Figure 4.4 shows an example of the DAC output current of one unit element.

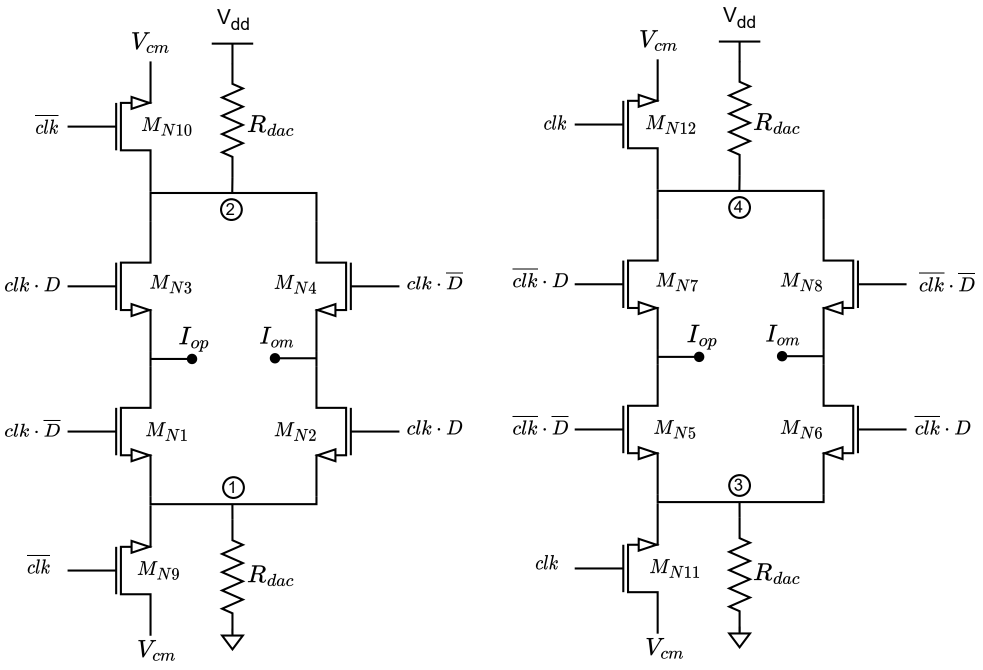


Figure 4.3: Proposed DRZ R-DAC unit cell

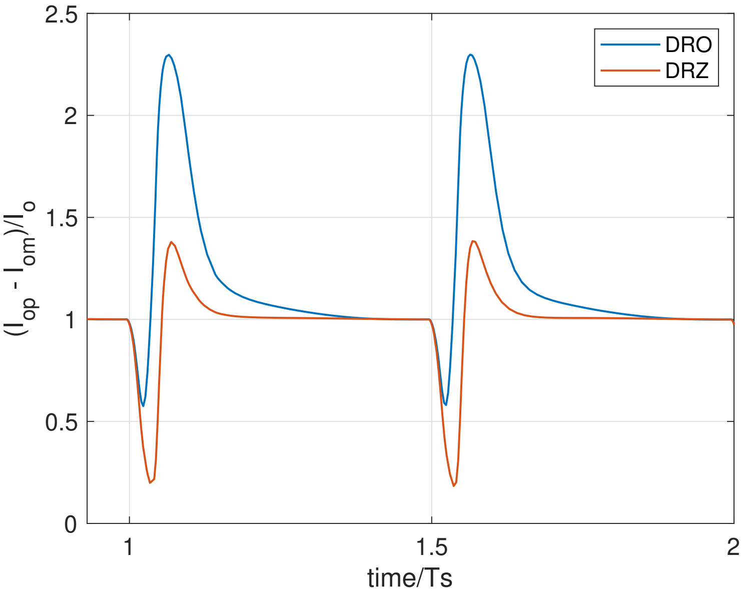


Figure 4.4: Normalized differential current of one DAC unit

4.1.1. Design Procedure

The value of the DAC resistor R_{dac} and the on-resistor of the switches R_{sw} are determined based on the value of the APF resistor and the matching requirements of $\sigma = 0.1\ \%$. Ideally, the static value of the differential current of the DAC, for an optimum residue signal is

$$I_{dac} = \frac{V_{dd}}{R_{APF}} \cdot \frac{M}{M+1} \tag{4.1}$$

and the sum of the DAC and the switch on-resistor is

$$R_{dac} + R_{sw} = (M+1)R_{APF} \tag{4.2}$$

For a 3-bits DAC $M = 7$. Therefore, $I_{dac} = 3.9375\,mA$ and $R_{dac} + R_{sw} = 1600\,\Omega$. However, the glitches shown in Figure 4.4 cause the DAC average current to shift from the nominal value. This shifts the DC gain of the DAC. Therefore, in order to recenter the DAC DC gain to the nominal value, the DAC static current must be adjusted to compensate for the glitches.

The dimensions of the resistor R_{dac} are chosen such that the unit width is $w = 750nm$ to match with the unit width of the R_{APF} and R_{LPF}. It is observed that if R_{APF}, R_{LPF} and R_{dac} have the same unit width, then they show better R-R tracking across PVT variations. Furthermore, 2 unit width are used to improve the resistor matching. This results $\sigma_{Rdac} = 0.09\%$. Figure 4.5 shows the switch number of fingers vs standard deviation σ.

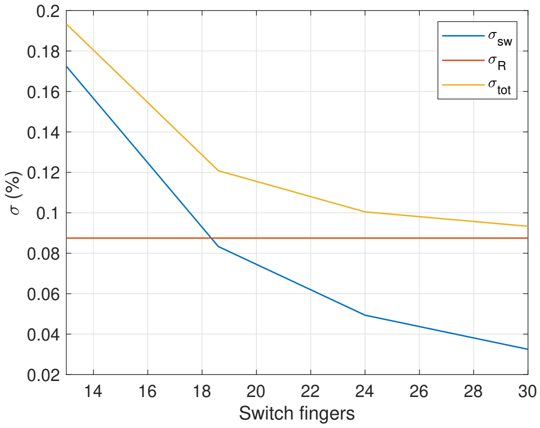


Figure 4.5: Switch number of fingers vs σ

Since the switches are connected to the virtual ground the source voltage is $\approx 0.5V_{dd}$. If the switch drive-waveform have the same supply $V_{dd} = 0.9\ V$, then very large switches are needed to reduce the value of R_{sw} and improve the matching σ_{Rsw}. Based on Figure 4.5, the minimum switch fingers is 24 for unit width of 500nm and drive voltage of $0.9V$. This is about three times larger compared to a switch connected to the supply for the same matching requirement. Having higher supply for the driver complicates the driver design and is challenging at high speed and therefore is not investigated.

The optimal value of the DAC current is found by keeping R_{sw} fixed, and sweeping R_{dac} to compensate the DC shift gain due to the glitches. the optimum value for the DAC resistor is $R_{dac} = 1500\Omega$ for DRZ DAC and $R_{dac} = 1850\Omega$ for DRO DAC to compensate for the average DAC current shift due to the glitches.

Finally, the thermal noise of the DAC output stage is determined by the value of the R_{APF}. For Both DRZ and DRO architecture, the spectral density of the DAC current thermal noise is estimated as follows

$$I_{n,dac} = \frac{8KT}{R_{APF}} \cdot \frac{M}{M+1} \qquad \left(\frac{A^2}{Hz}\right) \tag{4.3}$$

Where K is the Boltzmann constant and T is the temperature. When referred to the input, this translate to a noise voltage spectral density:

$$V_{n,dac} = 8KTR_{APF} \cdot \frac{M+1}{M} \qquad \left(\frac{V^2}{Hz}\right) \tag{4.4}$$

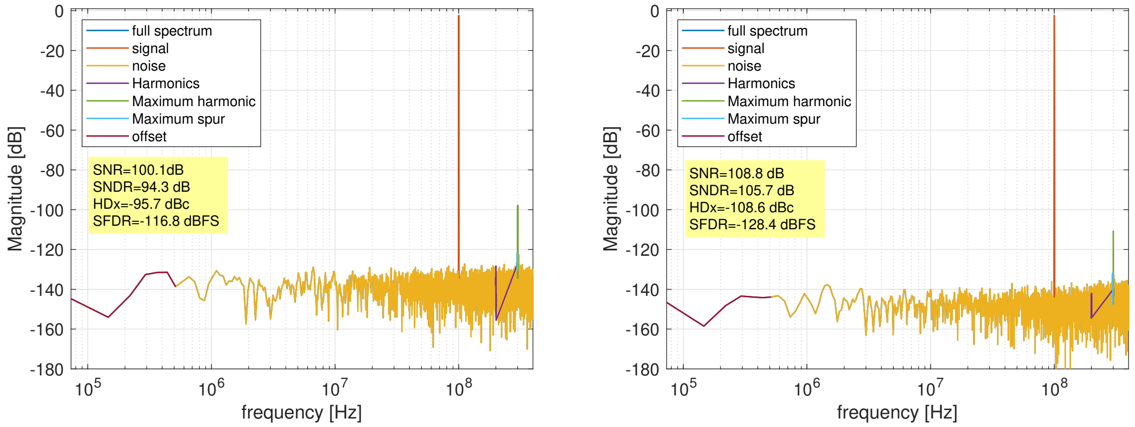
Furthermore, the power consumption of the DRO DAC output stage is given in Equation 4.5.

$$P = \frac{V_{dd}^2}{2R_{APF}} \cdot \frac{M}{M+1} \qquad (W) \tag{4.5}$$

This results $P = 1.77\,mW$. However, in the DRZ DAC, the current continuously flows through the DAC resistors for the entire duration of the clock cycle. This results in doubling of the power consumption of the DAC units compared to the DRO DAC. The power consumption of the DAC units of the DRZ DAC is $3.54\,mW$.
The driver circuitry is adapted from an existing design within NXP. However, it is confidential to NXP and therefore it is not shown in this report.

4.1.2. Simulation Results

The complete schematic of the DAC is tested in the test bench shown in Figure 4.1. Figure 4.6 shows the simulation results of the output spectrum of the CTP for both DAC architectures with the linear amplifier model. The DRZ switches, improves the linearity of the DAC by $\approx 10\,dB$ compared to the DRO DAC. This is because the nodes 1-4 show data-dependent glitches due to the leakage and finite GBW of the amplifier. The RZ switches minimize these data dependent glitches.



(a) With DRO DAC

(b) With DRZ DAC

Figure 4.6: Output spectrum of the first stage with the linear amplifier and DRO/DRZ DAC

Table 4.2: Summary of the results of the first stage

DAC	Amplifier	Results			
		SNDR (dB)	SNR (dB)	SFDR (dBFS)	HDx (dBc)
Ideal	Linear	130.1	130.1	-152.5	-155.2
	Real	88.5	88.5	-108.5	-112.4
Real DRO	Linear	94.3	100.1	-116.8	-95.7
	Real	76.3	80.3	-96	77.5
Real DRZ	Linear	108.8	105.7	-128.4	-108.6
	Real	85	85	-105.3	-104.5

Table 4.2 shows a summary of the results for the first coarse stage according to the test-bench in Figure 4.1. The proposed DAC architecture is tested with the real amplifier. The real DAC with the RZ switches results in $2.5\,dB$ degradation in $SNDR$ compared to the ideal DAC. The DRO DAC results in $\approx 12\,dB$ degradation in $SNDR$ compared with the ideal DAC. While the DRZ DAC meets the specification, the DRO DAC falls short. Nevertheless, it's worth noting that the output stage of the DRZ DAC consumes twice as much power. In section 4.2, a different DRO R-DAC is investigated to reduce the power consumption.

4.1.3. ADEM and ADWA DAC

In order to implement the proposed ADEM and ADWA techniques, the resistors R_{dac} of the coarse stages must have the same value. This means that the coarse stages must be identical. To implement the ADEM and ADWA calibration techniques proposed in chapter 3, both the rows and the columns of the DAC array must be swapped. To illustrate how this is implemented, consider a CTP with two coarse stages. The columns of the DAC array are swapped similar to the conventional DEM/DWA. This is done by shuffling the output of the coarse ADCs in stage 1 and 2 separately. To swap the rows, the DAC elements are combined and connected to both stages through switches that are controlled by the ADEM/ADWA shuffler. A conceptual block diagram is shown in Figure 4.7 and a half slice of the DRZ ADEM/ADWA DAC is shown in Figure 4.8. The other half slice is identical and is obtained by replacing clk with $\overline{clk}$. To implement $ADEM$, the control signal AD must be random. To implement $ADWA$, the control signal AD must be generated according to the noise shaping finite state machines described in chapter 3.

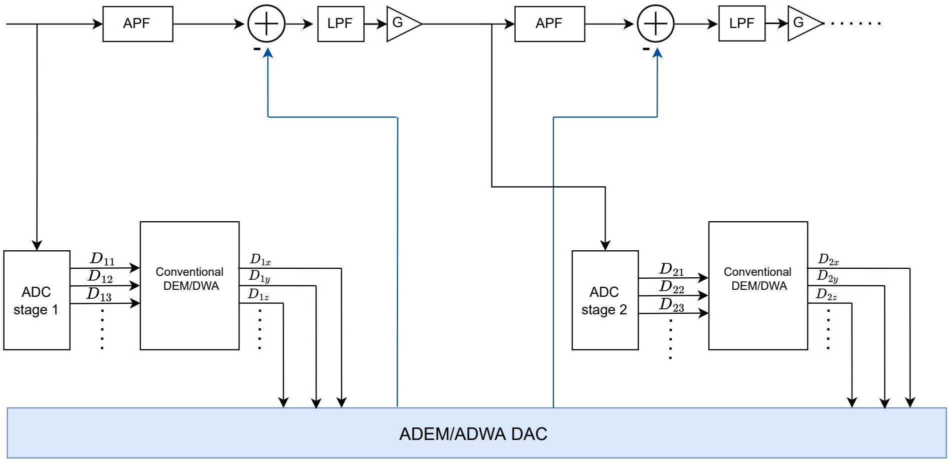


Figure 4.7: Block diagram of CTP ADC with the proposed ADEM/ADWA

The main key of implementing the proposed techniques is to minimize the mismatch of the switches compared to the mismatch of the resistors. However, this results in larger switches. In order to test the full potential of the ADEM technique, the switch size is chosen such that $\sigma_{sw} = 0.01\%$ with respect to $Rdac$. This yields $R_{sw} = 25\,\Omega$. Furthermore, to see the effect of the tonal leakage of the coarse ADCs, the dithering and noise shaping of the coarse ADCs are disabled.

The proposed ADEM is investigated for a 2-stages CTP ADC with the linear amplifier model. The test bench shown in Figure 4.1 is extended to include 2 coarse stages. Furthermore, the implementation of the ADEM with the real amplifier is not investigated. Moreover, The implementation of the ADEM driver is not investigated. However, a Verilog-A model of the driver is used to generate the required DAC signals. Finally, the implementation of the ADWA technique is not investigated.

The performance of the ADEM technique is verified by Monte Carlo simulations. Figures 4.9 and 4.10 shows the simulation results for different scenarios. Firstly, the AD control signal is disabled and an ideal digital filter calibration is applied. Secondly, the AD signal is disabled and the digital calibration filter does not calibrate the Monte Carlo runs. Finally, the AD signal is enabled and the digital calibration filter does not calibrate the Monte Carlo runs. In all scenarios, the conventional DEM is applied. It can be concluded from the results that the proposed ADEM technique improves the SFDR of the 2-stages CTP. If the ADEM is enabled, then all the quantization noise leakage is turned into white noise and therefore, the HDx and SFDR are limited to the noise floor only. Figure 4.11 shows the output spectrum for one of the Monte Carlo runs.

Since the resistors of the coarse stages have the same value, the thermal noise scaling of the coarse stages is not possible. This reduces the power efficiency of the CTP. Furthermore, large switches are needed to implement the ADEM. This increases the power consumption and complicates the design of the DAC driver. Therefore the ADEM and ADWA techniques are not investigated further. Furthermore, the current dither and noise shaping applied to the coarse ADC linearize the coarse ADC sufficiently such that the SFDR requirements of the CTP are met up to 0.28% of digital mismatch. Therefore, it is decided to investigate the implementation of the first stage DAC in more details with the conventional DEM digital calibration of 0.28% gain accuracy.

The virtual ground switched DRO R-DAC of the first stage does not meet the specification. However,

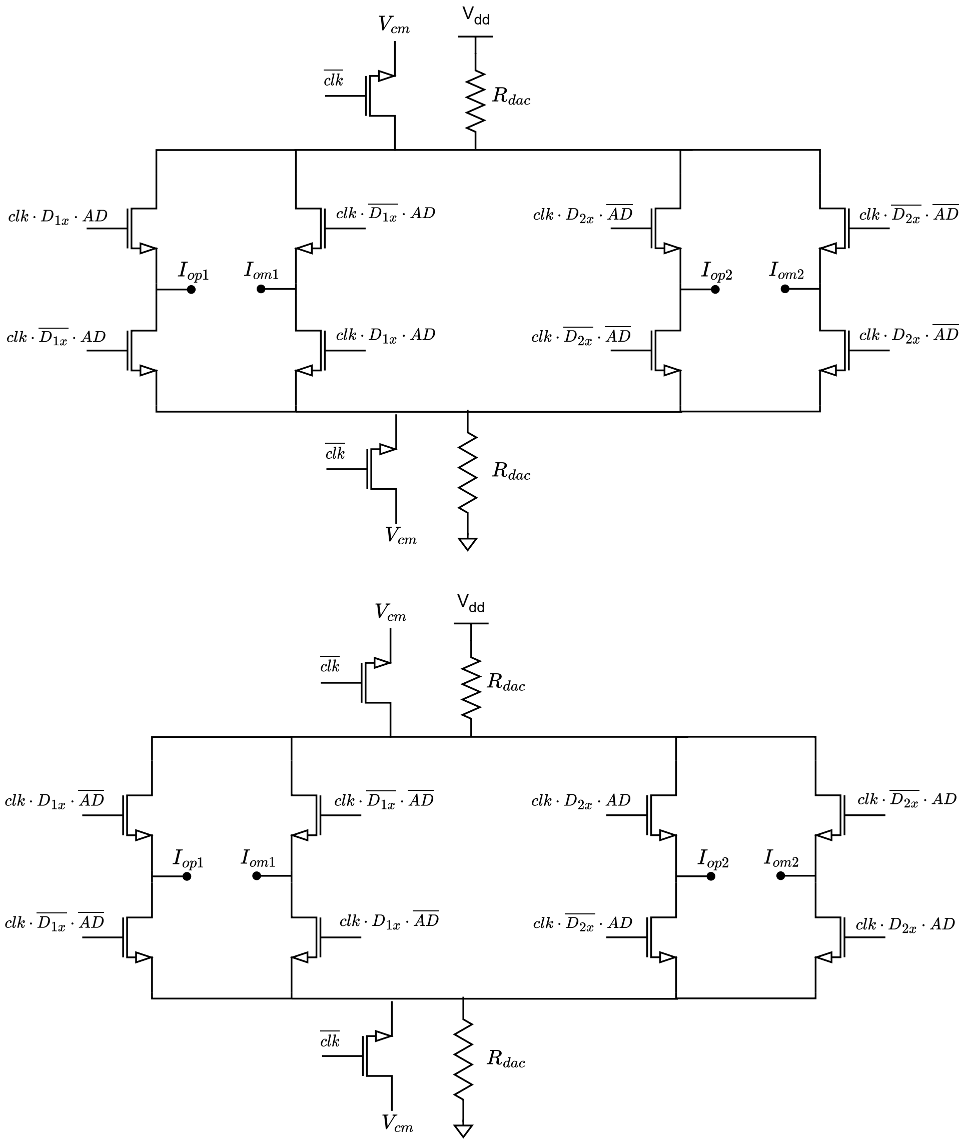
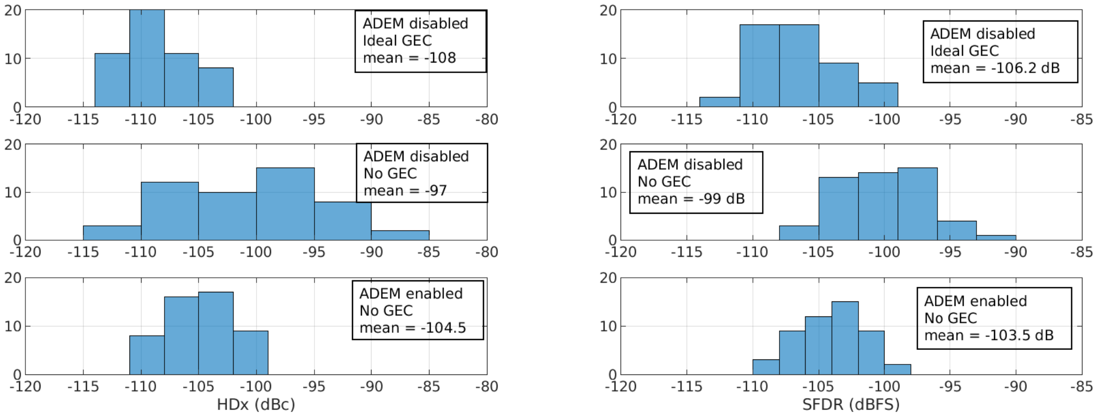

V_{cm}
Vdd
$\overline{clk}$
R_{dac}
$clk \cdot D_{1x} \cdot AD$
$clk \cdot \overline{D_{1x}} \cdot AD$
$clk \cdot D_{2x} \cdot \overline{AD}$
$clk \cdot \overline{D_{2x}} \cdot \overline{AD}$
I_{op1}
I_{om1}
I_{op2}
I_{om2}
$clk \cdot \overline{D_{1x}} \cdot AD$
$clk \cdot D_{1x} \cdot AD$
$clk \cdot \overline{D_{2x}} \cdot \overline{AD}$
$clk \cdot D_{2x} \cdot \overline{AD}$
$\overline{clk}$
R_{dac}
V_{cm}
V_{cm}
Vdd
$\overline{clk}$
R_{dac}
$clk \cdot D_{1x} \cdot \overline{AD}$
$clk \cdot \overline{D_{1x}} \cdot \overline{AD}$
$clk \cdot D_{2x} \cdot AD$
$clk \cdot \overline{D_{2x}} \cdot AD$
I_{op1}
I_{om1}
I_{op2}
I_{om2}
$clk \cdot \overline{D_{1x}} \cdot \overline{AD}$
$clk \cdot D_{1x} \cdot \overline{AD}$
$clk \cdot \overline{D_{2x}} \cdot AD$
$clk \cdot D_{2x} \cdot AD$
$\overline{clk}$
R_{dac}
V_{cm}

Figure 4.8: ADEM/ADWA DRZ half DAC slice for 2 stages

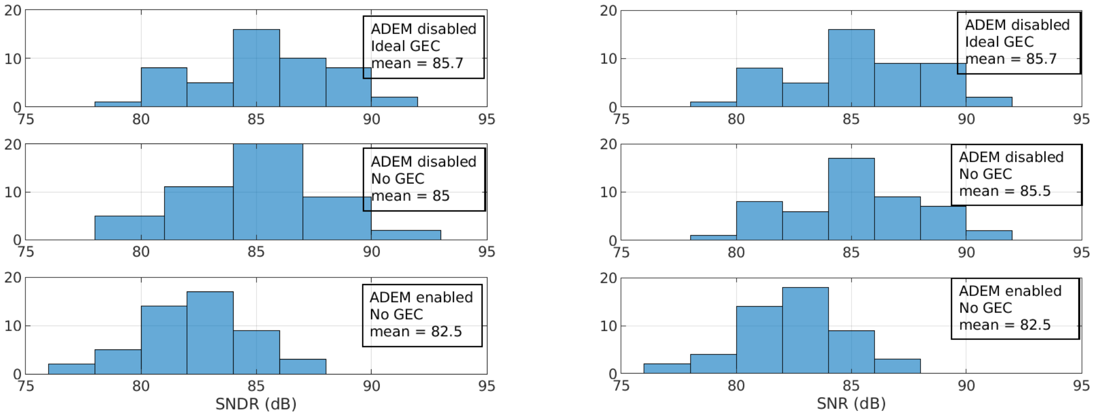
the virtual ground switched DRZ DAC of the first stage meets the specifications but it consumes twice as much power as the DRO DAC. Therefore, in the next section, a different DRO R-DAC architecture is investigated to reduce the power consumption of the DAC output stage.



(a) HDx

(b) SFDR

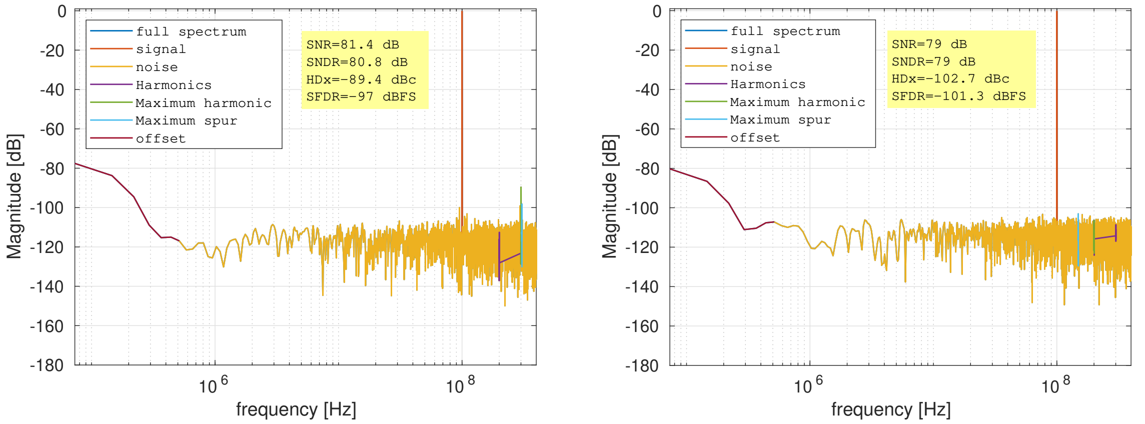
Figure 4.9: 50 runs Monte Carlo simulations for 2-stages CTP. FFT points = 2^{16}.



(a) SNDR

(b) SNR

Figure 4.10: 50 runs Monte Carlo simulations for 2-stages CTP. FFT points = 2^{16}.



(a) ADEM disabled

(b) ADEM enabled

Figure 4.11: Output spectrum of the 2-stages CTP for one of the Monte Carlo runs with and without the ADEM technique

4.2. Supply Switched DRO R-DAC

The purpose of this section is to explore an alternative DRO R-DAC architecture for the initial stage of the CTP. The architecture is shown in Figure 4.12. The DAC driver circuitry is confidential to NXP and therefore it is not shown.

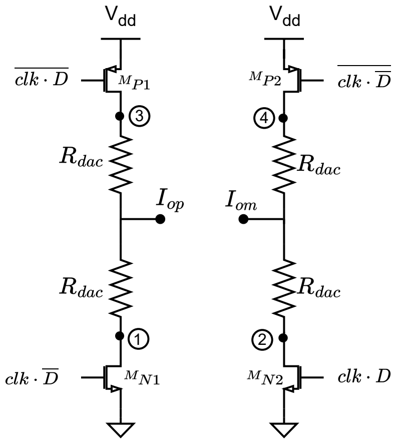
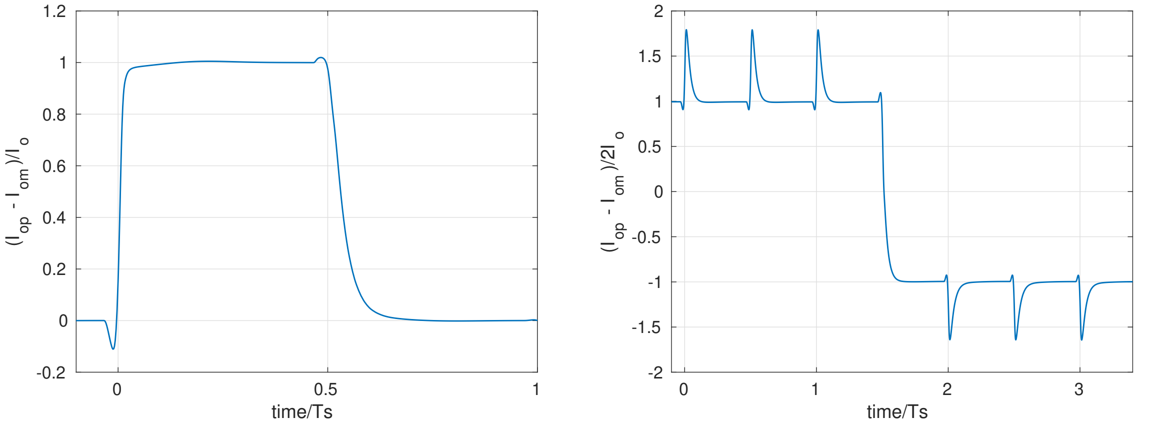


Figure 4.12: DRO R-DAC half slice [28]

Figure 4.13a, shows an example of the RTO current waveform at node 1. The rise and fall times of the current waveform depend on the resistor R_{dac} and the switch resistor R_{sw} and their parasitic capacitors C_p. The rise and fall times can be estimated as follows:

$$t_r = R_{sw}C_p \tag{4.6}$$

$$t_f = R_{dac}C_p \tag{4.7}$$



(a) Normalized RTO current waveform

(b) Normalized differential current of one DAC unit

Figure 4.13: DAC current waveform

The rise and fall time are inherently asymmetrical since the rise time is determined by the switch on-resistance and the fall time is determined by the DAC resistor. However, this asymmetry does not cause any non-linearity but rather data-independent glitches. Figure 4.13b shows an example of the waveform of the DAC unit.

4.2.1. Design Procedure

The design procedure is similar to the procedure described in section 4.1. Since in this architecture, the switches are connected to the supply, smaller switches are required compared to the virtual ground switched DAC. Figure 4.14 shows the switch number of fingers vs standard deviation σ. 8 fingers have been chosen for the switch of a unit width of $500nm$ such that the total standard deviation of the static current is $\sigma = \sqrt{\sigma_{dac}^2 + \sigma_{sw}^2} = 0.1\%$. This results an on-switch resistor of $R_{sw} = 70\Omega$. The optimal value of the DAC current is found by keeping R_{sw} fixed, and sweeping R_{dac} to compensate for the glitches. The optimum value is when $R_{dac} = 1630\,\Omega$. Therefore, the total optimum value is $R_{dac} + R_{sw} = 1700\Omega$.

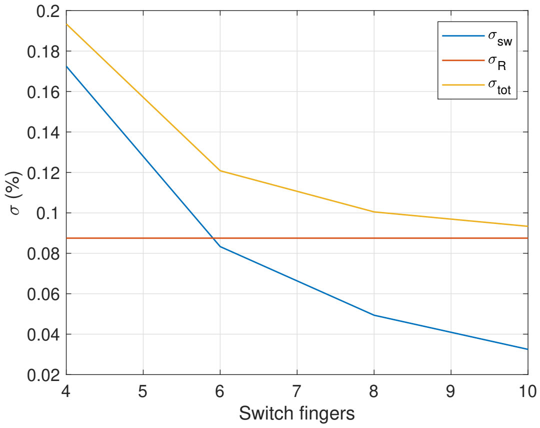


Figure 4.14: Switch number of fingers vs σ

To guarantee the robustness of the design, the DAC output current waveform is investigated within different process corners and varying temperatures. Figure 4.15 shows the output current of one unit element of the DAC for different corners. The output current is normalized to the typical current output at $65^o\ C$. The process and temperature variations affect the delay, the static value and the glitches of the DAC current.

The delay variations are mainly due to the clock path in the driver circuit. The worst values shifts the delay by $0.012T_s$ at SS_{150^oc} and $-0.01T_s$ at FF_{-40^oc} from the nominal delay $1.5T_s$. As a result, the residue signal increases. Figure 4.16 shows the maximum output of the real amplifier versus additional time delay in the DAC output. The residue signal increases marginally from $0.72\,V$ to $\approx 0.73\,V$ at the worst delay shift due to the DAC. Figure 4.17 shows the $SNDR$ and $SFDR$. The $SNDR$ and $SFDR$ degrade as the time delay varies. This is due to the non-linearity of the amplifier. However, the degradation in $SNDR$ and $SFDR$ due to the worst case delay variations in the DAC output, is negligible.

The static value of the DAC output current varies according to process and temperature variations. This is due to variation in R_{dac} and R_{sw}. Ideally, the DC gain is constant since the APF resistor R_{APF} and the LPF resistor R_{LPF} are subject to the same global and temperature variations and therefore the resistors values track each other. However, it is observed that this tracking is not exact. This is because the resistors have different dimensions. This tracking mismatch between the APF, LPF and DAC will shift the DAC gain. Figure 4.18 shows the effect of the DC gain variation on the performance. The DC gain is the total gain from the DAC to the output of the real amplifier. This simulations was done by sweeping the DAC resistor. The $SNDR$ and $SFDR$ drop as the gain varies from the nominal value of $12\ dB$. This drop is caused by the non-linearity of the amplifier. Based on the results, To meet the $SFDR$ requirement, the maximum tolerable DC gain shift is $0.5\ dB$ or 6%. To allow better R-R tracking and to minimize the DC gain shift, the R_{dac} unit width is unified with the APF and LPF resistors. Based on the values of DAC, APF, LPF resistors over process corners, the worst case error in the total DC gain is $< 3\%$. The worst case shift meets the specifications and therefore it is acceptable.

As shown in Figure 4.15 the glitches of the DAC current varies according to process and temperature variations. This is due to variation the rise and fall times of the DAC unit element. These variations in

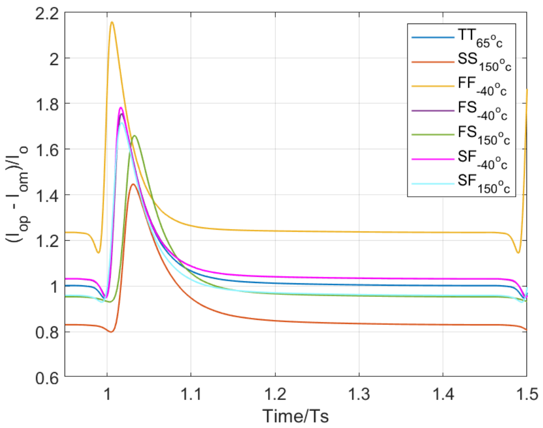


Figure 4.15: Output current of one unit element of the DAC at different process corners and temperatures.

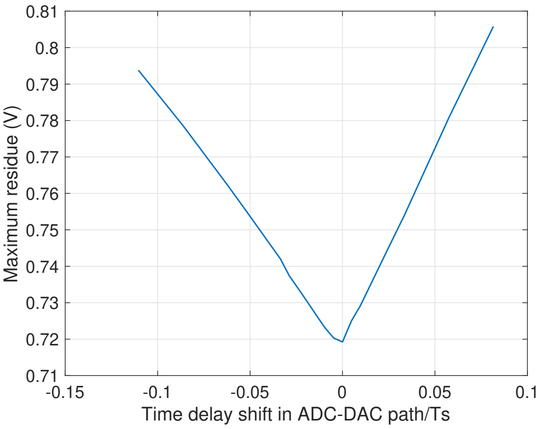
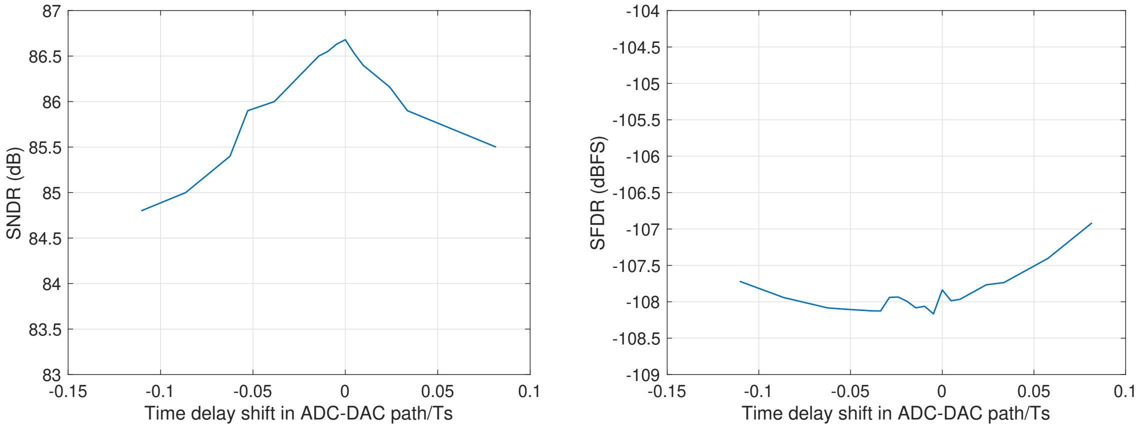


Figure 4.16: Maximum output of the amplifier vs additional time delay shift in DAC output. Delay shift of 0 refers to the nominal delay $1.5Ts$

the glitches will cause the average the DAC current decrease or increase. As a results the DC gain will also shift. Based on the simulation, the worst case DC gain shift is $< 1.4\%$ at $SS_{150^o c}$. This value is however small and acceptable.

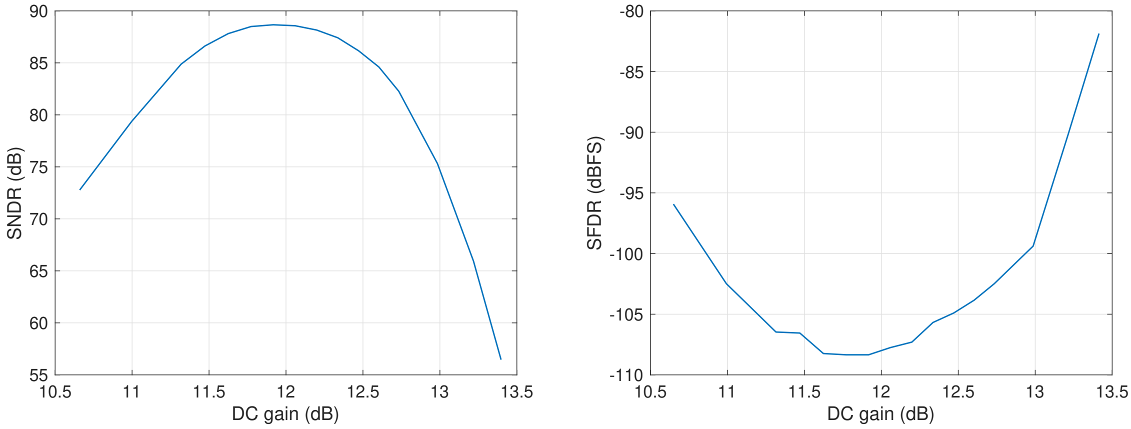
Finally, the thermal noise spectral density of the DAC current is similar to the VGS R-DAC and is given in Equation 4.3. Similarly, the power consumption of the DAC output stage is given in Equation 4.5 and it is equal to $1.77\ mW$.



(a) $SNDR$ at $F_{in} = 380\ MHz$

(b) $SFDR$ at $F_{in} = 380\ MHz$

Figure 4.17: $SNDR$ and $SFDR$ vs additional time delay shift in DAC output. Delay shift of 0 refers to the nominal delay $1.5Ts$



(a) $SNDR$

(b) $SFDR$

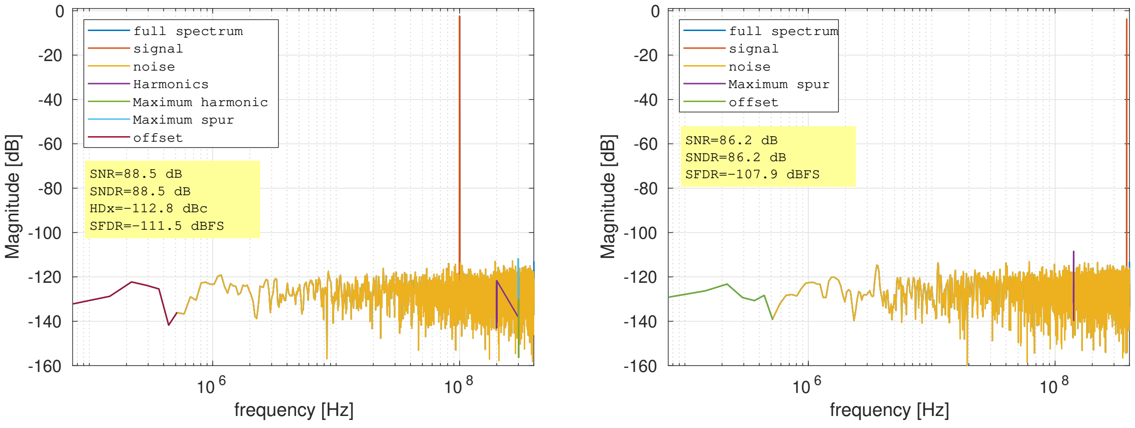
Figure 4.18: Total DC gain from the DAC input to the real amplifier output versus $SNDR$ and $SFDR$

4.2.2. Schematic simulation results

The complete DAC schematic circuit is simulated according to the test bench shown in 4.1. Table 4.3 shows the schematic simulation results. The real DAC refers to the complete DAC schematic including the DAC output stage, driver and the DEM circuitry. The performance of the DAC is similar to the ideal DAC and the linearity is limited by the real amplifier. Figure 4.19 shows the output spectrum of the first stage with the complete DAC schematic and the real amplifier.

Table 4.3: Schematic simulation results

DAC	Amplifier	Results Fin = 100 MHz /380 MHz			
		SNDR (dB)	SNR (dB)	SFDR (dBFS)	HDx (dBc)
Ideal	Linear	130.1/129.3	130.1/129.3	-152.5/-153.4	-155.2/x
Real	Linear	129/128.5	129/128.5	-149.3/-154.1	-146.2/x
Ideal	Real	88.5/86.2	88.5/86.2	-111.2/-108.5	-112.4/x
Real	Real	88.5/86.2	88.5/86.2	-111.5/-107.9	-112.8/x

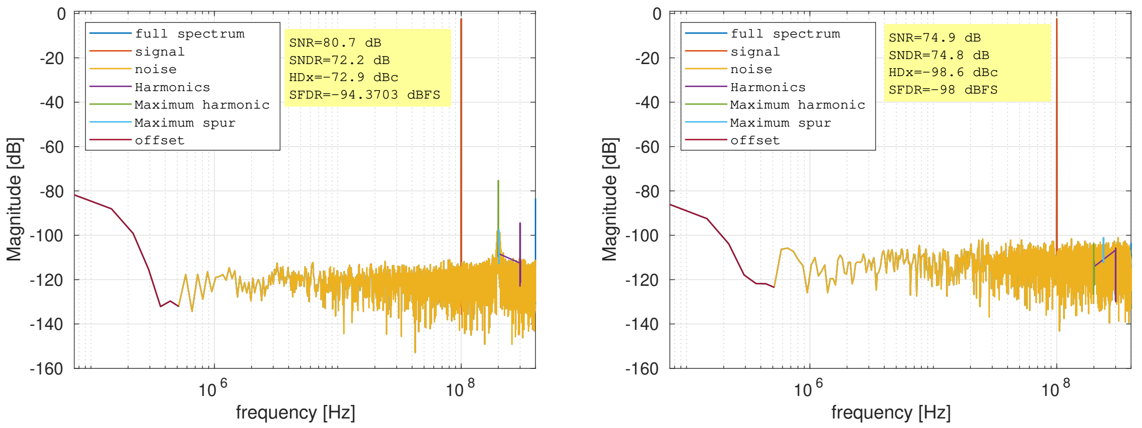


(a) $F_{in} = 100\ MHz$

(b) $F_{in} = 380\ MHz$

Figure 4.19: Output spectrum of the first stage with the real DAC and amplifier schematics. $V_{in} = -2.1\ dBFs$

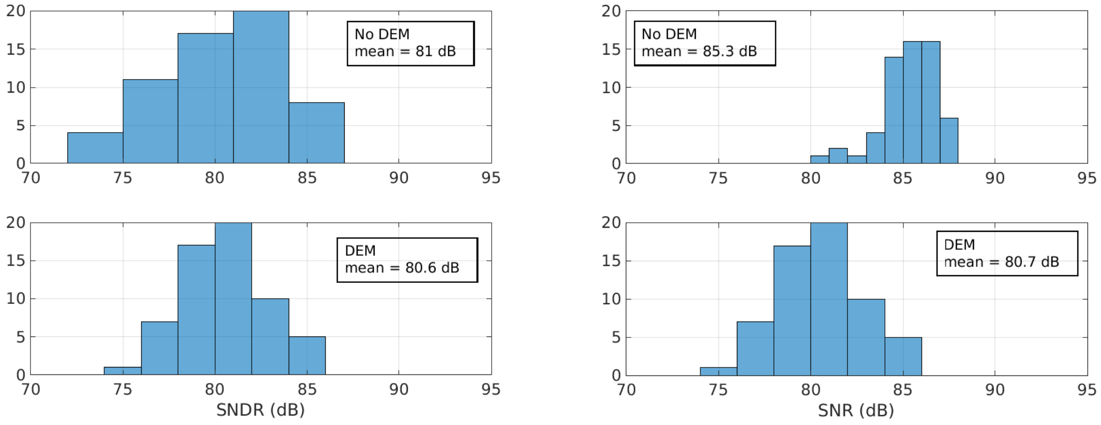
Figure 4.20 shows the output spectrum of the first stage with the real DAC and amplifier with random local mismatch and conventional DEM applied to the DAC with ideal digital calibration. As shown, the DAC distortion is converted into white noise which degrades the dynamic range. Furthermore, Figures 4.21 and 4.22 show the Monte Carlo simulations with and without applying the conventional DEM. After applying the DEM technique, the DAC is linearized and therefore $SFDR$ and HDx are limited to the noise floor only. Increasing the simulation time or applying FFT coherent averaging lowers the noise floor and allows for higher spectral accuracy. Finally, the average $SNDR = 80.6\ dB$ and the worst case $SNDR$ for the first stage is $\approx 75\ dB$.



(a) DEM disabled

(b) DEM enabled

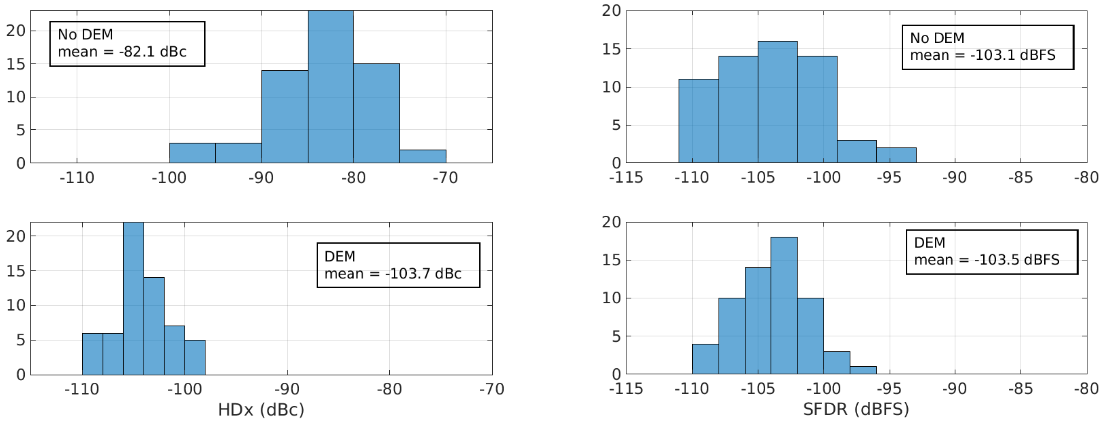
Figure 4.20: Output spectrum of the first stage with the real DAC and amplifier with local random mismatch applied to the DAC. $F_{in} = 100\ MHz$ and $V_{in} = -2.1\ dBFs$. FFT points $= 2^{16}$



(a) SNDR

(b) SNR

Figure 4.21: 60 runs Monte Carlo simulations. $F_{in} = 100\ MHz$ and $V_{in} = -2.1\ dBFs$. FFT points $= 2^{16}$.



(a) HDx

(b) SFDR

Figure 4.22: 60 runs Monte Carlo simulations. $F_{in} = 100\ MHz$ and $V_{in} = -2.1\ dBFs$. FFT points $= 2^{16}$.

4.2.3. Layout simulation results

The layout of the DAC output stage is very critical to achieve high linearity performance. Parasitic resistors from the supply cause different voltage drop for each DAC unit. This will cause harmonic distortion. Therefore, these parasitic resistors should be minimized and have balanced distribution across the DAC units to minimize the INL errors. Similarly, the routing of the output current of the DAC elements should have a balanced tree structure to minimize the INL distortion. Figure 4.23 shows the a balanced binary tree used for the supply and the DAC output signals routing. The goal of the tree is to minimize the delay difference between the root and the leafs and to equalize the voltage drop from each leaf node to the root [30].

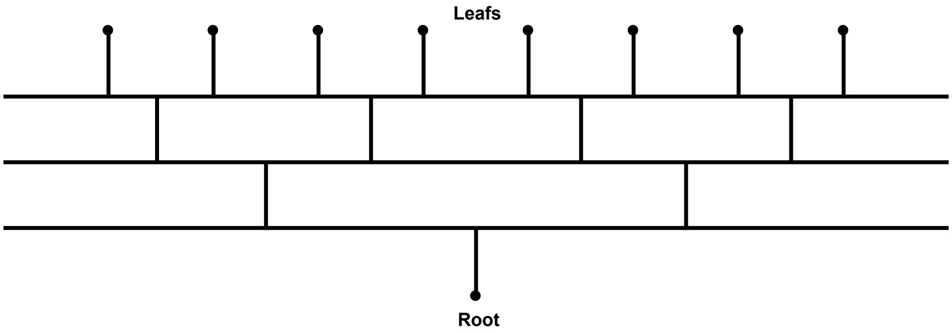
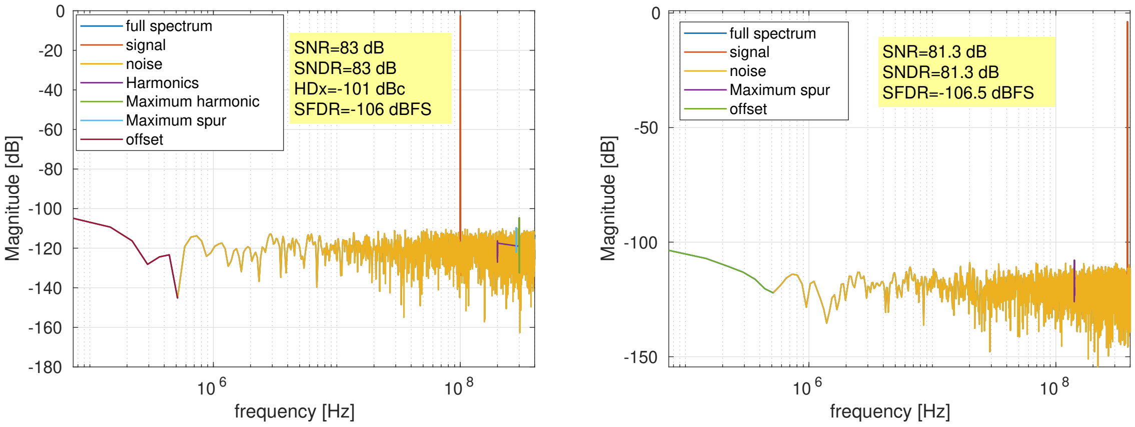


Figure 4.23: Binary tree routing for the supply and the output signals

Figure 4.24 shows the output spectrum with a full post-layout parasitic extraction of the DAC output stage. The results show $5.5\ dB$ SNDR degradation compared to the schematic results. Even with the balanced routing tree, the voltage drop from each unit to the output is not equalized completely. Further tuning of the tree might be done to improve the performance. For example widening some of the tree levels or shifting the connections between the levels to compensate the remaining INL errors. Although the layout could be further optimized, it has not been done since the specifications have been met.



(a) Fin = 100MHz

(b) Fin = 380MHz

Figure 4.24: Post layout simulation results of the DAC output stage

4.3. Summary

In this chapter, the implementation of the coarse DAC in the CTP ADC is investigated. Firstly, the design of a virtual ground switched DRZ and DRO R-DAC for the first stage of the CTP are presented. The schematic simulation results of the first stage of the CTP shows that the DRO R-DAC fails to meet the $SFDR$ requirement. The DRZ DAC meets the specification. However, the output stage of the DRZ DAC consumes twice the power compared with the DRO DAC. Thereafter, the virtual ground switched DRZ R-DAC is further developed to allow the implementation of the ADEM and ADWA calibration techniques. The ADEM calibration technique is modeled in Verilog-A and the ADEM DAC is tested with a linear amplifier model and ideal driver. Monte Carlo simulation results show that after applying the ADEM technique, the SFDR is limited to the noise floor only as all the DAC mismatch errors are turned into white noise. However, Implementing the ADEM/ADWA technique requires the coarse stages to be identical. This prevents thermal noise scaling of the coarse stages and therefore reduces the power efficiency of the CTP. Furthermore, the effective linearity of the coarse ADC is improved from 3-bits to an equivalent linearity of an 8 bit ADC. This relaxes the quantization noise leakage requirements to meet the SFDR specification of the CTP.

Finally, a different DRO R-DAC architecture for the first stage of the CTP is investigated. The architecture is verified within the first stage of the CTP with ideal digital calibration and conventional DEM technique. Schematic simulation results of the DAC achieves $SNDR = 88.5\,dB$ and $SFDR < -104\,dB$. Furthermore, Monte Carlo simulations of the DAC shows an average $SNDR$ of $80\,dB$ and a worst case $SNDR$ of $75\,dB$. However, post-layout simulation of the DAC output stage shows $5.5\,dB$ degradation in $SNDR$ compared to the schematic results. This degradation is due to INL errors introduced in the supply and output signals routing.

5 Conclusion

The goal of the thesis is to investigate the implementation and the design of the coarse DACs in continuous time pipeline ADC. The CTP consists of three coarse stages and a back-end stage. Each coarse stage has a resolution of 3 bits. The back-end stage has a resolution of 6 bits. The coarse stages are sampled at $4.8\ GHz$ and the back-end stage at $1.6\ GHz$. The targeted bandwidth is $400\ MHz$ and $SFDR < -104\ dBFS$ with $SNDR$ of $75\ dB$.
Firstly, based on MATLAB modeling and simulations it has been concluded that DRZ/DRO DAC architectures are desirable for the first two stages of the CTP to mitigate the ISI distortion. Furthermore, mismatch errors arising form the coarse DACs are very critical to achieve high linearity performance. Mismatch errors produce harmonic distortion and leak the quantization noise of the coarse stages to the output. Dynamic element matching technique can be used to linearize the DAC and to convert the DAC distortion into white noise at the cost of degradation in the dynamic range. Moreover, data weighted averaging can be used to improve the dynamic range by noise shaping the DAC distortion. However, after the linearization, the DAC will still exhibit a gain error. The gain errors leak the quantization noise of the coarse stages to the output. The gain errors of the coarse DACs of the initial stages of the CTP must be corrected in order to meet the CTP specification. The DAC gain errors can be corrected by the digital compensation filter if their values are estimated. However, it is observed that the relative gain errors between the DACs in the coarse stages may amplifies or minimize the quantization leakage. Based on this, two calibration techniques are proposed. The first calibration technique, ADEM, converts the distortion and gain errors of the DACs due to mismatch into white noise. The second calibration technique, ADWA, aims to noise shape the distortion and gain errors of the coarse DACs to improve the dynamic range. Furthermore, a DAC architecture is presented to implement the ADEM and ADWA calibration techniques. The ADEM DAC is verified in a 2 coarse stages CTP with a linear amplifier model. Cadence simulation results show that after applying the ADEM technique the distortion and gain errors of the DACs are translated into white noise and therefore the $SFDR$ specifications are met without the need to calibrate the gain errors of the DAC by the digital compensation filters. However, in the case of the conventional DEM technique, calibration of the DAC gain errors are required to meet the $SFDR$ specification of the CTP. It is worth to mention that the artifacts of the ADEM DAC architecture with a real driver circuitry and a real amplifier are not investigated. Finally, The ADEM/ADWA DAC architecture requires the coarse stages of the CTP to be identical. This might reduce the power efficiency of the CTP. However, the ADEM/ADWA do not require any additional digital calibration. Therefore, the design of the digital compensation filter might be relaxed since the errors arise from the DACs do not need to be estimated and corrected. This might reduce the total power consumption of digital filters and hence improve the power efficiency of the CTP. Furthermore, the implementation of the digital filter might be simplified if the DACs do not require estimation.

Finally, a 3-bit DRO R-DAC design is presented. The design employs conventional DEM technique and is verified within the first stage of the CTP with ideal digital calibration. Simulation results shows an $SFDR < -104\ dBFS$ and a worst case $SNDR$ of $75\ dB$ for 60 Monte Carlo runs. It is recommended for future work to investigate the implementation of the DWA to improve the $SNDR$. Furthermore, the layout of the output stage might be improved by tuning the level connections of the routing tree to further minimize the INL errors introduced by the routing of the supply and the output stage.

5.1. Thesis Contribution

My contributions to this thesis are summarized in the following list

- MATLAB modeling of the coarse DACs in CTP ADC.
- Investigating the non-idealities of the DACs in all CTP stages and derive requirements for the DACs based on MATLAB simulations.
- Investigating the calibration techniques of the DACs and proposing two new calibration techniques.
- Proposing a DAC schematic that allows the implementation of the proposed calibration techniques.
- Implementing a DRO R-DAC in the first stage of the CTP.

References

[1] M. Abo Alainein and O. El Boustani, “Bi-static sense and avoid system for drones: Signal design,” Jan 2021. http://resolver.tudelft.nl/uuid:af9a9e28-9b6d-4d98-8ac9-e2ffee50be91.

[2] M. I. Skolnik, *Radar Handbook*. McGraw-Hill Education Ltd, 2008.

[3] M. Jankiraman, *FMCW radar design*. Artech House, 2018.

[4] K. Doris, A. Filippi, and F. Jansen, “Reframing fast-chirp fmcw transceivers for future automotive radar: The pathway to higher resolution,” *IEEE Solid-State Circuits Magazine*, vol. 14, no. 2, pp. 44–55, 2022.

[5] H. Shibata, V. Kozlov, Z. Ji, A. Ganesan, H. Zhu, D. Paterson, J. Zhao, S. Patil, and S. Pavan, “A 9-gs/s 1.125-ghz bw oversampling continuous-time pipeline adc achieving −164-dbfs/hz nsd,” *IEEE Journal of Solid-State Circuits*, vol. 52, no. 12, pp. 3219–3234, 2017.

[6] S. Manivannan and S. Pavan, “A 65-nm cmos continuous-time pipeline adc achieving 70-db sndr in 100-mhz bandwidth,” *IEEE Solid-State Circuits Letters*, vol. 4, pp. 92–95, 2021.

[7] P. Cenci, M. Bolatkale, R. Rutten, M. Ganzerli, G. Lassche, K. Makinwa, and L. Breems, “A 3.2mw sar-assisted ctΔ∑ adc with 77.5db sndr and 40mhz bw in 28nm cmos,” in *2019 Symposium on VLSI Circuits*, pp. C230–C231, 2019.

[8] S. Pavan and H. Shibata, “Continuous-time pipelined analog-to-digital converters: A mini-tutorial,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 3, pp. 810–815, 2021.

[9] S. Billa, S. Dixit, and S. Pavan, “Analysis and design of an audio continuous-time 1-x fir-mash delta–sigma modulator,” *IEEE Journal of Solid-State Circuits*, vol. 55, no. 10, pp. 2649–2659, 2020.

[10] N. Basavaraj, S. Manivannan, and S. Pavan, “Simplified simulation and measurement of the signal transfer function of a continuous-time pipelined analog-to-digital converter,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 10, pp. 3993–3997, 2022.

[11] A. Abdelaal, J. G. Kauffman, M. A. Mokhtar, and M. Ortmanns, “A comparative study of isi errors in different dac structures for ct delta-sigma modulators,” in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2020.

[12] L. Risbo, R. Hezar, B. Kelleci, H. Kiper, and M. Fares, “Digital approaches to isi-mitigation in high-resolution oversampled multi-level d/a converters,” *IEEE Journal of Solid-State Circuits*, vol. 46, no. 12, pp. 2892–2903, 2011.

[13] W.-T. Lin, H.-Y. Huang, and T.-H. Kuo, “A 12-bit 40 nm dac achieving sfdr > 70 db at 1.6 gs/s and imd < –61db at 2.8 gs/s with demdrz technique,” *IEEE Journal of Solid-State Circuits*, vol. 49, no. 3, pp. 708–717, 2014.

[14] W.-H. Tseng, C.-W. Fan, and J.-T. Wu, “A 12-bit 1.25-gs/s dac in 90 nm cmos with >70 db sfdr up to 500 mhz,” *IEEE Journal of Solid-State Circuits*, vol. 46, no. 12, pp. 2845–2856, 2011.

[15] X. Su, Z. Yu, Y. Jiang, Z. Jiao, L. Zhao, Z. Chen, and H. Zhang, “A 14-bit 2.8gs/s dac with dtirz technique in 65 nm cmos,” *IEICE Electronics Express*, vol. 18, no. 6, pp. 20210043–20210043, 2021.

[16] R. Adams and K. Nguyen, “A 113-db snr oversampling dac with segmented noise-shaped scrambling,” *IEEE Journal of Solid-State Circuits*, vol. 33, no. 12, pp. 1871–1878, 1998.

[17] I. Galton, “Why dynamic-element-matching dacs work,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 2, pp. 69–74, 2010.

[18] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*. Wiley-IEEE Press, 2005.

[19] I. Galton and N. Rakuljic, “Tree-structured dem dacs with arbitrary numbers of levels,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 2, pp. 313–322, 2010.

[20] M. Neitola and T. Rahkonen, “A generalized data-weighted averaging algorithm,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 2, pp. 115–119, 2010.

[21] D.-H. Lee and T.-H. Kuo, “Random incrementing data weighted averaging for nyquist-rate digital-to-analog converters,” in *2007 International Symposium on Integrated Circuits*, pp. 552–555, 2007.

[22] M. Vadipour, “Techniques for preventing tonal behavior of data weighted averaging algorithm in /spl sigma/-/spl delta/ modulators,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 47, no. 11, pp. 1137–1144, 2000.

[23] A. Celin and A. Gerosa, “A reduced hardware complexity data-weighted averaging algorithm with no tonal behavior,” in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 702–705, 2016.

[24] J. Welz, I. Galton, and E. Fogleman, “Simplified logic for first-order and second-order mismatch-shaping digital-to-analog converters,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 11, pp. 1014–1027, 2001.

[25] N. Rakuljic and I. Galton, “Tree-structured dem dacs with arbitrary numbers of levels,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 2, pp. 313–322, 2010.

[26] S. Pavan, R. Schreier, and G. C. Temes, *Understanding Delta Sigma Data Converters*. Wiley-IEEE Press, 2nd ed., 2017.

[27] X. Li and L. Zhou, “A survey of high-speed high-resolution current steering dacs,” *Journal of Semiconductors*, vol. 41, p. 111404, nov 2020.

[28] M. Bolatkale, R. Rutten, H. Brekelmans, S. Bajoria, Y. Gao, B. Burdiek, and L. Breems, “A 28nm 6ghz 2b continuous-time $\delta\sigma$ adc with −101 dbc thd and 120mhz bandwidth using digital dac error correction,” in *2022 IEEE International Solid- State Circuits Conference (ISSCC)*, vol. 65, pp. 416–418, 2022.

[29] R. Theertham, P. Koottala, S. Billa, and S. Pavan, “Design techniques for high-resolution continuous-time delta–sigma converters with low in-band noise spectral density,” *IEEE Journal of Solid-State Circuits*, vol. 55, no. 9, pp. 2429–2442, 2020.

[30] E. Bechthum, *Wide-band mixing DACs with high spectral purity*. PhD thesis, Electrical Engineering, Mar. 2015. Proefschrift.