POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell'informazione
Corso di Laurea Magistrale in Ingegneria Elettronica

POLITECNICO
MILANO 1863

Sigma-Delta Analogue-to-Digital converter for column-parallel CMOS image sensors

Relatore: Prof. Marco Sampietro

Correlatore: Ing. Iain Sedgwick

Tesi di laurea di:

Michele SANNINO Matr. 817325

Anno accademico 2015/2016

Acknowledgements

I would like to thank Iain Sedgwick whose precious, patient and close support in both technical and formal matters has greatly helped me in the accomplishment of this work. I would also like to thank Renato Turchetta and the whole CMOS Sensor Design Group of the Rutherford Appleton Laboratory (in Harwell, Oxfordshire), for the passionate contribution they offered.

Special thanks also to Professors Marco Sampietro and Giorgio Ferrari, who devoted some of their own time to help me move forward with my work.

Summary

List of Figures

List of Tables

Introduzione

Il mondo dell'imaging digitale ha visto una crescita stabile negli ultimi anni, con i sensori d'immagine CMOS (CMOS Image Sensors, CIS) alla guida dell'espansione grazie alla loro versatilità e capacità di prestazioni ad alta velocità senza compromettere la qualità dell'immagine. I CIS stanno così diventando la scelta designata per un crescente numero di applicazioni, di tipo industriale - da fotocamere per telefoni cellulari, al campo automobilistico (ad esempio nei cosiddetti Advanced Driver Assistance Systems, "ADAS") e di tipo scientifico (ad esempio in ingegneria aerospaziale, balistica e microscopia). Una costante per tutte queste applicazioni è la richiesta di una sempre maggiore velocità, il che costituisce una stimolante sfida, sia a livello tecnologico che progettuale.

Uno dei componenti chiave nella catena di lettura in un CIS è il convertitore analogico-digitale (Analogue-to-Digital Converter, ADC), la cui rapidità contribuisce a determinare il numero di fotogrammi per secondo (frame-rate) ottenibili dal sensore. Il chip di un CIS può ospitare diversi ADC per permettere una parallelizzazione della lettura (ossia la conversione dell'informazione di svariati gruppi di pixel in simultanea): la soluzione più comunemente adottata è la topologia a "colonne parallele" (column-parallel), con un ADC assegnato a ciascuna colonna della matrice di pixel del sensore.

In aggiunta ai requisiti sull'alta velocità di conversione, le principali specifiche di un ADC per sensori d'immagine riguardano basso rumore, dimensioni ridotte e un contenuto mismatch tra le prestazioni dei diversi ADC. Rispettare questi vincoli è una sfida, poiché migliorare uno di questi aspetti (ad esempio velocità di conversione e contenimento del mismatch) porta spesso al peggioramento di un altro (ad esempio rumore e dimensioni).

Gli ADC Sigma-Delta ($\Sigma\Delta$) sono una promettente soluzione per il superamento di questi limiti. Nonostante la loro applicazione nell'ambito dei sensori d'immagine sia stata suggerita sin dal 1997 [1], questi convertitori hanno iniziato ad attirare l'interesse della comunità dell'imaging solo in tempi recenti, dopo essere stati utilizzati tradizionalmente in altre applicazioni - quali ad esempio quelle del mondo audio.

I $\Sigma\Delta$ sfruttano una ben nota proprietà degli anelli di retroazione, ovverosia il fatto che non idealità introdotte nel cammino di andata dell'anello – come mismatch e distorsione – hanno un impatto ridotto sul trasferimento complessivo, poiché il segnale è per la maggior parte trasferito nel cammino di retroazione. I $\Sigma\Delta$ ADC applicano questo concetto in un sistema misto analogico-digitale, dove il percorso d'andata è completamente analogico mentre l'uscita ed il percorso di retroazione sono quantizzati: la loro peculiare architettura e l'uso di

oversampling (il campionamento dell'ingresso ad una frequenza maggiore della minima necessaria) permettono di operare conversioni in grado di dare la risoluzione desiderata, ottenendo allo stesso tempo basso rumore e buona tolleranza alle variazioni di processo senza rinunciare alle ridotte dimensioni.

Il presente lavoro di tesi descrive il progetto di un ADC $\Sigma\Delta$ per sensori column-parallel, in grado di dare 12 bits di risoluzione nel competitivo tempo di conversione di $1\mu s$. Il sistema è stato realizzato nella tecnologia TowerJazz $0.18\mu m$ per CIS. È stato progettato all'interno del CMOS Sensor Design Group al Rutherford Appleton Laboratory, in Harwell (Oxfordshire), dove vengono realizzati CIS e sensori di radiazione a basso rumore allo stato dell'arte per applicazioni in ambito scientifico.

Questo lavoro è suddiviso in otto capitoli, come di seguito riepilogato.

Nel Capitolo 1 sono descritti i fondamenti dell'operazione di un sensore d'immagine CMOS. L'esposizione degli argomenti in questo capitolo è indirizzata ad aspetti che verranno richiamati nelle successive discussioni riguardanti il progetto dell'ADC.

Il Capitolo 2 tratta gli ADC in generale ed in relazione al frame-rate del sensore. A partire da una revisione delle principali figure di merito degli ADC, si passa ad una esposizione delle principali topologie di readout in un CIS, spiegando in che modo migliorino le prestazioni del sensore. Successivamente, vengono analizzati i trade-off più rilevanti negli ADC per CIS, evidenziando le ragioni per cui i $\Sigma\Delta$ promettano di essere un miglioramento rispetto ad altre soluzioni. Il capitolo termina con l'esposizione generale delle specifiche di progetto.

Il Capitolo 3 analizza in dettaglio l'architettura classica di un Sigma-Delta del primo ordine, fornendo le conoscenze necessarie per comprendere le sue principali caratteristiche (ossia oversampling e noise-shaping), il filtraggio del rumore in ingresso ed alcuni problemi di linearità legati a componenti non ideali. Il capitolo introduce peraltro una particolare tipologia di convertitore $\Sigma\Delta$, detta Incremental Sigma-Delta (ISD), comunemente utilizzate nei sensori d'immagine CMOS ed è la scelta adottata in questo progetto.

Il Capitolo 4 continua l'analisi dei $\Sigma\Delta$, allargando l'orizzonte ad architetture più complesse. Dopo aver descritto diverse possibili soluzioni, vien presentata la scelta finale sull'architettura del convertitore implementato in questo progetto, e vengono elencate le specifiche per i componenti analogici – opportunamente derivate da simulazioni globali del sistema. In particolare, l'architettura scelta per l'ADC è quella di un Incremental Sigma-Delta del 2° ordine, che compie 100 cicli per conversione e lavora dunque ad una frequenza di $100MHz$.

Il Capitolo 5 è dedicato ai circuiti a capacità commutate: l'obiettivo di questo capitolo è spiegare il principio di funzionamento di tali circuiti e giustificare le formule utilizzate nel capitolo successivo, riguardante la progettazione analogica. Alla luce di quanto qui esposto il lavoro potrà quindi proseguire focalizzando la discussione sul *design flow*.

Il Capitolo 6 tratta la progettazione analogica del sistema, ossia l'implementazione del modulatore del ΣΔ. Questo blocco è composto da due integratori a capacità commutate ed un comparatore, che costituiscono il cuore dell'operazione del ΣΔ.

Il Capitolo 7 inizia esaminando il design digitale del decimatore dell'ADC e la sintesi dei clock del sistema, spiegando in particolare le misure prese per garantire la sincronizzazione di segnali con diverse alimentazioni. Si prosegue quindi illustrando il layout dell'ADC e la realizzazione della struttura di test, spiegando come i clock sono distribuiti, come il readout è organizzato e quali variazioni del design dell'ADC sono state aggiunte a scopo di test nel chip e perché.

Il Capitolo 8 conclude la dissertazione presentando i risultati delle simulazioni dell'ADC in termini di non-linearità integrale e differenziale, di rumore in ingresso e di consumo di potenza, evidenziando possibili miglioramenti di design per future implementazioni.

Introduction

The world of digital imaging has seen a steady growth in the past years, with CMOS Image Sensors (CIS) leading the expansion thanks to their high versatility and their capability to give high speed performance without compromising the image quality. The variety of applications in which CIS are used is increasing, ranging from mobile phone cameras to the automotive field (e.g. Advanced Driver Assistance Systems, "ADAS") and to scientific applications - such as aerospace, ballistics and microscopy. In all of these applications, the demand for ever improving speed performance is a major requirement and challenge.

One of the key components in the readout chain of a CIS is the Analogue-to-Digital Converter (ADC), whose speed directly impacts the achievable frame-rate. A CIS can host several ADCs in order to allow for parallelization of the readout (i.e. the simultaneous conversion of the information from multiple groups of pixels): the most commonly employed solution is the column-parallel topology, with one ADC assigned to each column of the pixel matrix.

In an ADC for image sensors good noise performance, reduced size and contained spread of the converters' performance characteristics are required in addition to high conversion speed. Meeting these specifications is a challenge, since improving one (e.g. conversion speed and limited spread) often leads to a degradation of the other (such as size and noise).

Sigma-Delta ($\Sigma\Delta$) ADCs are a promising solution to overcome these limitations. Despite having been suggested for image sensors as early as 1997 [1], these converters have become of interest to the image sensors community only in recent years: traditionally, this type of ADC had been used in other applications, for example audio.

$\Sigma\Delta$ exploit a well known property of feedback loops, i.e. that non idealities in the direct path, such as spread and distortion, have a low impact on the overall transfer, since the signal is mostly transferred through the feedback path. $\Sigma\Delta$ ADCs apply this concept in a mixed analogue-digital system, where the direct path is completely analogue whereas the output and the feedback signal are quantized: their peculiar architecture and the use of oversampling (i.e. sampling the input at a frequency higher than the minimum) allow to perform conversions able to provide the desired resolution while achieving at the same time low noise and good tolerance to spread without giving up area efficiency.

This dissertation describes the design of a $\Sigma\Delta$ ADC for column-parallel image sensors, able to give 12-bits resolution in the competitive conversion time of $1\mu s$. The system was realised using TowerJazz $0.18\mu m$ process for CMOS Image Sensors. It was designed with the CMOS Sensor Design Group at the Rutherford Appleton Laboratory, in Harwell (Oxfordshire), who design state of the art, low noise CMOS image sensors and radiation detectors intended for scientific applications.

This work is in eight chapters. Chapter 1 describes the fundamentals of operation of CMOS image sensors. The topics reviewed in this chapter target aspects that will be recalled in the following discussions regarding ADC design.

Chapter 2 deals with ADCs in general and their relation to the sensor's frame-rate. It starts with an overview of the main figures-of-merit of ADCs, followed by a review of the main readout topologies in CIS, explaining how they can improve a sensor's performance. Subsequently, the most relevant trade-offs in ADCs for CIS are analysed, providing the argument for why $\Sigma\Delta$ ADCs promise to be an improvement with respect to other solutions. At the end of the chapter the target specifications of our design will be described.

Chapter 3 analyses the basic Sigma-Delta architecture in depth, providing the reader with the background necessary to understand its main features (i.e. oversampling and noise-shaping), the filtering of input noise and some linearity issues related to non-ideal components. The chapter also introduces a particular type of $\Sigma\Delta$ converter, called Incremental Sigma-Delta, which is commonly used in CIS and was the adopted solution for this project.

Chapter 4 continues the analysis of $\Sigma\Delta$ ADCs, expanding the view to complex architectures. After describing and comparing different solutions, the final architecture of the converter implemented in this project is chosen, and the specifications for the analogue components - derived through system-level behavioural simulations – are listed. In particular, the ADC architecture chosen was a 2nd order Incremental Sigma Delta performing 100 cycles per conversion, thus working at a frequency of $100MHz$.

Chapter 5 is dedicated to switched capacitor circuits: the purpose of this chapter is explaining the principle of operation of such systems and justifying the formulas that are used in the following analogue design chapter, where the discussion can thus be focused on the design flow.

Chapter 6 deals with the analogue design of the system, i.e. with the implementation of the modulator. This block is composed of two switched capacitors integrators and a comparator, which are the core of the $\Sigma\Delta$ operation.

Chapter 7, on the other hand, deals in the first part with the digital design of the converter's decimator and with the synthesis of the system clocks, in particular explaining the measures taken to guarantee the synchronization of different signals at different supplies. The second part of this chapter covers the layout of the ADC and the realization of the test structure, explaining how the clocks are distributed, how the readout is organized and which design variations of the ADC were added for test purposes and why.

18

Chapter 8 concludes the dissertation by providing results from simulations of the ADC performance in terms of integral non-linearity, differential non-linearity, input noise and power consumption, and commenting about possible design changes for future improvements.

Chapter 1

CMOS Image Sensor basics

Before starting the discussion regarding Analogue-to-Digital Converters (ADCs) for CMOS image sensors, it is convenient to introduce the fundamentals of operation of the sensor itself. This chapter provides a brief description of a basic pixel in a CMOS image sensor and some of the most relevant noise sources in it. The topics reviewed in this chapter target aspects that will be recalled in the following discussions regarding ADC design.

1.1 Principle of operation

Figure 1.1 shows the basic architecture of an active pixel, employing three transistors. It works in three phases, whose diagram is shown in Figure 1.2:

- Reset: the voltage across the photodiode V_{PD} is brought to V_{reset}.
- Light integration: the light shining on the photodiode generates electron-hole pairs at a rate proportional to the incoming photon rate; each carrier will drift to the point of lower potential energy (ground for holes and the photodiode n-well for electrons), thus generating a current which gradually discharges the capacitance C_{PD} (which includes the photodiode's inversion capacitance, the gate capacitance of M2 and all other parasitic capacitances).
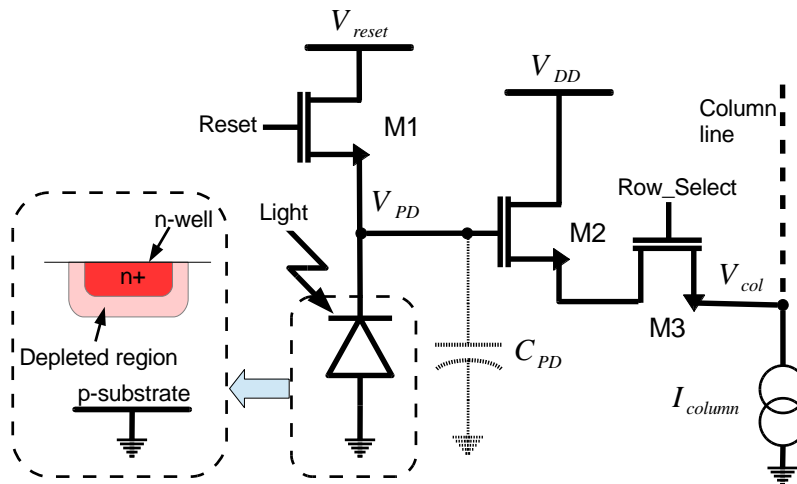
Figure 1.1 Basic architecture of a 3-transistors pixel
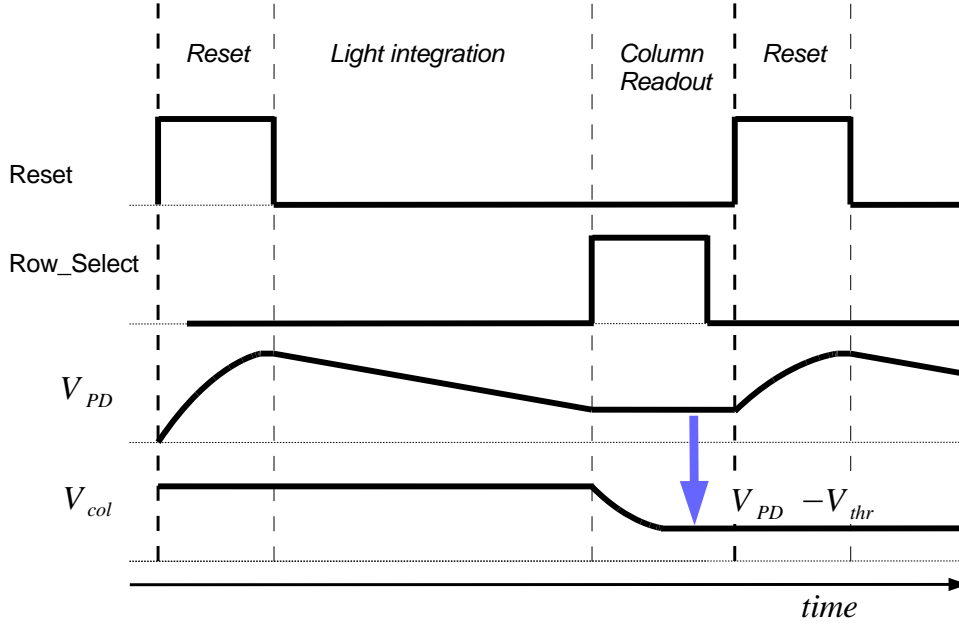
Figure 1.2 Time diagram of a pixel operation

- Column readout: the switch Row_Select connects the source of M2 to the column bias current; M2 will hence turn on, acting as a source follower and buffering V_{PD} (offset by the threshold voltage V_{thr} of M2) to the column line. Here, the buffered voltage V_{col} will be sampled on a capacitor which will hold the information for the readout while the pixel is reset once again.

1.1.1 Non-linearity

The charge generated in the photodiode during the light integration phase is proportional to the incoming radiation intensity and the integration time T_{int}. However, the value read out of an active pixel is the voltage across the reverse-biased p-n junction, and its relation with the collected charge is – within reasonable approximation - quadratic rather than linear, as shown in Eq. (1.1): this is due to the fixed spatial charge in the junction's depleted region.

$$V_{PD} = -V_{bi} + \frac{(Q_{dep} - Q_{ph})^2}{A^2 \cdot 2q\varepsilon_{Si}N_A} \tag{1.1}$$

In Eq. (1.1) V_{bi} is the built-in potential across the junction, Q_{dep} the charge in the depleted region after reset, Q_{ph} the photo-generated charge, A the effective photosensitive area of the pixel, q is the charge of the electron, ε_{Si} the electric permeability of silicon and N_A the acceptor doping of the substrate.

As a consequence of the photodiode's voltage-charge relation, the bottleneck of an imager's linearity is often the pixel itself, thus loose requirements with respect to this specification are allowed for other components in the readout chain.

1.1.2 Full well and dynamic range

The photodiode can only retain a finite amount of charge in its n-well before becoming forward biased and saturating, which is normally referred to as the "full well". This parameter is key to the sensor's dynamic range, i.e. the ratio between the maximum and the

minimum measurable, which for a pixel can be given by the ratio between the full well and the minimum noise.

1.2   Noise contributions

This section will review some of the main noise sources in a pixel: shot noise, reset noise and fixed pattern noise. The topics are limited to those that are relevant for the following discussions; other types of noise, such as $1/f$ and Fano noise are not mentioned although their contribution might be significant - especially $1/f$. These are all treated in depth in [2].

1.2.1   Shot noise
Shot noise is associated to single events – such as the arrival of a photon - occurring at a certain rate. This type of process is described by Poisson statistics, and the main result is that the variance of the number of incoming photons is equal to their mean value:

$$\sigma_{N_{ph}}^2 = N_{ph} \tag{1.2}$$

Therefore, even if the number of incoming photons N_{ph} during the integration time T_{int} was measured with extreme precision, the associated photon rate $r_{ph} = N_{ph}/T_{int}$ would still have some uncertainty: it is in fact possible that the light source intensity was higher (or lower) and just happened to emit a low (or high) amount of photons in the finite measurement time. The noise variance of the light source emission rate is thus:

$$\sigma_{r_{ph}}^2 = \frac{r_{ph}}{T_{int}} \tag{1.3}$$

A consequence of the proportionality between noise power and signal magnitude shown in Eq. (1.3) is that, for large input signals, the dominant source of noise will be shot noise.

Shot noise is also introduced by the photodiode's leakage current, which is caused by thermal generation of carriers; this noise is significant for low signal, thus contributing to the pixel's minimum readout noise.

1.2.2   Reset noise
When the switch resetting the photodiode is on, its finite resistance introduces thermal noise, thus causing a fluctuation of the charge stored on the photodiode. Every time the switch is turned off, this noise charge will remain sampled on the photodiode and be read out together with the signal. This contribution is generally referred to as reset or kTC noise.

The variance of reset noise depends on the total capacitance C_{PD} connected to node V_{PD}, according to Eq. (1.4). This formula will not be derived here: we refer instead to Chapter 5.7, where the impact of white noise sampled on capacitors (of which reset noise is a special case) is treated in depth.

$$\sigma_Q^2 = kTC_{PD} \tag{1.4}$$

1.2.3   Fixed Pattern Noise

Random variations in the chip fabrication process will cause each pixel to have different characteristics and each readout path to have a different offset and conversion gain: this causes fixed pattern noise (FPN). Unlike other noise contributions, FPN affects the transfer in the same way at every readout, and it can hence be strongly reduced by a calibration of the sensor.

Offset

Offset FPN can be caused for example by the spread of the source follower's threshold voltage and – in chips hosting many ADCs, which will be discussed in Chapter 2.2- the spread of the ADCs input offset. It can be expressed as:

$$\sigma^2_{FPN-offset} = \sigma^2_{V_{os}} \cdot \frac{C^2_{PD}}{q^2} \tag{1.5}$$

In Eq. (1.5), $\sigma^2_{FPN-offset}$ is the noise variance in electrons, $\sigma_{V_{os}}$ the standard deviation of the offset in Volts – referred to node V_{PD} – q the charge of an electron and C_{PD} the total capacitance at node V_{PD}.

Gain

Some sources of gain FPN are variations in the pixels sensitivity (such as its charge collection efficiency) and in the conversion gain of each stage of transduction of the signal, e.g. the LSB of the ADCs. The variance of such contribution is proportional to the square of the signal:

$$\sigma^2_{FPN-gain} = \sigma^2_{G} \cdot \eta \cdot N_{ph} \tag{1.6}$$

In Eq. (1.6) $\sigma^2_{FPN-gain}$ is the noise variance in electrons, σ^2_{G} the variance of the gain, η the quantum efficiency and N_{ph} the number of photons arrived at the pixel.

Chapter 2

Analogue-to-Digital-Converters for image sensors

The aim of this chapter is to introduce the reader to Analogue-to-Digital Converters (ADC) for image sensor applications, outlining typical topologies that are employed in imagers and the typical requirements and trade-offs in their design. After this overview, the purpose of the project depicted in this dissertation and the specifications for the ADC developed will be explained.

2.1   ADCs figures of merit

Before entering the discussion about digital converter architectures and readout, it is important to review the main figures of merit of Analogue-to-Digital Converters (ADC).

An ADC is the interface between the analogue world and the digital world in an electronic system: it takes an analogue value at the input and outputs a digital binary number proportional to the input level within a fixed scale, the span of which is called full scale range (FSR). The main instrument to assess its operation is the transfer characteristic, or transfer curve, which plots the output code as a function of the input. The transfer curve of an ideal ADC is shown in Figure 2.1: the interval between two successive transitions is called the least-significant-bit[1] (LSB) of the ADC, since it corresponds to an increment of $2^0 = 1$ in the digital output.

---

[1] This term is a bit ambiguous since it refers to a bit but, rather than being a non-dimensional number, it is measured with the same units as the input; for this reason it is sometimes replaced by the term analog-to-digital-unit, ADU.
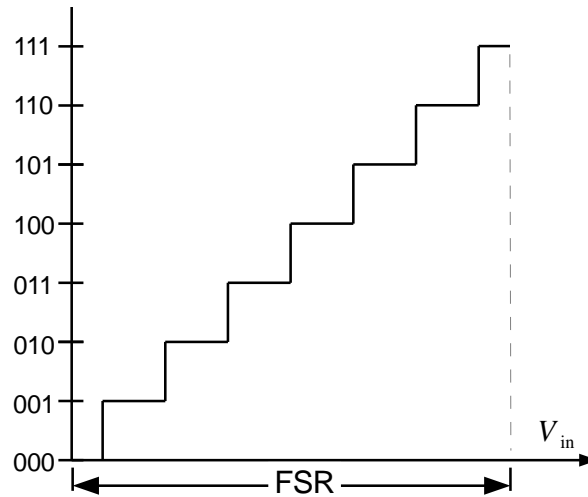
Figure 2.1 Trans-characteristic of an ideal ADC with 3 bits of resolution

Real ADCs however will suffer from some imperfections due to the complex nature of the electronic components that constitute them. Their transfer curves will be qualitatively similar to that in Figure 2.2, where the width of each step changes along the scale. The approximation of the input to a digital value (i.e. its quantisation) thus won't be optimal for a given number of coding bits. Several figures of merit exist to express quantitatively the conversion quality of an ADC: the main ones are the effective number of bits (ENOB), differential non linearity (DNL) and integral non linearity (INL).
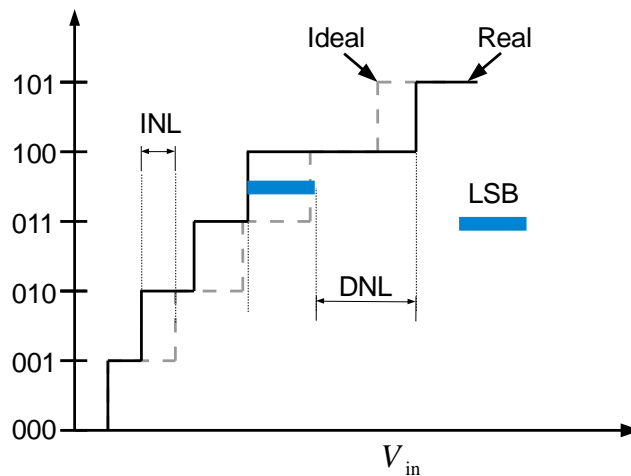
Figure 2.2 Transfer curve of an ADC with INL and DNL in evidence

2.1.1 ENOB

The effective number of bits (ENOB) is the number of bits that an ideal ADC would have to give the same overall signal-to-noise (SNR) of the real ADC, including all the performance-degrading contributions, such as electronic noise, harmonic distortion and quantization error [3].

Given the peculiar nature of quantisation noise, which is correlated to the signal, it is not trivial to consider its contribution together with other non-idealities. Hence, for simplicity in

many cases only the quantisation noise is considered when assessing the ENOB, i.e. only the signal-to-quantisation-noise (SQNR) is calculated. In this dissertation, this is the procedure considered.

For an ideal ADC, a typical assumption is to consider the quantisation error to be uniformly distributed in an interval $[-LSB/2 , LSB/2]$, thus having a standard deviation:

$$\sigma_Q^2 = \frac{LSB^2}{12} \tag{2.1}$$

For a real ADC with a DC input, the ENOB can thus be calculated as the number of bits for which the measured rms of the quantization error equals that of an ideal ADC:

$$2^{ENOB} = \frac{\Delta V_{in}}{LSB_{id}} = \frac{\Delta V_{in}}{\sigma_Q \cdot \sqrt{12}} \tag{2.2}$$

2.1.2 DNL

Differential non linearity (DNL) is the difference between the actual step width and the LSB, normalised to the LSB (see Figure 2.2).

It can happen that a code is skipped in the trans-characteristic (this case is normally referred to as missing code): in this case, DNL=-1.

2.1.3 INL

Integral non linearity (INL) for a certain output code is measured as the difference between the inputs of a real ADC and an ideal ADC which give a transition to the same code in the transfer curve (see Figure 2.2). It is also usually measured in LSBs, although sometimes it is expressed as a percentage of the full scale.

For practicality of the measurement, the transfer curve of the ideal ADC is sometimes replaced with the best fitting straight line or with an endpoint line (i.e. a straight line connecting the first and last points of the transfer curve). In this dissertation the latter method is used.

If INL is defined as a deviation from the endpoint line, then INL at the kth code is the integral of the DNL of all codes up to k.

2.2 Readout topologies for fast imagers

2.2.1 Frame rate versus conversion and readout time

The frame time of a camera depends on the conversion time of all the analog pixel values into digital numbers (T_{conv}) and the time needed to readout all the resulting bits, (T_{RdOut}). The time to obtain a whole frame T_{frame} is a combination of these two times: if the two operations are performed in series (see Figure 2.3 (a)) then $T_{frame} = T_{conv} + T_{RdOut}$; if they are performed in parallel (see Figure 2.3 (b)), then $T_{frame} = \max(T_{conv} , T_{RdOut})$. In the following discussion the former of the two relations will be used. Note that the time between the selection of a row and the complete settling of the pixel's output voltage is

being neglected: while it is, in practice, comparable to the other times T_{conv} and T_{RdOut}, it is in our interest to simplify the analysis and leave the focus entirely on the data conversion and readout.
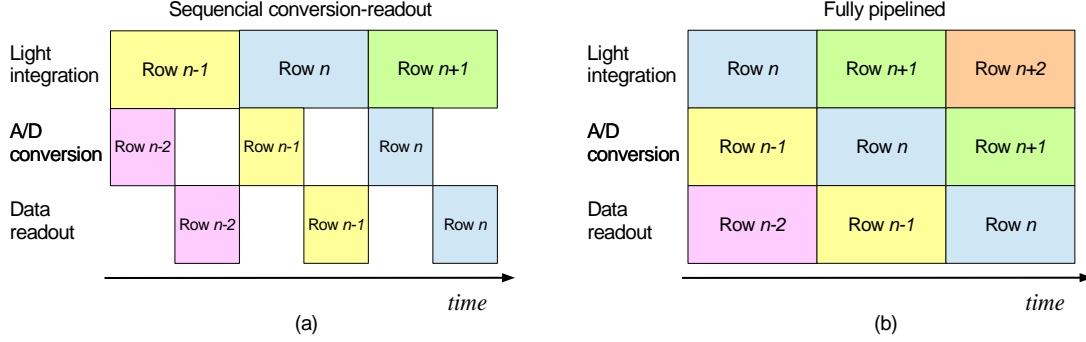
Figure 2.3 Pixel readout time diagrams. Conversion and readout operated in series (a) and conversion and readout operated in parallel (b)

The expressions for T_{conv} and T_{RdOut} are [4]:

$$T_{conv} = b \cdot T_{ADC} \tag{2.3}$$

$$T_{RdOut} = N_R \cdot N_C \cdot \frac{n_{bits}}{n_{parallel}} \cdot \tau_{bitOut} \tag{2.4}$$

In Eqs. (2.3)-(2.4) N_R and N_C are the number of rows and columns respectively, n_{bits} the number of bits of one digitally converted value, $n_{parallel}$ the number of bits that can be read out at the same time and b is a "burden" factor, defined as the number of pixels assigned to one ADC; τ_{bitOut} is the time necessary to deliver one bit to the output, whereas T_{ADC} is the conversion time of one ADC.

T_{conv} is usually dominant compared to T_{RdOut}, which constitutes a lower limit for T_{frame}, since the former typically entails delicate analogue operations whereas the latter simply represents the streaming of digital information. For this reason, the majority of design efforts are addressed towards reducing T_{conv} by working on both the burden factor b and the ADC latency T_{ADC}. The former is reduced by employing highly-parallelized topologies - some of which will be described in this section - while the latter by adopting smart and fast architectures for the ADC: the most commonly used options will be discussed in Section 2.3.

The basic conversion topology for CMOS image sensors employs only one ADC in the whole chip, converting the pixel outputs one by one. While there are clear advantages from a design point of view in terms of area budget and FPN, the achievable frame rate is heavily limited by the lack of parallelization: the ADC has to carry the burden of converting all the pixels in the chip in a serial way. For this topology, following Eqs. (2.3)-(2.4), we therefore have:

$$b = N_R \cdot N_C \tag{2.5}$$

$$T_{frame} = N_R \cdot N_C \cdot \left(T_{ADC} + \frac{n_{bits}}{n_{parallel}} \cdot \tau_{bitOut} \right) \tag{2.6}$$

In order to reach higher frame rates without trading off image resolution it is necessary to resort to topologies with higher parallelism, i.e. with many converters on the chip, each one having only a few pixels assigned to it. However, as it is often the case when increasing the speed of an electronic system, these topologies also give power dissipation and area occupation issues. A particular case where there is a trade-off between area and T_{conv} is the stacked chip sensor, which will be examined in Section **2.2.3**. Moreover, FPN could increase due to statistical spread in parameters of the ADCs (such as offset and conversion gain), which may therefore require inconvenient calibrations.

The main topologies used for fast readout will now be overviewed: these are the column-parallel, stacked chip and in-pixel ADC topologies.

2.2.2   Column-parallel

In this topology, the sensor has N_C ADCs, each one assigned to one column of the pixel matrix, as shown in Figure 2.4 (although sometimes one column has assigned even more than one ADC). The burden factor and T_{frame} are thus, respectively:

$$b = N_R \tag{2.7}$$

$$T_{frame} = N_R \cdot \left(\text{T}_{\text{ADC}} + N_C \cdot \frac{n_{bits}}{n_{parallel}} \cdot \tau_{\text{bitOut}} \right) \tag{2.8}$$

Column parallel image sensors are very commonly used, since they give a good compromise between routing simplicity, contained FPN and speed.
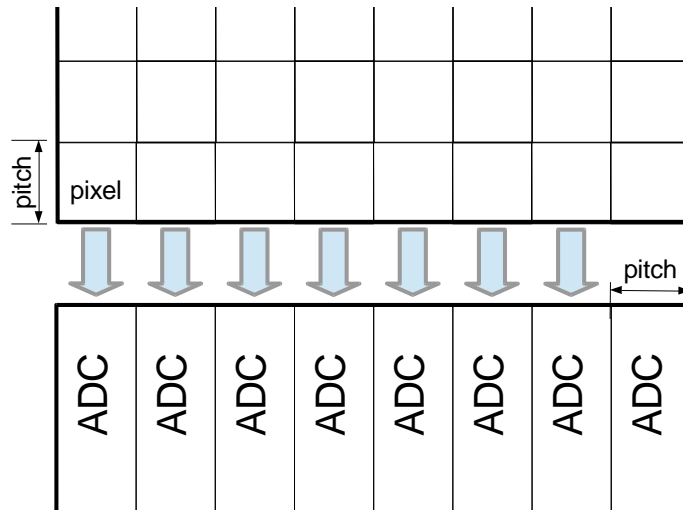
Figure 2.4 Column parallel architecture diagram

2.2.3   Stacked chip

A further step towards parallelisation is assigning a small group of pixels to one ADC: one way to make this possible is having two separate chips, one for the light-to-charge transduction and one for the readout: the former would be sitting on top of the latter, and the two would be connected with through-silicon vias (TSV). Figure 2.5 provides a visual understanding of the concept.

The technological challenges related to the process made the implementation of such a topology problematic until relatively recent years. However, this technology has now been employed in some commercial image sensors [5].

In this case we have:

$$b = \frac{N_{pixels}}{N_{ADC}} = \frac{Area_{ADC}}{Area_{pixel}} \tag{2.9}$$

$$T_{frame} = \frac{N_{pixels}}{N_{ADC}} \cdot \text{T}_{ADC} + N_R \cdot N_C \cdot \frac{n_{bits}}{n_{parallel}} \cdot \tau_{bitOut} \tag{2.10}$$

Combining Eqs (2.9)-(2.10) it can be shown that there is an inherent trade off between ADC area and conversion time, given by the fact that a large ADC would have more pixels to convert per frame (since there would be many of them lying on top of it), thus would need to be faster to achieve the same frame rate. This is expressed quantitatively in Eq. (2.11), where for simplicity it was considered $\tau_{bitOut} \sim 0$.

$$Area_{ADC} \cdot T_{ADC} = Area_{pixel} \cdot T_{frame} \tag{2.11}$$

Note that this is not the case for the column parallel topology where, while the pitch has to match the pixel's width, the ADC length could in principle be un-constrained.
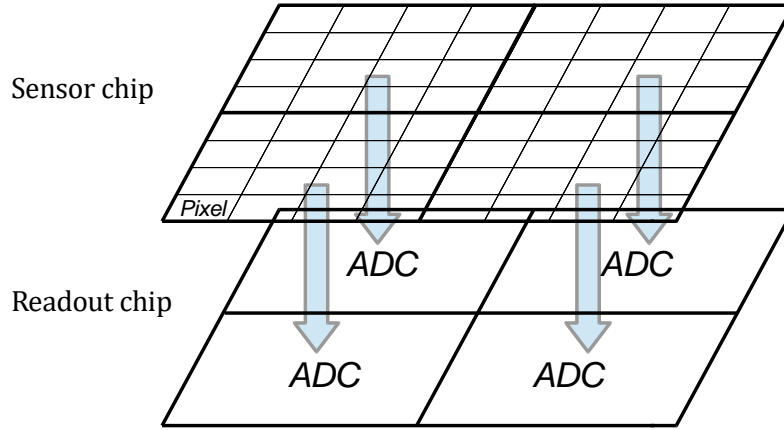
Figure 2.5 Conceptual illustration of a stacked chip

2.2.4 Pixel-level ADC

The ultimate step in terms of parallelisation is when each pixel can rely on one ADC for the conversion: in this case, the dominant factor in T_{frame} is more likely to be the readout time. We have:

$$b = 1 \tag{2.12}$$

$$T_{frame} = T_{ADC} + N_R \cdot N_C \cdot \frac{n_{bits}}{n_{parallel}} \cdot \tau_{bitOut} \tag{2.13}$$

This solution has however several drawbacks: firstly, the displacement of charge in the substrate due to the operation of the ADC could interfere with the pixel's activity; secondly, it reduces the fill factor of the pixel (the ratio between active photodiode area and overall pixel area, comprising of circuitry) and, consequently, its sensitivity; lastly, the necessity to bring out an n_{bits} output for each pixel substantially complicates the layout of the routing, which can also potentially disturb the activity of the pixels. Despite these challenges, ADCs of this kind have been realised [6] and used in commercial sensors [7].

An interesting point can be made regarding the development of pixel-level Sigma-Delta ADCs. These ADCs need an analogue integrator to perform the conversion: in a pixel, the photodiode itself integrates the light intensity shining on it; therefore, ADCs of this kind could be realised inside the pixel, and the converted signal could be the light directly shining on the pixel rather than its output voltage. Moreover, in this sensor the negative feedback necessary for Sigma-Delta operation would be given by packets of charge partially resetting the photodiode: this would enable the pixel to receive more signal than its full well, thus effectively increasing the dynamic range [8].

2.3   Trade-offs in ADC architectures for image sensors

The discussion in Section 2.2 showed that to achieve high throughput the chip needs to host a large number of ADCs. As a consequence, together with providing a high conversion speed, an ADC for image sensors needs to be able to comply with other important requirements:

- small area occupation
- low spread in parameters (hence low susceptibility to device mismatch)
- low input noise.

The purpose of this paragraph is to analyse how some of the most commonly used ADC architectures for image sensors deal with these constraints and to argue why the converter of the type developed in this project – a Sigma Delta ($\Sigma\Delta$) ADC - promises to be an improvement from this point of view.

The most common architectures used for image sensor converters are ramp, SAR, cyclic, pipeline and Sigma-Delta ADCs [4]: while it is not in the scope of this dissertation to review them all, two of these architectures will now be briefly overviewed by way of example in relation to the constraints listed above: these are ramp and SAR ADCs.

Ramp ADCs (Figure 2.6) use a comparator, a ramp (a voltage source increasing linearly with time) and a counter: the converted value is the output of the counter, given by the time needed by the ramp to overcome the input of the ADC.

In order to obtain an n_{bits} output, $2^{n_{bits}}$ clock cycles are necessary; to improve the speed, many image sensors use multiple slope ADCs: in these ADCs, a coarse conversion with a fast ramp defines the first m most significant bits (MSB), which are used to set the offset of the ramp of the following conversion giving the remaining bits.

Despite its long conversion time, this type of ADC is widely used in image sensors thanks to its low area requirement, its simplicity and the predisposition for parallelism: in fact, many ADCs can share the same ramp voltage source, thus eliminating its contribution to FPN. This is not true, however, for most multi-slope architectures, where the spread of misalignment between the coarse conversion ramp and the fine conversion ramp can increase FPN [4].
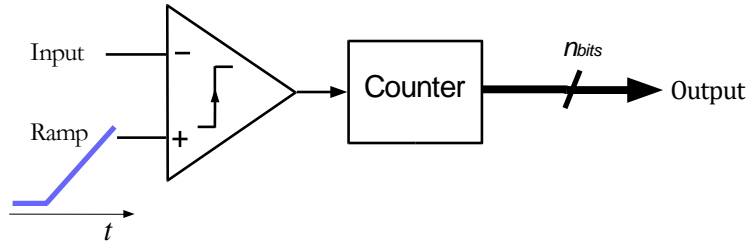
Figure 2.6 Ramp ADC

Successive Approximation Register (SAR) ADCs (Figure 2.7), instead, feature a comparator, a $2^{n_{bits}}$ Digital-to-Analogue converter (DAC) and a digital block for the control logic. The digital output is obtained by a recursive comparison of the input to a variable reference voltage in the following way: after the kth cycle, the results of the comparisons operated by the comparator form a k-bits binary representation of the ADC input; during cycle $k+1$ this number is converted to an analogue voltage by the DAC and subsequently compared to the input to obtain the following the $(k+1)^{th}$ bit. Therefore, at each cycle one bit of resolution is added and only n_{bits} clock cycles are necessary to complete a conversion.

Their characteristics are complementary to a ramp ADC: they are significantly faster but have higher spread due to the unavoidable mismatches in the DAC components; furthermore, reducing the spread usually leads to designs of large area (since larger devices statistically have lower relative mismatch), making ADCs of this type hardly area-efficient.
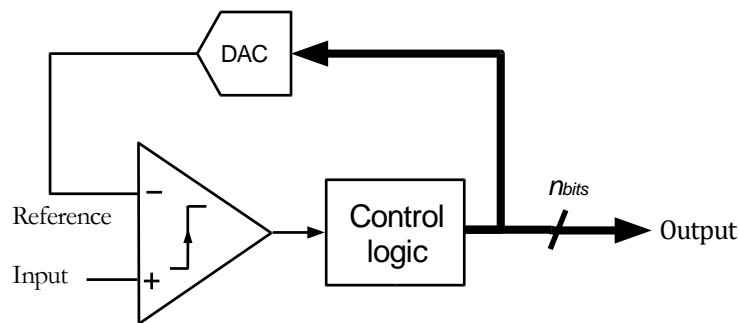
Figure 2.7 SAR ADC

The examples of SAR and ramp ADCs show that there is a trade-off between high conversion speed, limited spread and good area efficiency: SAR ADCs tend to prioritize the speed performance, while ramp ADCs are a good compromise in terms of mismatch and size containment. Furthermore, in both ADCs the low noise constraint requires the comparator to have an input noise of the order of the LSB (the case of SAR ADCs is particularly delicate, since an incorrect firing of the comparator can affect a significant bit if it occurs early in the conversion), which can be a rather strict requirement at large bandwidth.

Sigma Delta ($\Sigma\Delta$) ADCs have the potential to circumvent all of these limitations: as will be thoroughly explained in Chapter 3 and Chapter 4, the impact of the mismatch of most components is strongly reduced by noise shaping; the input noise is reduced thanks to oversampling and is not contributed to by the comparator, which can have very poor noise performance without affecting the conversion; since they don't require a $2^{n_{bits}}$ DAC, their structure can be fairly simple and their size small with respect to a SAR - although they are not as competitive in terms of speed. Moreover, all of these characteristics make $\Sigma\Delta$ ADCs suitable to be realized with scaled technologies without compromising their performance.

2.4   Purpose of the project

2.4.1   Objective
The CMOS Sensor Design Group (CSDG), working at the Rutherford Appleton Laboratory, in Harwell (Oxfordshire) designs state of the art, low noise CMOS image sensors and radiation detectors intended for scientific applications.

The next generation of image sensors will need to be able to achieve high data rates while maintaining good resolution and satisfying increasingly demanding noise requirements: in order to make this happen, the CSDG has decided to follow the image sensor community's increasing interest in $\Sigma\Delta$ ADCs as components to be employed in the next generation of image sensors, and to investigate the feasibility of an ADC employing this architecture.

The objective of this work is to investigate the main characteristics of $\Sigma\Delta$ ADCs, assess their feasibility in image sensor applications and lastly, design such a converter and its test chip.

2.4.2   Specifications
The ADC would need to be able to give a resolution of at least 12 bits, to have an input equivalent rms noise lower than $100\mu V$ and to dissipate less than $330\mu W$. The specification for DNL was set to $DNL^{(max)} < 1$. In terms of integral non-linearity, ADCs for image sensors have generally loose specifications: as explained in Chapter 1 in fact, the main source of non-linearity in an active pixel is the photodiode itself, because of its non-linear charge-voltage relation. As a consequence, relatively high values of integral non-linearity (INL) are tolerated compared to other applications: $INL < 0.5\%$.

FPN can be eliminated through calibration, although this operation can be inefficient in sensors where it is severe. For the ADC developed in this project, no FPN specification was set, but the aim was to implement a converter that would require as little calibration as possible.

Topology and size-conversion time trade-off
As already stated in Section 2.2, state of the art high data rate imagers mainly employ the column parallel architecture. The quest for ever-increasing frame-rate however, makes it interesting to investigate a different configuration - the stacked chip topology seen in Section 2.2.3. The ADC developed in this project should have characteristics that make it compatible with both solutions, provided changes in layout and routing are applied. It was not designed to be associated with a specific imager: in fact, the objective is to make this ADC a stand-alone IP block for future development.

To provide some specifications for layout, we will assume that the initial application for this ADC will be a column-parallel sensor with a pitch of $15\mu m$. Note that this pitch is larger than that seen in commercial imagers, which can have pixel pitches as low as $1\mu m$ [9]: the field of applications for which the sensors delivered by the CSDG are intended is in fact scientific, and in this case requirements such as high yield and large full well capacity (i.e. the maximum charge generated by the photodiode before saturating) are more important than area density.

For the ADC to also be compatible with the stacked architecture, it will have to satisfy the constraint in Eq. (2.11): assuming that the performance wanted by the stacked chip is $22.5k$fps we get the following area vs T_{ADC} curve:
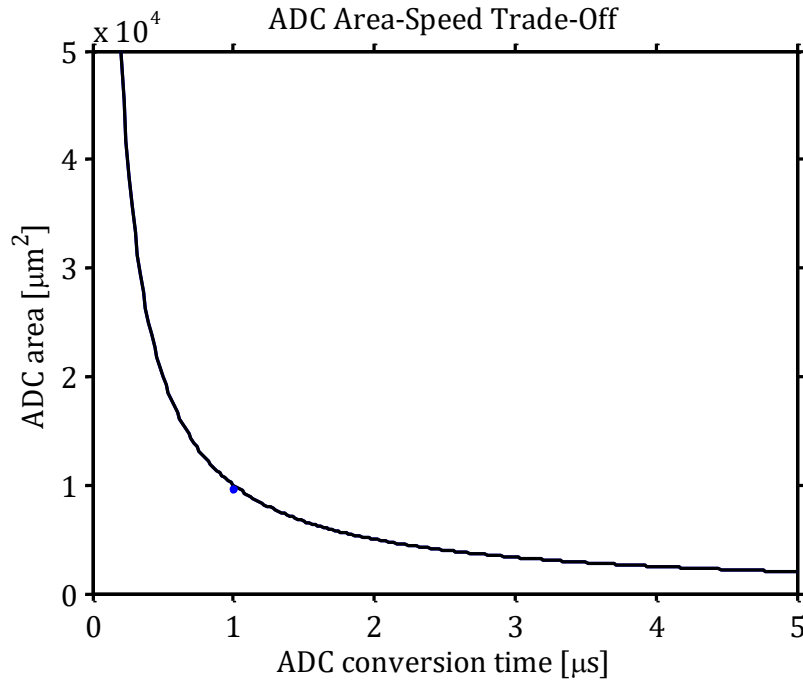
Figure 2.8 Trade-off between ADC area and conversion time. As a specification, the point corresponding to the developed ADC has to lie below the curve.

We can see that an area of $\sim 10000\mu m^2$ corresponds to $T_{ADC} = 1\mu s$: these values were set as specifications for the converter regardless of the topology adopted. In terms of layout, this would mean that a stacked chip ADC is expected to have a size of $100\mu m$ x $100\mu m$, while a column parallel ADC would be $15\mu m$ x $670\mu m$. The test chip designed for this project adopts the latter of the two topologies.

Table 2.1 Global specifications of the developed ADC

Number of bits	12
Conversion time T_{ADC}	$1\mu s$
Maximum DNL	$1\ LSB$
Maximum INL	0.5% or $20\ LSBs$
Power consumption	$330\mu W$
Maximum size	$10000\mu m^2$ or $15\mu m$ x $670\mu m$

2.4.3 Expected achievable frame-rate

It is interesting to estimate for each topology what would be the expected frame-rate of a sensor hosting the ADC developed in this project. Table 2.2 provides a comparison between all the alternatives discussed in Section 2.2. The assessments were carried out using Eqs. (2.3) through (2.13), and assuming $N_R = N_C = 2^{10}$, $Area_{ADC} = 10000\mu m^2$, $T_{ADC} = 1\mu s$, $n_{parallel} = 16$, $n_{bits} = 12$ and $\tau_{bitOut} = 1ns$. Note that the readout time of the pixel is still being neglected, thus the results given in Table 2.2 should be taken only as approximations.

Table 2.2 Estimated frame-rate (in Hz) achievable with the developed ADC. Comparison between topologies.

Frame-rate [Hz]	Serial readout	Parallel (pipelined)
Global ADC	0.95	0.95
Column parallel	552	977
Stacked chip	1204	1272
In-pixel ADC	1270	1272

2.4.4 Tools used

The ADC was first simulated on Matlab[®] Simulink[®] to study its behavior and to derive system level specifications; it was subsequently designed at schematic level with the assistance of Cadence[®] Virtuoso[®], using the design kit provided by the foundry TowerJazz[®] for its 0.18μm CMOS image sensors process.

Chapter 3
Sigma-Delta ADC basics

In Chapter 2 we have introduced the reader to the constraints regarding ADCs for image sensors and illustrated the potential of Sigma Delta ADCs in this field. The purpose of this chapter is to provide the background necessary to understand how these ADCs work and what are issues and advantages related to their design.

3.1   Working principle

3.1.1   Structure: modulator and decimator

A $\Sigma\Delta$ ADC is formed by two stages: a modulator and a decimator, as shown in Figure 3.1.

The input is connected to the modulator, which is composed of analogue circuitry and a quantizer (typically a 1-bit quantizer, i.e. a comparator): together, its components form a feedback loop which is essential to the operation of the $\Sigma\Delta$. The modulator performs oversampling, i.e. it samples the input at a frequency higher than the desired output sample rate: for DC inputs, this means that the input is sampled more than once. Because of this, the output of the quantizer corresponding to one sample is a continuous stream of bits (or digital numbers if the quantizer has more than one bit).

The output of the modulator is then delivered to the decimator; the role of this block is exploiting the redundancy in the bits to get rid of most of the quantization error and, at the same time, reducing the number of bits to be delivered to the output of the ADC. The simplest example of a decimator is a counter, which averages the continuous input bit-stream producing n_{bits} at the output. From now on, the decimator will also be referred to as digital low-pass filter (DLPF).
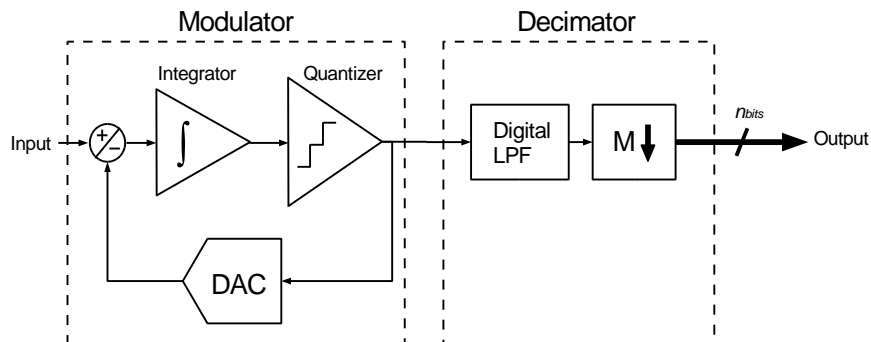
Figure 3.1 General architecture of a Sigma-Delta
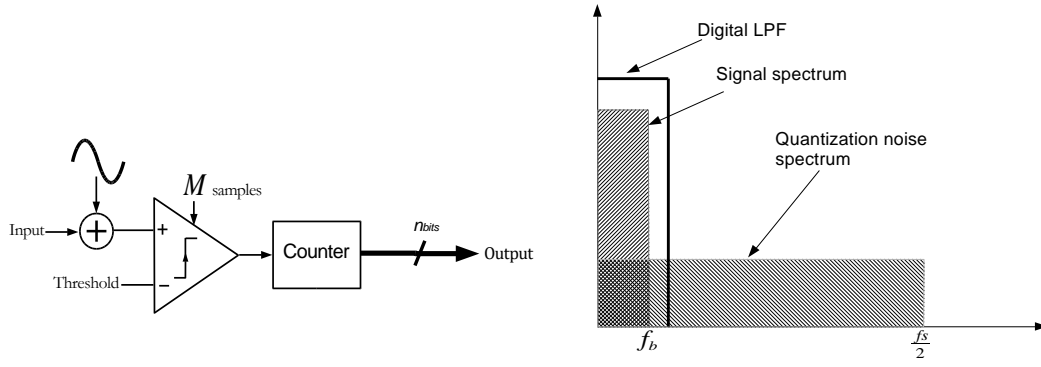
Figure 3.2 Block diagram of a basic oversampler (left) and corresponding signal and quantization noise frequency spectrum

3.1.2 Oversampling - 0th order modulator

It is known that the minimum frequency f_s at which an analogue signal can be sampled without aliasing is $f_s = 2f_b$, f_b being the bandwidth of the input (Shannon-Nyquist theorem, [10]). It is moreover intuitive that sampling at a frequency higher than the minimum, despite being useless on its own, can reduce noise from quantization or other sources if the samples in excess are averaged and then discarded – i.e. if a digital low-pass-filter (digital LPF or DLPF) and a decimator are applied. A known example of this is sampling a DC signal to which a dither is added: multiple samples are taken and then averaged to quench the quantization noise e_q (see the diagram in Figure 3.2).

A quantitative assessment of the improvement given by oversampling alone on the resolution of the ADC can be carried out in both frequency and time domain. Arguments of both types can be found in literature (see for example [11] and [12]); in this section, we will use the frequency domain explanation - which lays the ground for the treatment of the other feature of ΣΔ ADCs, i.e. noise shaping.

Let's consider an ideal ADC with full scale range FSR which operated at the Nyquist limit has a quantization step LSB_{Nyq}: for example, if a comparator is used, it would have a resolution $LSB_{Nyq} = FSR/2$. Our purpose is to study how its resolution can be improved by oversampling.

If the quantization noise is assumed to be uncorrelated to the signal, then the white noise model can be applied: its power spectrum S_q will be constant throughout the whole frequency range, i.e. from 0Hz until $f_s/2$ (the maximum frequency containing information associated to a sampled signal). The necessary condition for this approximation to be valid is that the dithering signal magnitude be significantly larger than the quantization steps in the ADC.

Since the quantization error total standard deviation, computed by integrating its power spectrum from 0Hz to $f_s/2$, must always be equal to $LSB_{Nyq}^2/12$, it has to be that:

$$S_q = \frac{LSB_{Nyq}^2}{12} \cdot \frac{2}{f_s} \tag{3.1}$$

If a digital LPF with cut-off equal to the signal bandwidth f_b is applied, then the expected standard deviation of the noise will be:

$$\left(\sigma_q^2\right)_{OS} = \frac{LSB_{Nyq}^2}{12} \cdot \frac{2f_b}{f_s} = \frac{LSB_{Nyq}^2}{12} \cdot \frac{1}{OSR} \tag{3.2}$$

hence lower than the usual value $LSB_{Nyq}^2/12$ by a factor OSR, called oversampling rate, defined as $OSR = f_s/(2f_b)$. The SQNR (signal-to-quantization-noise-ratio) will consequently increase by the square root of the same factor. The digital filter can be designed to perform also decimation: the redundant sampled values will be incorporated in one digital number composed of more bits. These numbers are hence the result of a conversion finer than that given by the ADC when sampling at the Nyquist frequency: the effective new LSB, LSB_{OSR} , will be given by:

$$LSB_{OSR} = LSB_{Nyq} \cdot \sqrt{\frac{\left(\sigma_q^2\right)_{Nyq}}{\left(\sigma_q^2\right)_{OSR}}} = \frac{LSB_{Nyq}}{\sqrt{OSR}} \tag{3.3}$$

In the case of DC signals, which is the one of interest for the conversion of a pixel output, OSR can equivalently be defined as the number of times M that the input has been sampled. Then:

$$LSB_{OS} = \frac{LSB_{Nyq}}{\sqrt{M}} \tag{3.4}$$

At every doubling of the OSR (or M), half a bit is gained.

3.1.3 Noise shaping

The use of a frequency higher than the minimum does not only cause the noise spectrum to decrease: it also leaves a range of frequencies beyond f_b where no useful information is contained. A ΣΔ modulator exploits this fact by shaping the quantization noise power spectrum, concentrating its energy at high frequencies. The in-band power of noise is hence further reduced.

The only way to shape the spectrum of the quantization noise e_q leaving the signal unaltered is having the two being subject to different transfer functions: a low-pass filter (LPF) for the signal, with cut-off $\sim f_b$, and a high-pass filter (HPF) for e_q. This is possible in a feedback loop where signal and quantization noise are added at different points of the chain. Figure 3.3 shows the simplest loop that can achieve this.
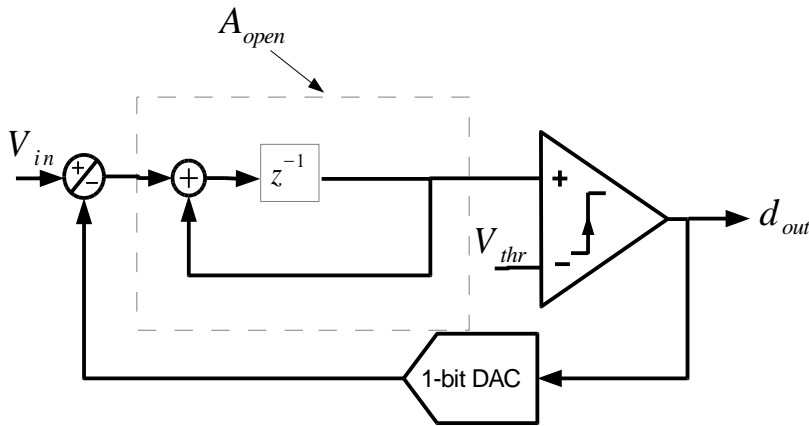
Figure 3.3 Architecture of a first order, time-discrete, Sigma-Delta with binary quantizer
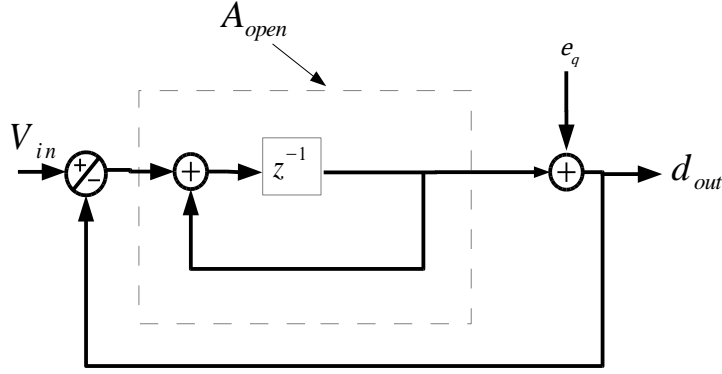
Figure 3.4 Linear equivalent model of a first order Sigma Delta

Quantitative analysis - frequency domain

Regardless of whether the ADC has a time-discrete or time-continuous architecture (this distinction will be examined in Chapter 4.1), it is best to evaluate the effect of the loop on the signal and quantization noise in the discrete frequency domain, i.e. using Z-transforms, since this form is most suitable to study the effect of the following digital filter. If the ADC has a time-continuous behaviour, it is possible to pass from a description in the Laplace domain to the Z domain using conventional methods such as backward and forward Euler transformation or the Tustin rule.

Figure 3.3 and Figure 3.4 show respectively the architecture and equivalent block diagram of a first-order $\Sigma\Delta$ which uses a time-discrete integrator in the forward path of the loop. If, as was done in the previous section, quantization error and signal are assumed to be uncorrelated, then the output can be considered to be given by a linear superposition of the two:

$$d_{out} = STF(z)\, V_{in}(z) +\ NTF(z)\, e_q(z) \tag{3.5}$$

where we have introduced the signal transfer function (STF) and noise transfer function (NTF); it can easily be derived that, for the diagram in Figure 3.4:

$$A_{open} = \frac{z^{-1}}{1 - z^{-1}} \tag{3.6}$$

$$STF = \frac{A_{open}}{1 + A_{open}} = z^{-1} \tag{3.7}$$

$$NTF = \frac{1}{1 + A_{open}} = 1 - z^{-1} \tag{3.8}$$

Therefore:

$$d_{out}(n) = V_{in}(n-1) + \left[e_q(n) - e_q(n-1) \right] \tag{3.9}$$

The quantization error is differentiated, which is in fact a high pass filtering operation.

The noise reduction factor can be computed as:

$$\int_0^{fb} \left| NTF\left(e^{j\omega Ts}\right)\right|^2 df = \int_0^{fb} (1 - e^{-j\omega Ts}) df \approx \frac{\pi^2}{3}\frac{1}{\text{OSR}^3}. \tag{3.10}$$

Therefore:

$$\text{ENOB} = \frac{3}{2}\log_2 \text{OSR} - 0.86 \tag{3.11}$$

Note that in the previous equation the digital decimation filter is assumed to have a boxcar frequency response with cut-off f_b, to simplify computation. This corresponds to the ideal result: in practice, different LPFs provide different resolutions, hence the choice of the digital stage is critical to the ADC performance. Figure 3.5 illustrates the effect of noise shaping on the noise transfer function in the frequency domain.
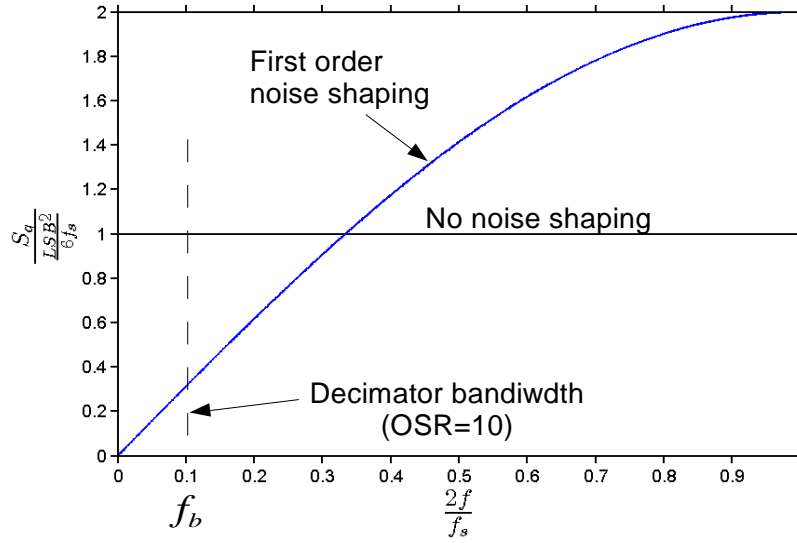
Figure 3.5 Frequency spectrum of the noise transfer function with and without noise shaping

It is important to note that this peculiar feedback has its forward path ending with a quantizer, a very non-linear element. Hence, some caution should be taken in trusting the noise shaping argument. Because of the quantizer's non-linearity, in fact, there are effects that this simple linear model cannot represent, such as limit cycles and dead zones (discussed in Section 3.3.1), that reduce the ADC's INL, DNL and overall SQNR. In principle, the linearity of the $\Sigma\Delta$ (and with it the reliability of the proposed analysis) should improve if multi-bit quantizer and DAC are used, since the DAC output will be a better approximation of an analogue voltage; however, multi-bit DACs are susceptible to mismatch and, since their output is fed to the input node of the $\Sigma\Delta$ and therefore isn't noise-shaped, their non-linearity will directly degrade that of the ADC. From now on the Sigma-Delta ADC will always be assumed to have a binary DAC.

Qualitative analysis - time domain view with a DC input
Figure 3.6 shows the signals in a first order, time-discrete Sigma Delta Modulator (SDM) at the start of its conversion: from top to bottom, the waveforms belong respectively to the output of the integrator V_{INT}, the output of the 1-bit DAC V_{DAC} and the output of the digital LPF (a simple counter in this case). The common mode (CM) of the system, coincident with the threshold of the comparator, is 1, and the DAC voltage - which defines the full-scale

range – can either assume the values $V_{DAC}^H = V_{ref} = 2$ or $V_{DAC}^L = 0$; the input voltage V_{in} in this example is 1.82 DC.

At every cycle, the comparator measures whether the integrator's output is larger or lower than the CM; the DAC voltage will be high in the former case and low in the latter, being given by the result of the logical operation:

$$V_{DAC}(n) = V_{ref} \cdot Truth[V_{INT}(n-1) > V_{CM}] \tag{3.12}$$

Where $Truth(\cdot)$ is 1 if the proposition in its argument is true and 0 otherwise.

At the beginning of cycle $n+1$, the integrator's output will then move by $\Delta V_{INT} = V_{in} - V_{DAC}(n)$: in this example, ΔV_{INT} can be either $+1.82$ or -0.18. Hence, when V_{INT} is above the threshold, at every cycle it will perform a small negative step of -0.18, until it overcomes the comparator threshold and the DAC re-kicks it up by $+1.82$. As a result, V_{INT} will stay above threshold most of the time and the DAC waveform will have more logic 1's than 0's, which will be counted by the DLPF. This is the case in our example because the input is close to V_{DAC}^H; a similar behaviour would be observed for input close to V_{DAC}^L: the core message is that the total time during which the integrator output is higher than the threshold (hence, the number of logic 1's at the comparator's output) is directly related to the position of the input within the full scale.

To understand how noise shaping works and why it improves with increasing cycles, let's consider a "faulty" ΣΔ which has an offset of -0.2 in the threshold of the comparator: this DC signal is added at the same node as the quantization error in the linear model in Figure 3.4, thus we expect it to get high-pass filtered and not have a meaningful impact on the output of the conversion. Looking at Figure 3.6 we can understand why that is the case: this offset does indeed cause the DAC to give the wrong output at some cycles (e.g. $+2$ in place of 0), but the feedback will react and make the DAC compensate at some later cycle (e.g. delivering a $+2$ to compensate for the previous 0), hence the output of the counter in the faulty system will mostly be coincident with that of the ideal system. This will happen periodically, and the number of cycles that separate two epochs of the two DAC waveforms being coincident (only one cycle in the example) is (ideally) proportional to the input offset of the whole ADC (since the input magnitude modulates the occurrence of 1's and 0's, as explained above), and it has to be weighed against the total number of cycles M: the larger this number, the smaller will the impact of the threshold offset on the ADC be.

3.1.4   Stability and full scale range
In a ΣΔ ADC, in order for it to be stable, it is required that the input is bounded by [12]:

$$V_{DAC}^L < V_{in} < V_{DAC}^H \tag{3.13}$$

In fact if, for example, we had $V_{in} > V_{DAC}^H$, then the input of the integrator $(V_{in} - V_{DAC}(n))$ would always be positive regardless of the value of $V_{DAC}(n)$, hence its output V_{INT} would diverge indefinitely – in practice, until reaching saturation.

In other terms, we can say that the FSR of a 1st order ISD is set by the DAC reference voltages.
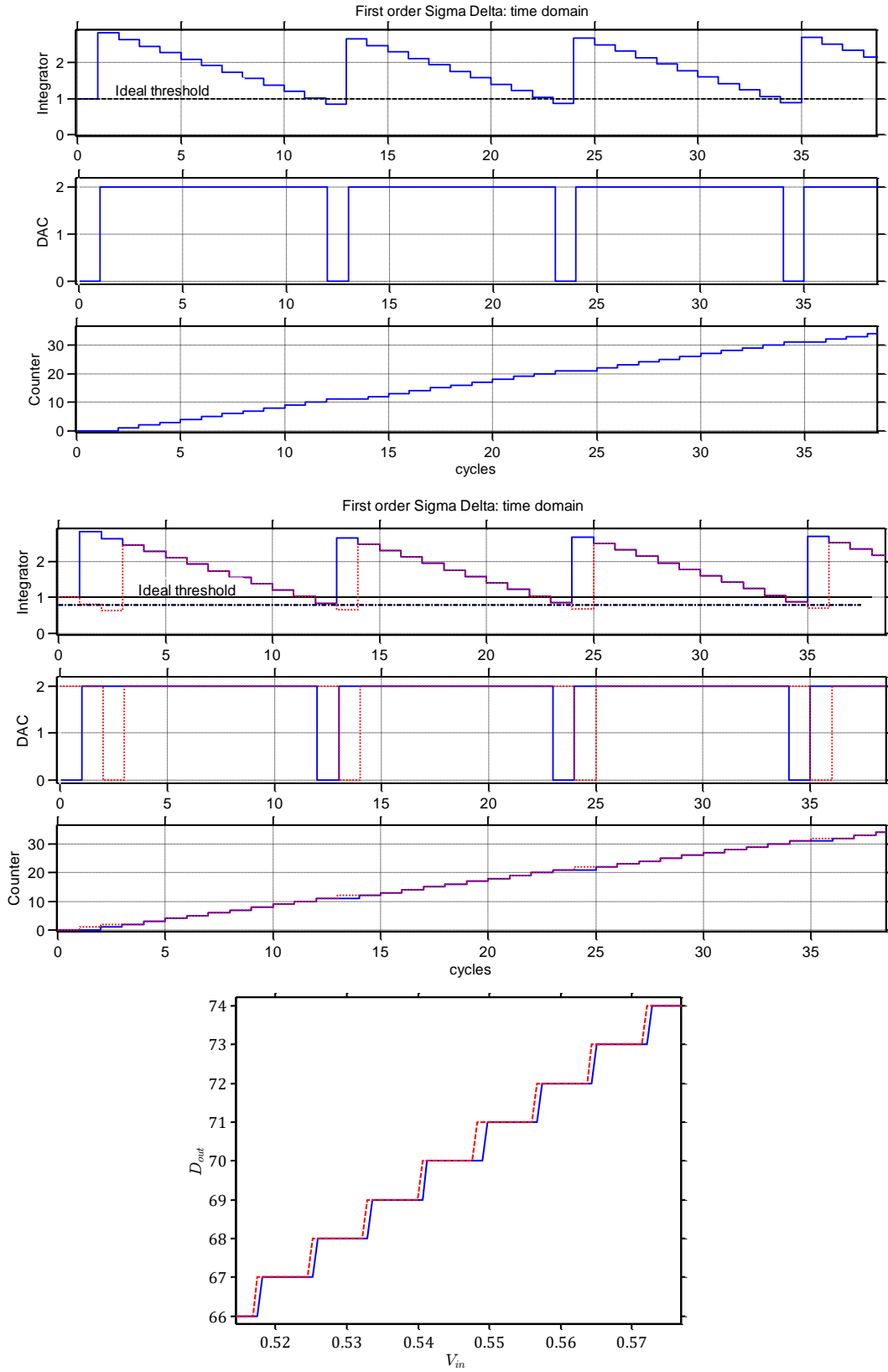
Figure 3.6 Waveforms in an ideal Sigma-Delta (top), with offset at the quantizer input (middle) and comparison of the corresponding transfer curves in the case _M=256_ (bottom).

3.1.5   Input noise

Noise due to analogue components which is added at the input will be transferred at the output by the STF, hence it won't be noise-shaped. However, it will indeed be low-pass filtered by the decimator, and therefore reduced. For white noise, to get the same reduction factor as that of a simple oversampling ADC, the DLPF needs to be a counter, which is the optimum case; for any different filtering, a worsening factor W must be included (see [13],[14], [15]). Specifically, if the standard deviation of the input white noise sampled at every cycle σ_{sample} is known, the output noise power can be computed as a weighted sum of the variances and subsequently be referred to the input:

$$\sigma_{n-out}^2 = \sigma_{sample}^2 \sum_{j=1}^{M} |w(j)|^2 \tag{3.14}$$

$$\sigma_{in}^2 = \frac{\sigma_{n-out}^2}{\left(\sum_{j=1}^{M} w(j)\right)^2} = \sigma_{sample} \cdot \frac{W}{M} \tag{3.15}$$

In order to do so, the weighting function $w(j)$ of the decimator has to be known [15],[14]: if the DLPF is a counter, a 2nd or a 3rd order cascade of integrators, the noise worsening factor will be respectively, $W_1 = 1$, $W_2 = 4/3$ [15] and $W_3 = 6/5$ [14].

3.2   Incremental Sigma Delta

This work will largely focus on a sub-type of $\Sigma\Delta$ ADCs known as Incremental Sigma Delta (ISD). ISD employ a sample-and-hold (S&H) at the input and are reset every time one sample is taken. ADCs of this kind are the natural and, indeed, most frequent choice for image sensors, where the output of the pixel is by default sampled-and-held on a capacitor prior to conversion.

The modulator is generally time-discrete, using switched capacitor integrators. A comparison between time-discrete and time-continuous modulators is given in Chapter 4.1, while an analysis of how switched capacitor circuits work is given in Chapter 5.

The decimator filter employed, especially in the case of image sensors, is usually formed by cascaded integrators (e.g. a counter and an accumulator for a second order filter [16],[17],[13]): as already mentioned, the use of a counter is the best choice to reduce the input analogue noise, however it is not the optimum filter for quantization noise [18]. Cascaded integrators are anyway mostly preferred for their design and layout simplicity, which are essential requirements for image sensors.

The noise-shaping efficiency of ISD converters is lower than that of other ADCs working with time-continuous inputs – rather than the DC input given by a S&H - and employing more complicated filters: as a consequence, a larger number of cycles M is needed to obtain the same ENOB, and the ADC will be marginally slower. We shall derive the relation between M and ENOB for a 1st order ISD (ISD1) in the following paragraph. The case of higher order ISD (ISD2 for 2nd order) will be discussed in Section 4.4.1.

3.2.1 First order ISD resolution analysis

The following discussion is based after [17],[16] and [13]; it is here reported because it is necessary to understand this work and some of the following paragraphs analyzing alternative ΣΔ architectures.
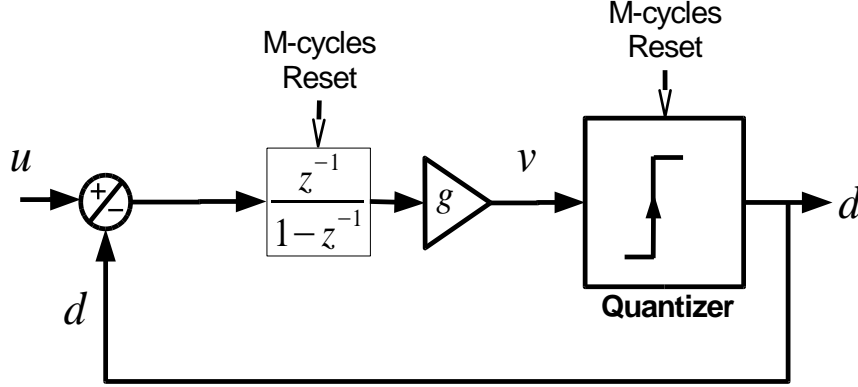
Figure 3.7 First order Incremental Sigma Delta

The behaviour of the ISD modulator is best understood in the time (or samples') domain. An analysis in the frequency domain would in fact need to carefully take into account the presence of the synchronous reset, which unnecessarily complicates the analysis – moreover, understanding the behaviour of the waveforms is useful since these are what the designer will directly evaluate when simulating or measuring.

For this analysis we will assume $V_{DAC}^H = V_{ref}$ and $V_{DAC}^L = 0$, where V_{ref} is an arbitrary reference voltage; the centre of the input span will thus be $V_{CM} = V_{ref}/2$. We furthermore define the normalized quantities $u = V_{in}/V_{ref}$, $v = (V_{INT} - V_{CM})/V_{ref}$ and $d = V_{DAC}(i)/V_{ref}$, which are respectively the scaled versions of the input, the output of integrator and the DAC output. Eq. (3.12) hence becomes:

$$d(i) = Truth[v(i-1) > 0] \tag{3.16}$$

The gain of the integrator will be called g. The comparator output at cycle i will be referred to as $D(i)$, and can either be 0 or 1.

After the M-th cycle, the output of the integrator will be:

$$v(M) = g \cdot \sum_{i=0}^{M-1} [u(i) - d(i)] = g \cdot \left[M \cdot u - \sum_{i=0}^{M-1} d(i) \right] \tag{3.17}$$

Observe that the digital code $d(i)$, whose history will in the end determine the output of the DLPF, is given by Eq. (3.16) and is hence independent of the magnitude of $v(i)$: therefore, the integrator gain g here plays the role of a scaling factor for the signals which has ideally no influence on the ADC performance. This observation will be important to discuss some aspects of the design of the ΣΔ developed in this project. We can hence refer to the re-scaled integrator output $v' = v/g$ and rewrite eq. (3.17) as:

$$v'(M) = \sum_{i=0}^{M-1} [u(i) - d(i)] = M \cdot u - \sum_{i=0}^{M-1} d(i) \qquad (3.18)$$

All that is left to do is observe that v' is bounded as well, specifically $u - 1 \leq v' \leq u$ (as shown in Appendix A). Applying this condition to Eq. (3.18), we can see that:

$$-\frac{1}{M-1} \leq u - \frac{\sum_{i=0}^{M-1} d(i)}{M-1} \leq 0$$

or $\qquad (3.19)$

$$-\frac{1/2}{M-1} \leq u - \frac{\sum_{i=0}^{M-1} d(i) - 1/2}{M-1} \leq \frac{1/2}{M-1}$$

The last equation resembles the boundary conditions for the quantization error of an ideal ADC:

$$-\frac{LSB}{2} \leq e_q \leq \frac{LSB}{2} \qquad (3.20)$$

Equations (3.19)-(3.20) together suggest that the digital output D_{out} should be synthesized using a counter, so that the estimated input $\hat{u}$ will be:

$$\hat{u} = \frac{\sum_{i=0}^{M-1} d(i) - 1/2}{M-1} \qquad (3.21)$$

If this is done, then eq. (3.21) lets us derive a condition on the quantization error $e_q = V_{in} - \hat{V}_{in}$ and the expected resolution:

$$-\frac{V_{ref}}{M-1} \leq e_q \leq \frac{V_{ref}}{M-1} \qquad (3.22)$$

$$LSB = \frac{V_{ref}}{M-1} = \frac{(V_{DAC}^H - V_{DAC}^L)}{M-1} = \frac{FSR}{M-1} \qquad (3.23)$$

$$n_{bits} = \log_2(M-1) \qquad (3.24)$$

An important observation should be made at this point: a consequence of what we have shown is that, in an ISD1, the analogue output of the integrator at cycle $M-1$ is the quantization error itself: Eq (3.18) can in fact be rearranged to give:

$$v'(M-1) = (M-1) \cdot \left(u - \frac{\sum_{i=0}^{M-1} d_i}{M-1} \right) \cong (M-1) \cdot \frac{e_q}{V_{ref}} \qquad (3.25)$$

This interesting fact is sometimes exploited by clever architectures to improve the trade-off between number of steps and resolution; some of these will be presented in Chapter 4.3.

3.3   Non idealities

3.3.1   Limit cycles and dead zones

In traditional Sigma-Delta theory, limit cycles (LC) and dead zones (DZ) are two separate but closely related non-idealities. They are both phenomena arising directly from the $\Sigma\Delta$ non linearity, and are characteristic of $\Sigma\Delta$ with DC or slowly varying signals at the input. Dead zones are regions of the transfer characteristic where the output does not change for a varying input: they therefore degrade a converter's DNL. They can be understood in a time-domain approach, and they appear in the same input regions where LC are observed: in fact, to an extent they can be considered a drastic worsening of limit cycles caused by the finite DC gain of the amplifier used to implement the integrator of the SDM ([19]). LC consist of undesired tones being generated within the DLPF pass-band, and are a critical design issue for distortion-sensitive applications, such as audio. In our case however, since a S&H is used and there is a sample-by-sample correspondence between DC input and output of the ADC, it doesn't make sense to talk about distortion in the frequency domain; for this reason, a strict distinction between LC and DZ (which is necessary for continuously-running ADCs) is unnecessary and possibly confusing for the application of interest, hence the two will be treated together in this paragraph.

LC and DZ are closely related, since they are both associated to the output of the integrator being a periodic waveform for inputs close to rational numbers [12]. However, there is a significant difference: while limit cycles can always be reduced by increasing M (i.e. the oversampling ratio), hence narrowing the DLPF's pass-band, the width of dead zones is independent of oversampling: the only way to get rid of them is to increase the amplifier's gain.

Let's consider a traditional $\Sigma\Delta$, i.e. without S&H or reset, with a constant or slowly varying input: the DC input would require a "linear" ADC to give a constant waveform at the output, whereas the nature of the $\Sigma\Delta$ is so that the output will have some oscillation. In particular, if the input u of a Sigma-Delta is a rational number a/b, the input of the quantizer could be (or will be, in the case of a 1st order converter) periodic (with a duty-cycle proportional to the position of the input within the full scale), and so will be its output. Hence, the output of the digital filter will not be constant, but will have some oscillations: the longer the period of the quantizer input, the lower the frequency of the corresponding tone, hence the least will it be filtered by the DLPF. This can result in a discernible spurious signal at the output, which degrades the SQNR: this problem is known as a limit cycle. In audio applications this effect is reduced by employing high-order architectures or resorting to dither signals being added at the input of the quantizer (to break the periodicity and decrease the temporal correlation of the waveforms) [19].

In an ideal ISD1 limit cycles manifest themselves as a degradation of the ENOB. They occur when the number of cycles M is too low to discriminate between a rational input (which causes a periodic pattern in the quantizer waveform) and an irrational input of similar value (which would cause the same pattern up to a certain cycle n, after which the periodicity is broken). However, the situation is slightly more complex for higher-order ISD, where local increases in DNL can be observed (as will be mentioned in Chapter 4.5.3). Moreover, an

ISD1 whose amplifiers have finite DC gain will be more vulnerable to this problem, eventually leading to the occurrence of dead zones.

Dead zones occur when the integrator has a non-ideal transfer function. The ideal transfer function of an integrator should in fact have a pole $p = 1$, that is:

$$A_{INT}^{id} = g\frac{z^{-1}}{z^{-1} - p} = g\frac{z^{-1}}{z^{-1} - 1} \tag{3.26}$$

However, a finite DC gain of the amplifier G_{DC} will shift this pole closer to 0 – as will be demonstrated in Chapter 5.4 – by a quantity ε_p, i.e.:

$$p = 1 - \varepsilon_p \tag{3.27}$$

The output of the integrator V_{INT} corresponding to DC input thus won't be a ramp; it will be progressively quenched at every cycle, adopting an exponential behaviour. In a ΣΔ, when the input is in the range of a LC, this may cause V_{INT} to never break out of the limit cycle, i.e. to fall into an infinite periodic pattern. This is shown quantitatively in [19] and [12]; here, we just refer to Figure 3.8 to give a visual understanding.
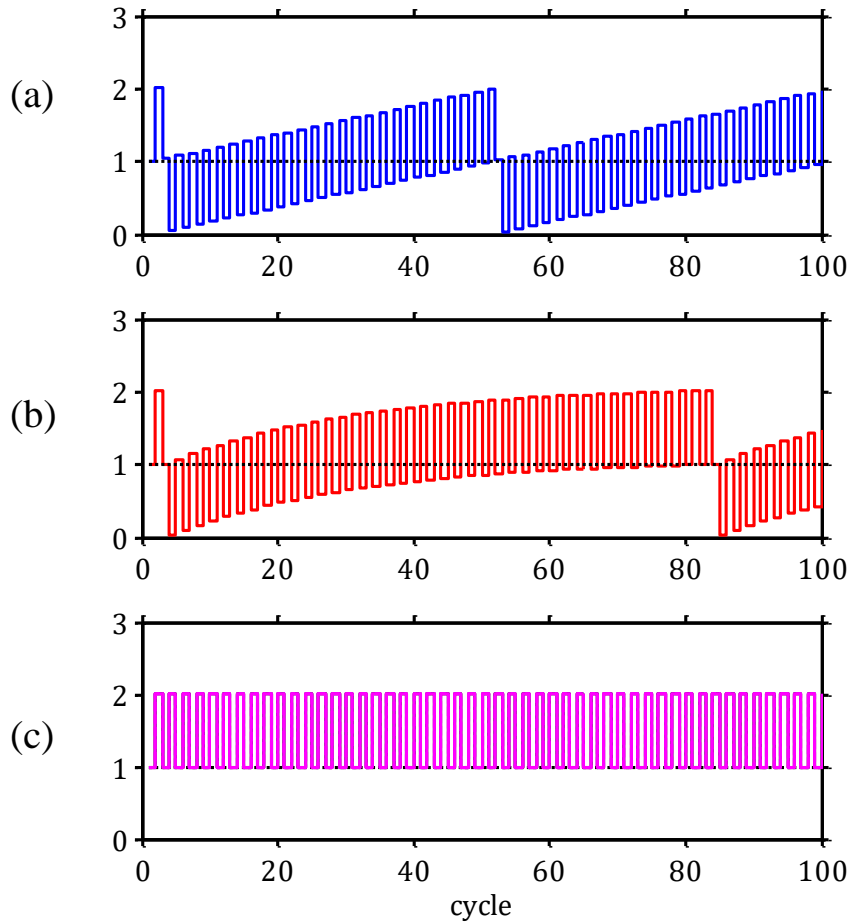
Figure 3.8 Integrator's output voltage. Ideal case (a), low DC gain (b) and very low DC gain (c). In case (c) the integrator is unable to break out of a limit cycle

As a result, there will be a wide input region converted with the same output; for example, in a 1st order Sigma-Delta with input close to the half-range $V_{CM} = (V_{DAC}^H + V_{DAC}^L)/2$, the dead zone range will be:

$$V_{CM} \cdot \left(1 - \frac{\varepsilon_p}{1+p}\right) < V_{in} < V_{CM} \cdot \left(1 + \frac{\varepsilon_p}{1+p}\right) \tag{3.28}$$

note that it is independent of the oversampling ratio, as proven in Figure 3.9.
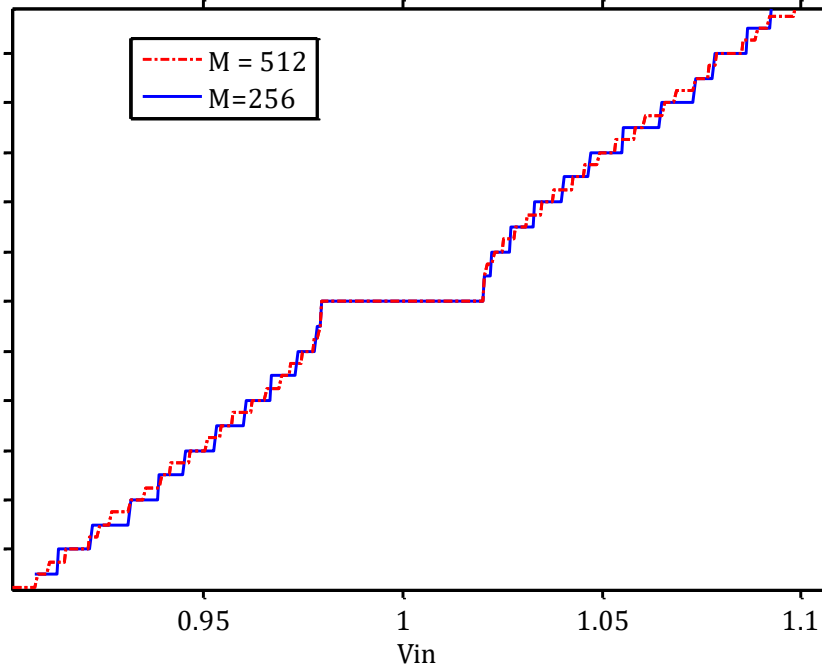
Figure 3.9 Transfer curve showing the independence of dead zones on the oversampling rates. Values on ordinate axes are not shown because the digital output is different for the two curves.

3.3.2 Noise-shaping degradation

Another consequence of finite G_{DC} is the degradation of noise-shaping: the NTF of the modulator will have a non-nil DC gain, thus the spectral power of the quantization error won't be perfectly cancelled in the pass-band of the DLPF. The consequent degradation of SNR can be assessed by considering, for a first-order ΣΔ:

$$|NTF|^2 \sim G_{DC}^{-2} + \left(\frac{2f}{f_s}\right)^2 \tag{3.29}$$

integrated over the interval $[0, f_s/(2OSR)]$ [12].

Figure 3.10 shows a plot obtained through simulations on MATLAB® Simulink® of the quantization noise (extracted as the mean square deviation from the straight line best fitting the transfer-curve) versus G_{DC}. The behaviour for small gain, $\sigma_q^2 = (\text{constant})/G_{DC}$, is consistent with Eq. (3.29). For large gain the mean error approaches $\frac{LSB}{\sqrt{12}}$, as expected.
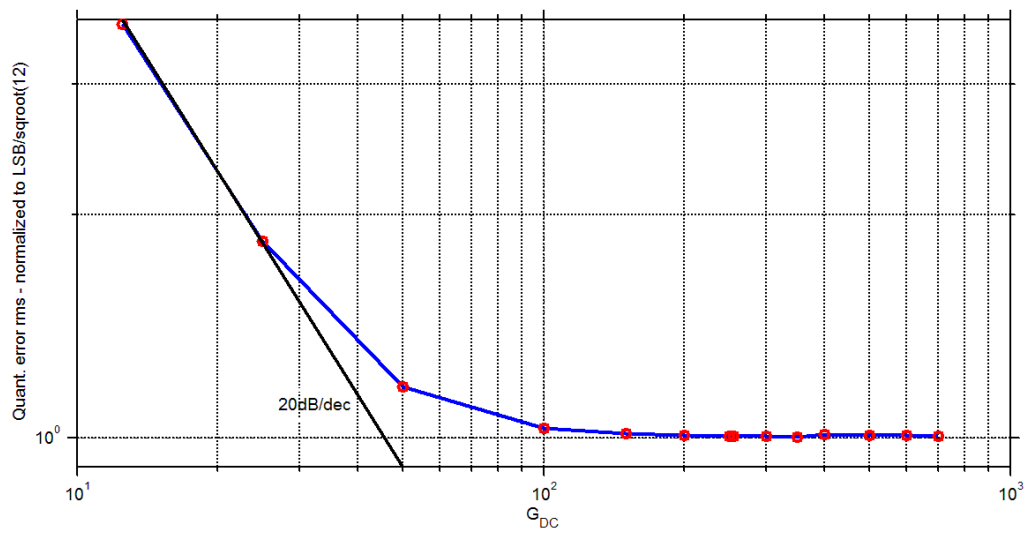
Figure 3.10 Rms of the quantization error noise increase for low OpAmp DC gain

Chapter 4

Architecture design and behavioural simulations

The simple Sigma Delta Modulator (SDM) architecture seen in Chapter 3 is subject to a clear trade-off of resolution (higher OSR) against conversion time T_{ADC} (lower OSR). A first-order Incremental Sigma Delta (ISD) for example, in order to meet our specification of 12 bits conversion, would need to complete – according to Eq. (3.24) – $M = 2^{12} = 4096$ clock cycles, and thus run at $4.1GHz$ to respect the timing constraint of $1\mu s$ set in Chapter **2.4** – a speed of operation which is at the limit of what achievable by logic gates at the technology node used in this project, let alone any analogue circuit.

To overcome this limit it is necessary to resort to a more complex architecture: this chapter deals with the analysis and design of such architecture. We will start with a distinction between time discrete and time continuous SDM, followed by a review of the main architectures that improve the intrinsic trade-off of in the 1st order ΣΔ; a choice regarding the ΣΔ architecture will then be made and, lastly, the specifications for the analogue components of the circuit - derived through system-level behavioural simulations - will be listed.

4.1 Time discrete versus time-continuous converters

The loop in ΣΔ modulators can be implemented either with time-continuous modulators (TCM) or time-discrete modulators (TDM) components, i.e. switched capacitors (SC). Switched capacitor circuits will be analysed in detail in Chapter 5.

TCM have the advantage of having relatively low power requirements compared to TDM (especially for high sample rate) and not requiring an anti-aliasing filter. They also benefit in terms of input noise: in TDM, in fact, noise is aliased and sampled on the capacitor, which (as will be explained thoroughly in Chapter 5.7) needs to be made large enough to have a sufficiently low input white noise. However, they suffer from relatively high absolute spread in position of poles and zeroes. Moreover, timing in TCM is of major importance: for example, a delay in the instant at which the DAC output starts being fed back at the loop input directly affects the charge accumulated at the integrator's output, and consequently the transfer function - a delay term has to be introduced. Both deterministic delay and jitter are therefore an issue. All of these problems ultimately constitute uncertainty in the poles of the

sampled transfer function (i.e. assessed with Z-transform), which leads to large spread and lower reliability ([3],[11]). The considerations just made are summarised in Table 4.1.

For the reasons above TCM are hardly ever used in systems with a large number of ADCs, despite TDM being generally more power hungry, and the ΣΔ designed in this work was in fact chosen to be of TDM type.

Table 4.1 Comparison between time discrete and time continuous modulators

	Time discrete	**Time continuous**
Anti-aliasing	Pre-filtering needed	Can be intrinsic in ΣΔ loop (closed-loop pole must be $< f_s/2$)
Relative position of singularities	Defined by capacitor ratios	Matching of capacitor ratios and ratios between resistors
Absolute position of singularities	Capacitor ratios and sample rate →Well controlled	Filter curve modulated by absolute spread of parameters
Op-amp constraints	Small settling error High DC gain	Good linearity
Power requirement	Fast sampling (hence relatively high power)	Linearity and noise of first integrator (variable)
Impact of jitter and loop delay	Nil if complete settling occurs	Relevant since it directly affects the transfer function
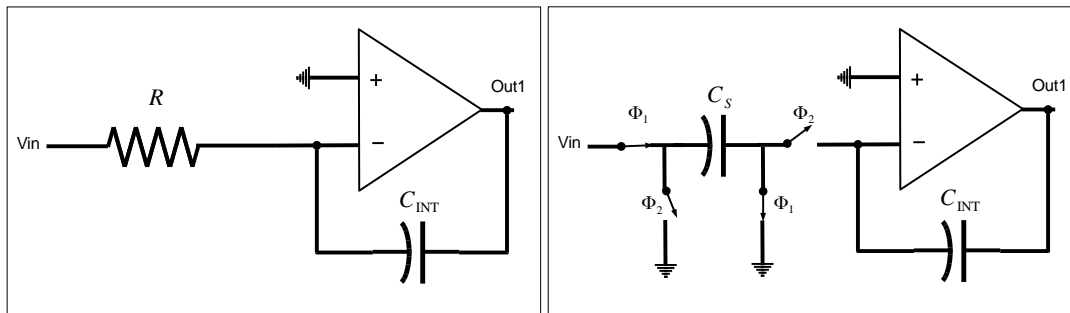
Figure 4.1 Blocks in a Sigma-Delta modulator can be implemented with either time-continuous (left) and time-discrete (right) integrators.

4.2   Composite structures versus high-order architectures

ΣΔ ADCs achieving resolution higher than the 1st order can be divided into two groups: composite structures and higher order structures. Composite ΣΔ ADCS can be implemented using only a first-order modulator or variations of it and still obtain a better $n_{bits}\ vs\ M$ relation than a first-order loop. High-order architectures, instead, respect the classic structure of the 1st order ΣΔ, but employ filtering of order higher than 1 in both the forward path of the modulator and in the decimator. The following sections briefly review some of the solutions in these categories that were considered when making system-level design choices for the Sigma Delta: in particular, since the SDM was chosen to be of time-discrete type and the

ADC will be reset between one conversion and the next one, the candidate architectures were either Incremental Sigma Delta (ISD) or variations of it.

Before starting this process, it is worth remembering that the crucial requirements for an ADC in a column parallel or stacked chip image sensor are (as explained in Chapter 2.3):

- small area (hence relative simplicity in its structure is desirable)
- low spread in parameters (hence low susceptibility to device mismatch)
- low input noise.

Compared to high order architectures, ADCs adopting composite structures tend to be more area and power efficient; another interesting characteristic is that, avoiding the implementation of high-order feedback loops, they also avoid the related stability problems. However, unlike high-order architectures, they fail to overcome some of the limitations of the 1st order $\Sigma\Delta$ - such as the sensitivity of the transfer characteristic to the analogue amplifiers' high gain - which eventually lead to increased spread in the ADCs' overall behaviour.

4.3 Composite structures

4.3.1 MASH

Multi-stAge noise-SHaping (MASH) modulators (see Figure 4.2) employ two 1st order $\Sigma\Delta$ modulators: the input to the first one is the analogue value to be converted, while the input to the second one is the quantization error E_1 itself, obtained by analogue subtraction of the input of the quantizer from the output of the DAC [12]. The outputs of the two ADCs are then processed by the filters $H_1(z)$ and $H_2(z)$, then subtracted.
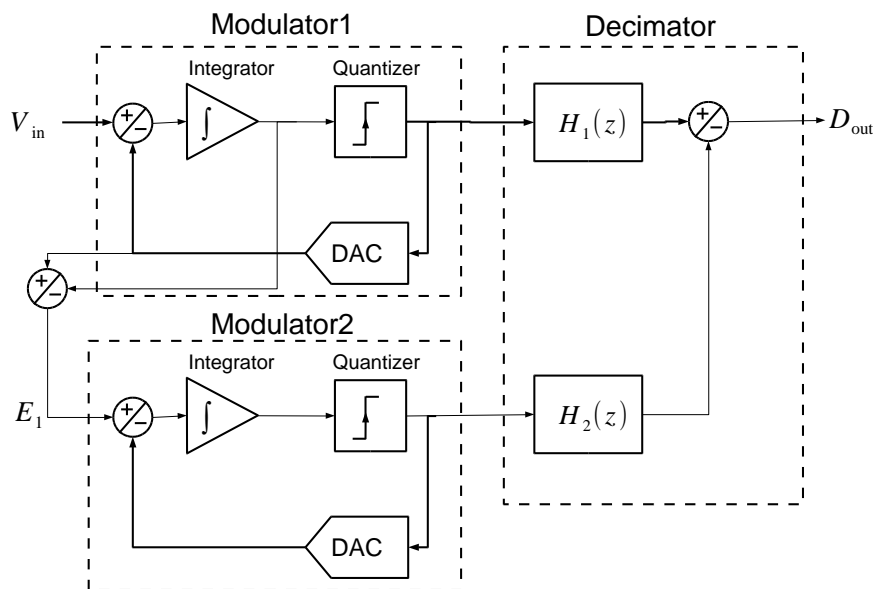
Figure 4.2 Block diagram of a MASH modulator

A good description of how it works is given in [12]. The main idea is to "tune" the transfer functions so that E_1 is cancelled at the output: the quantization error left on the digital output will hence be that of a 2nd order system, but with the stability characteristics of a 1st order ΣΔ. However, this architecture has the drawback of intrinsically relying on the "tuning" accuracy between two analogue transfer functions, thus being sensitive to spread and mismatch: it is likely that a chip employing thousands of these ADCs would manifest a wide performance range from one ADC to another. For this reason, this solution was discarded.

4.3.2   Two-step conversion

One advantage of working with a DC input is that, as shown in Chapter 3.2.1, the analogue output of the integrator at cycle $(M - 1)$ is the quantization error itself: hence, an operation similar to that of a MASH ADC can be carried out using one modulator only, without performing an analogue subtraction between input and DAC-converted value. This is the idea behind the two-step conversion [20]: instead of continuously performing a conversion of both the input and the error, the conversion is split into two phases, always using the same modulator: first, the modulator operates with V_{in} at its input for M_1 cycles, thus obtaining the $\log_2(M_1)$ most significant bits (MSB) of conversion; then, the residual value on the integrator is sample-and-held and converted by the modulator for M_2 cycles, thus gaining the last LSBs. The resolution is thus:

$$n_{bits} = \log_2(M_1) + \log_2(M_2) = \log_2(M_1 M_2) \tag{4.1}$$

Comparing Eq. (3.24) with (4.1) we can see that a conventional, 1st order ISD would need $M_1 \cdot M_2$ cycles to obtain this resolution.

This architecture has been investigated and tested in image sensors (in [21], for example, a proof-of-concept with an 8x8 imager is presented), making it a good candidate for this project.

4.3.3   Extended counting

Extended counting ΣΔ ADCs were first proposed by Jansson for column-parallel CMOS image sensors in [22]. The working principle will now be briefly explained; for more detailed and complete analysis, one can consult [22] and [23].

The conversion is operated in two phases: the first phase, called "counting" phase, lasts M_1 cycles and is the same as a normal conversion performed by a 1st order ISD: it employs a SC integrator, a comparator and a counter. The only exception is that during Φ_1 of the last cycle of this phase, the sampling capacitor C_S is reset instead of sampling V_{in}: the final output voltage of the integrator will then be:

$$V_{count} = V_{INT}(M_1) = M_1 \frac{C_S}{C_{INT}} V_{in} - V_{ref} \frac{C_S}{C_{INT}} \sum_{i=1}^{M_1} D(i) \tag{4.2}$$

During the second phase, the residual voltage V_{count} is converted using a "more efficient but less accurate algorithmic A/D conversion technique" ([23]): this is performed using the same hardware as the first phase, i.e. integrator, comparator and counter, except that two

additional capacitors of value C_S are used. The circuit is arranged in such a way that the equation describing the output voltage variation becomes:

$$V_{INT}(j) = \left(1 + \frac{C_S}{C_S}\right) V_{INT}(j-1) - D(j-1)\frac{C_S}{C_S}V_{ref}$$
$$\cong 2V_{INT}(j-1) - D(j-1)V_{ref} \tag{4.3}$$

After M_2 cycles, provided V_{count} was sampled on the extended counting capacitors at the start of the phase, we have:

$$V_{count} = V_{ref} \sum_{j=1}^{M_2} 2^{-j} D(M_1 + j) = V_{ref}D_{count} \tag{4.4}$$

The quantity D_{count} represents a conversion of V_{count} of M_2 bits. The input can therefore be recovered combining equations (4.3)-(4.4):

$$V_{in} = V_{ref}\frac{\sum_{i=1}^{M_1} D(i) + D_{count}\frac{C_{INT}}{C_S}}{M_1} \tag{4.5}$$

The first term, given by the first-order ISD conversion, has a resolution $n_1 = \log_2 M_1$, and represents the n_1 MSBs of the digitalized input; the second term has a resolution $n_2 = M_2$, and represents the n_2 LSBs. It is remarkable that after the first slow, ΣΔ-like conversion, every cycle of the second phase adds one bit of resolution! Another important observation is that non-idealities in the extended counting phase (such as capacitors' mismatches) cause an error in D_{count} whose impact on the output, according to Eq. (4.5), is reduced by a factor M_1. However, this phase still suffers from the comparator's offset, noise and hysteresis (to which the counting, ΣΔ-like phase is instead virtually immune), which can dramatically reduce the ADC linearity; to resolve this issue, [23] proposes the use of two comparators with different thresholds.

4.4 Higher order architectures

4.4.1 Noise shaping and resolution

In Section 3.1.3 we saw how the loop in a 1st order Sigma-Delta performs noise-shaping. The goal is differentiating the quantization error while preserving unaltered the information carried by the signal, so that the in-band power of this noise S_Q is reduced below what achievable by oversampling alone. It is natural to see, then, that noise-shaping can be brought to the "next level" by increasing the order of differentiation, so that S_Q is further "squeezed" to high frequencies. This can be done in several ways: we will refer to the 2nd order ΣΔ in Figure 4.3 to derive the expected performance of such architecture.
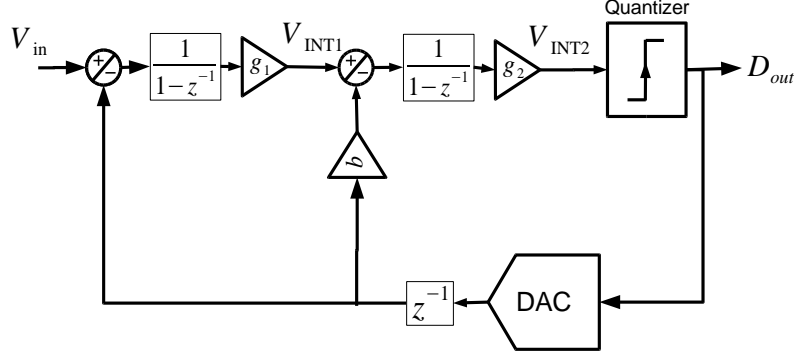
Figure 4.3 Example of a 2nd order SDM

The enhanced noise-shaping in a lth order SDM gives – for some setting of the coefficients - a noise-transfer-function of the form:

$$NTF = (1 - z^{-1})^l \tag{4.6}$$

Proceeding as done for a 1st order architecture in Eqs. (3.10)-(3.11), the resolution of an ideal decimator would thus be:

$$n_{bits}(l) = (l + 0.5) \log_2 OSR - \log_2(\pi) * l - 0.5 \log_2(2l + 1) \tag{4.7}$$

Which for $l = 2$ becomes:

$$n_{bits}(2) = \frac{5}{2} \log_2 OSR - 2.14 \tag{4.8}$$
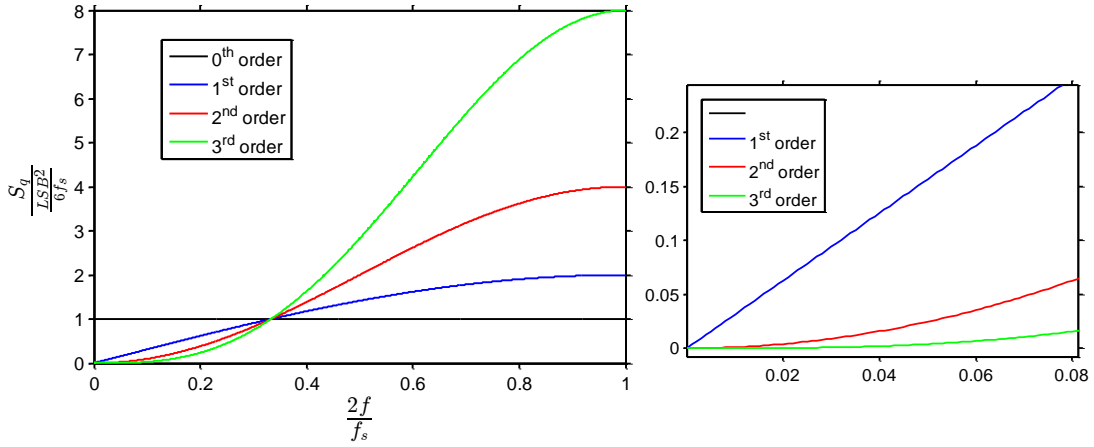
Figure 4.4 NTF for different values of the order *l* (left) and detail around *f=0* (right), where the pass-band of the DLPF is.

Incremental Sigma-Delta

As already stated in Section 4.2, the ADC will be an ISD, which will have a worse performance in terms of ENOB compared to a traditional ΣΔ. The resolution of a 2nd order ISD (ISD2) can be found with an argument not too different from that one given for ISD1: here we will not carry out all the passages (if interested, the reader can consult [13] [15] [16][17]), but simply report and comment the results. The diagram in Figure 4.3 can be held

as reference. The focus is on the output of the last integrator V_{INT2} and, once again, the key passage is assuming that it is bounded within a range $\Delta_{min}^{max} V_{INT2}$:

$$\Delta_{min}^{max} V_{INT2} = V_{INT2}^{max} - V_{INT2}^{min} = g_1 g_2 \cdot (V_{DAC}^H - V_{DAC}^L) \tag{4.9}$$

The resolution found with this approach is:

$$2^{n_{bits}} = \frac{V_{DAC}^H - V_{DAC}^L}{\Delta_{min}^{max} V_{INT2}} \cdot \frac{M(M-1)}{2} \cdot g_1 g_2 = \frac{M(M-1)}{2} \tag{4.10}$$

Therefore, two bits are gained at every doubling of M.

The decimator is here assumed to be a cascade of integrators, since architectural complexity is preferably avoided in column-parallel image sensors. Other filters could give a slightly enhanced resolution, as shown in [16].

Note that the equivalence proposed in Eq. (4.9) differs from what found in references [13] [15][16][17], where $\Delta_{min}^{max} V_{INT2}$ is not considered proportional to $g_1 g_2$, and the two integrators' coefficients thus remain present in Eq. (4.10) for the ENOB. However, this work considers valid the relation in Eq. (4.9), since no severe dependence of the LSB on g_1 and g_2 was observed in simulations (see Figure 4.5).
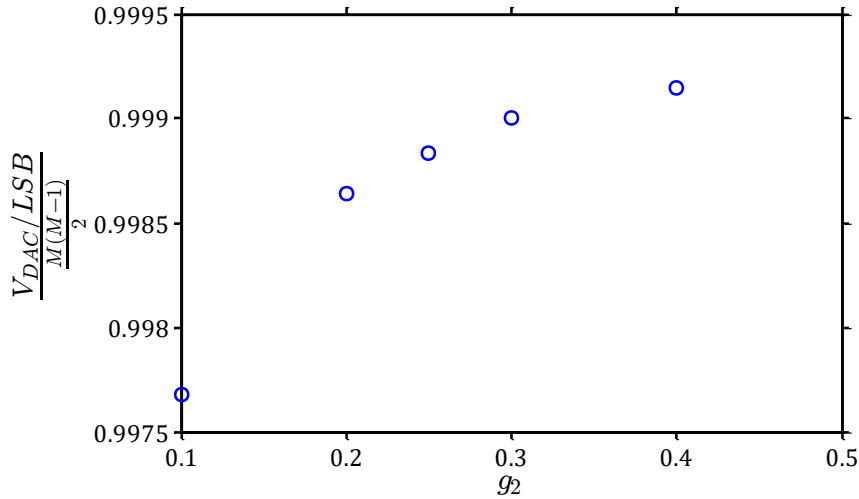
Figure 4.5 Dependence of the LSB on the gain of the second integrator g_2. On the ordinate is the ratio between the extracted LSB and that estimated with Eq. (4.10).

Resolution comparison
Following this discussion, Table 4.2 compares the resolution achievable from an ideal $\Sigma\Delta$ and an ISD as a function of the oversampling ratio (OSR or M) and the order l.

Table 4.2 Maximum expected ENOB as a function of the oversampling ratio for different digital filters

$ENOB_{(OSR)}$	Sharp cut-off digital LPF	ISD
1st order	$\frac{3}{2}\log_2 OSR - 0.86$	$\log_2 M$
2nd order	$\frac{5}{2}\log_2 OSR - 2.14$	$\log_2 \frac{M(M+1)}{2}$ $\sim 2\log_2(M) - 1$
lth order	$(l+0.5)\log_2 OSR - \log_2(\pi)*l - 0.5\log_2(2l+1)$	$\log_2\binom{M}{l}$ $\sim l\cdot\log_2(M) - \log_2(l!)$

4.4.2   Advantages over first-order composite structures

Robustness against unreliable analogue components

Another important advantage of increasing the order of a ΣΔ is that looser specifications can be set regarding the precision, noise and reliability of the constituting analogue components, provided that stability can always be guaranteed. This can qualitatively be understood by once again considering the linear model and supposing that noise is added at every summing node of the modulator, referring to Figure 4.6.
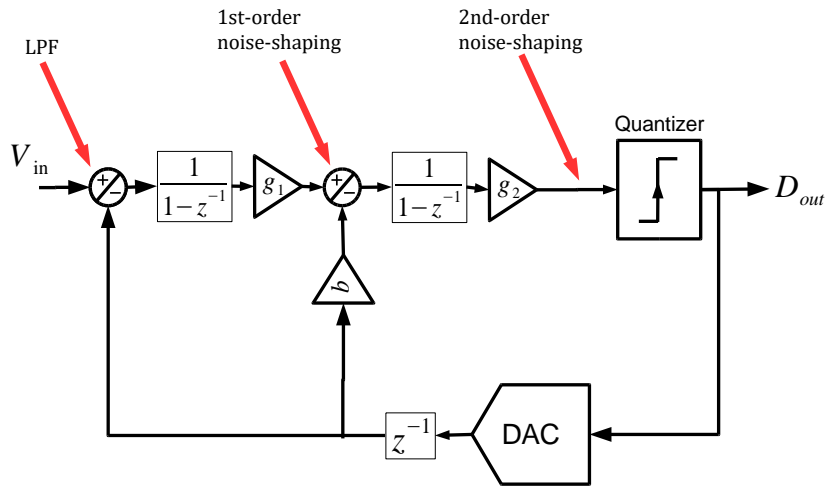
Figure 4.6 Noise filtering at different SDM nodes

Considering for simplicity $g_1 = g_2 = b = 1$, the transfer functions of the inputs from left to right are:

$$STF = z^{-1} \tag{4.11}$$

$$NTF1 = (1 - z^{-1}) \tag{4.12}$$

$$NTF2 = (1 - z^{-1})^2 \tag{4.13}$$

We can see, therefore, that noise shaping does not only occur for the quantization error at the rightmost node, but it can also affect other signals entering the ADC at different points, regardless of their physical origin (offset, coupling, white and $1/f$ noise, etc.).

This implies a very important result that can easily be generalized. In an nth order Sigma-Delta modulator (SDM), which has $(n+1)$ nodes, any input added at the kth node undergoes a noise shaping of order $(k-1)$ (counting from left to right). At the input of the ADC (where $k=1$) there will be no noise shaping, but the input will be affected by the low-pass filtering of 0th order modulators, as already explained in Section 3.1.4. The impact of the noise (or the offset due to mismatch) of stages further to the right is progressively eliminated by increasingly efficient noise shaping, *without the need to amplify the signal as it moves forward in the chain*. In fact, the choice of the integrators' gain is never set by noise specification but, instead, by stability and linearity requirements.

A very important consequence for image sensors is that the ADC is also resilient to the spread of the components of stages other than the first (unless they directly influence its stability, such as the gains of the integrators) as was already mentioned in Chapter 2.3.

Limit cycles and dead zones
High-order SDMs have been found to be less vulnerable to limit cycles and dead zones, thanks to a steeper cut-off of the DLPF regarding the former and to a decreased autocorrelation of the input of the quantizer for the latter [12, 11]. In [12] the dead zone span dependence on G_{DC} is estimated after simulations to be $\Delta/FSR \sim 3/(4G_{DC}^2)$

4.4.3 Disadvantages

Overload of the quantizer
In Section 3.1.4 it was shown that the input of a 1st order $\Sigma\Delta$ needs to be bounded between the DAC reference voltages at all times. If this condition is broken, the SDM is not able to effectively apply the feedback to stabilize the integrator's output. This situation is called overloading, and in converters of higher order it gets worse: the input will need to be confined within a range smaller than the FSR of the DAC, otherwise the resulting transfer characteristic will be highly irregular, as Figure 4.7 shows. For a 2nd order, the safety margin is roughly $0.8(V_{DAC}^H - V_{DAC}^L)$, where $\alpha_{OL} = 0.8$ is the so-called overloading level [3].
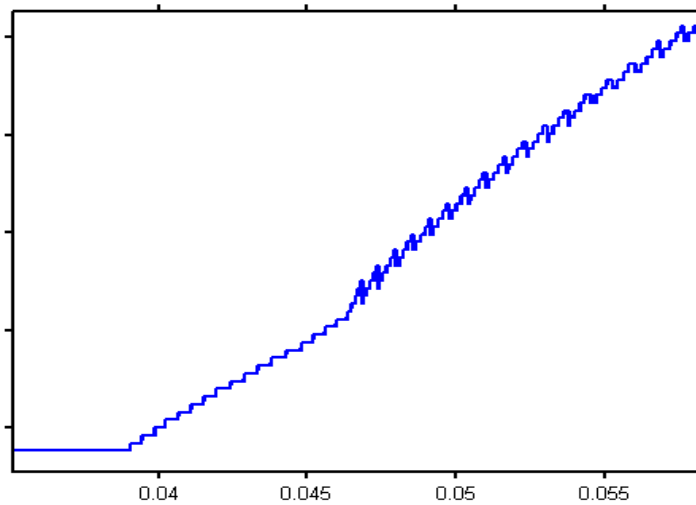
Figure 4.7 Non monotonic transfer curve for input close to the bottom of the FSR: effect of quantizer overloading

When V_{in} is close to the overloading range, the input of the quantizer will have a long autocorrelation (see Figure 4.8), which breaks the validity of the noise-shaping argument. For this reason, even for inputs within the acceptable range, the quantization error, INL and DNL measured on the static trans-characteristic will be higher as the input approaches the edges of the full scale.

As a consequence of overloading, the number of cycles M for an ISD2 must always be set higher than the minimum computed with Eq. (4.10), so that the desired $2^{n_{bits}}$ steps of the trans-characteristic will occur in a range narrower than the full scale set by the DAC. The ratio between effective input range and the full scale is given by:

$$\frac{\Delta V_{in}}{V_{DAC}^H - V_{DAC}^L} = \frac{M^{min} \cdot (M^{min} - 1)}{M \cdot (M - 1)} \cong \left(\frac{M^{min}}{M}\right)^2 \qquad \textbf{(4.14)}$$
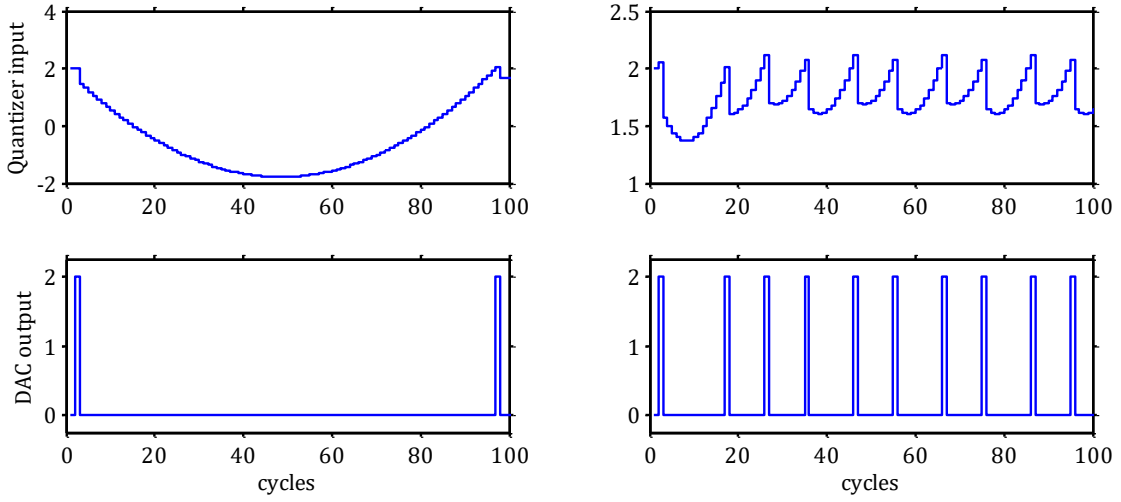
Figure 4.8 Quantizer input and output waveforms when it is overloaded (left, V_{in} at 96% of FSR) and when it's not (right, V_{in} at 80% of FSR) . Note the clearly lower autocorrelation of the waveform to the right compared to that on the left.

Stability

As is well known, ensuring the stability of any feedback system becomes more and more problematic as the order of the loop of the filter increases. SDM loops are no exception to this rule, and they have the added complication of non-linearity, which makes hand-written stability analysis difficult. To exemplify this, consider the loop shown in Figure 4.9: the loop gain G_{loop} derived with a linear analysis will be clearly proportional to coefficient g_1; a stability analysis carried out with the linear model would hence suggest, according to the Bode criteria ([24],[25]), that g_1 needs to be bounded so as to not excessively increase the DC gain of the loop and consequently lead it to instability. On the contrary, coefficient g_1 has absolutely no effect on the loop, since it simply scales the magnitude of the signals fed to the comparator, whose output (ideally) only depends on the sign of its input relative to the threshold! This was already noted and expressed in formulas in Section 3.2.1, Eq. (3.18) for ISD1, where the situation was similar.
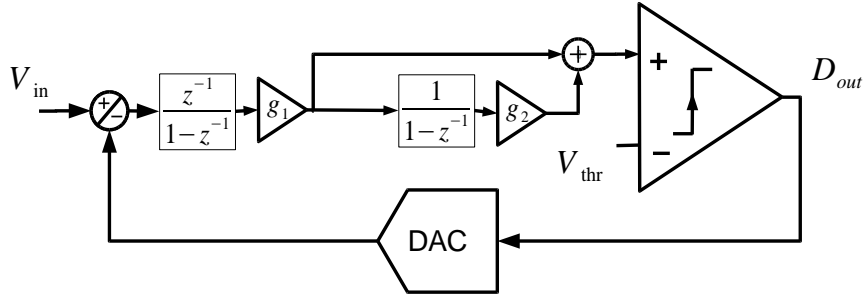
Figure 4.9 2nd order SDM which is stable regardless of coefficient *g1*

Having acknowledged that the linear model is inapplicable to this purpose, other methods have been devised; here, we briefly mention two of them:

- *Analysis of the state equations* (for 1-bit quantizers) [11]: the outputs of the n stages in a nth order modulator are expressed as a function of their previous states and the other nodes' values, and the trajectories in an nth dimensional state space are derived. This is done considering the DAC output an independent input of the ADC, having either the *constant* value V_{DAC}^{H} or V_{DAC}^{L}. The trajectories can be studied analytically to find the values of the input which make the state variables bounded.

- *Describing function method* [26]: two separate loops are considered, one to model the *STF* and another one to model the *NTF*. In each loop, the quantizer is modelled by a non-linear gain K; hence, there will be two quantizer gains, a "signal gain" and a "noise gain", both obtained by minimizing a mean-square-error criterion. By assuming the noise has a certain probability-density function, the internal signals and noise can be characterized statistically, and hence the SNR estimated as a function of the input.

Both methods still require extensive and complex computer analysis and hand calculations; in this work, it was preferred to perform simulations sweeping the coefficients and deriving a partial I/O quantization curve, so that also INL, DNL and the eventual appearance of glitches (see Figure 4.15) could be measured and studied.

4.4.4   Typical architectures

There are two main ways of increasing the order of noise shaping, which use cascaded integrators: Cascaded Integrator Feed Back (CIFB – see Figure 4.3) and Cascaded Integrator Feed Forward (CIFF – see Figure 4.9).

In CIFB the DAC output is fed to the input of every integrator in the loop. In CIFF, instead, the DAC output is only fed to the input of the first stage, and the internal signals are added at the input of the quantizer

Between the two options, CIFF was preferred to CIFB because it maintains the number of DACs to one per modulator, thus avoiding the issue of large voltage swing at the input of amplifier of further stages, hence decreasing their power demand [27],[13].

4.5   Implemented architecture

4.5.1   Order, oversampling ratio and input range

The ADC was chosen to be a second order Incremental Sigma Delta. Composite structures are interesting for the simplicity of their building blocks and for power consumption reasons (since they can employ the same integrator in different phases to achieve high bit-depth [23]), but were discarded for two reasons. In the first place, they require more careful design to limit the spread and amplifiers with higher gains to contain the dead zones. Secondly, they implicitly require additional capacitors for S&H of signals other than the input (the input of the quantizer after M_1 cycles for both two-step conversion and extended counting architectures), which is a delicate operation in fast circuits, especially on a mixed-signal chip, and simulations can only partially help in evaluating the effect of charge leakage and injection in a real circuit. On the other hand, $\Sigma\Delta$ of order higher than 2 were excluded for the related issues of power consumption and stability.

Therefore, as mentioned in Section 4.4.1, the 2nd order DLPF will be composed of a cascade of integrators, the first of which will be a $\log_2(M)$ bits counter (in case all M cycles give a '1') and the second of which needs to be a $\log_2[M(M-1)/2]$ bits accumulator. Reversing Eq. (4.10), the minimum number of cycles that theoretically ensures a 12-bit conversion is thus found to be:

$$M^{min} \cong 2^{\frac{n_{bits}+1}{2}} \cong 91 \; cycles \qquad\qquad \textbf{(4.15)}$$

In order for the actual input range to reside within the overloading limits of the ADC, the actual number of cycles was hence set using Eq. (4.14) to:

$$M = \frac{M^{min}}{\sqrt{\alpha_{OL}}} \cong 100 \; cycles \qquad\qquad \textbf{(4.16)}$$

This set the oversampling frequency of the ADC:

$$f_S = \frac{M}{T_{ADC}} = 100MHz \qquad\qquad \textbf{(4.17)}$$

Given that the ENOB is set only by M and hence no changes in the architecture of the modulator are required to obtain a higher number of bits, for testing purposes it was decided that the ADC developed in this project should be able to give a resolution of up to 16 bits (achievable with $M\sim400$), hence the real number of bits in the DLPF would need to be 9 and 17, for the 1st and 2nd stage respectively.

The values of the DAC high and low voltages were chosen to fit the typical output range of a pixel: we have $V_{DAC}^{L} = 0$ and $V_{DAC}^{H} = 2$; the corresponding input range goes from $V_{in}^{min} = 0.171V$ to $V_{in}^{max} = 1.826V$.

Table 4.3 Summary of ADC characteristics of operation

M	12 *bits conversion*: $M^{(nominal)} = 100\ cycles$ 16 *bits conversion*: $M^{(16)} = 400\ cycles$
Conversion time T_{ADC}	$T_{ADC} = 1\mu s$
Sampling frequency f_S	$fs = \dfrac{100}{T_{ADC}} = 100MHz$
$FSR = V_{DAC}^H - V_{DAC}^L$	$2V - 0V = 2V$
Input range	$\Delta V_{in} = 1.655V \sim 83\%$ of FSR $V_{in}^{min} = 0.171V$; $V_{in}^{max} = 1.826V$
LSB	12 *bits conversion*: $LSB = 404\mu V$ 16 *bits conversion*: $LSB = 25.06\mu V$
Decimator output	17 bits

4.5.2 InFF versus DiFF

Two architectures of ISD2 were considered, both of the CI Feed Forward type; their block diagrams and equivalent circuits are shown in Figure 4.10 and Figure 4.11. To distinguish between the two, the first one will be called Input Feed Forward (InFF), and the second one Difference Feed Forward (DiFF) – since, as will be explained, the feed forward is obtained at schematic level as the difference between two signals.

In the InFF (also known as Silva-Steensgaard's architecture), shown in Figure 4.10, the input and the two outputs of the integrators are all added at the positive input pin of the comparator; the summing node is implemented exploiting charge sharing between three capacitors. This is one of the most commonly encountered topologies in other works ([16][13],[28]), since it claims to be particularly resistant to the amplifiers' non-linearity and limited gain. With the coefficients shown in Figure 4.10, in fact, a linear-model analysis shows that the forward chain in the loop only processes the quantization error, thus its non-idealities (in principle) won't affect the signal transfer [28].

In the DiFF, shown in Figure 4.11, the input feed forward branch is removed: only the two output signals of the integrators are now added. This allows us to make a modification to the conventional design at schematic level: since there are now only two signals determining the output of the quantizer, one can be connected to the positive pin of the comparator, while the other can be first inverted and then connected to its negative pin. As a consequence, the three summing capacitors of InFF can be removed, thus allowing a reduction of the modulator's area occupation.

The DiFF, compared to the InFF, has the following advantages:

- Simpler, more compact structure
- More freedom in choice of coefficients: while g_2 directly influences the modulator's stability, g_1 can in principle assume any value, as noted in Section 4.4.1.

Nevertheless, it has drawbacks:

- OTAs non-idealities have a slightly larger impact on conversion linearity (as shown in the following section)
- The negative input of the comparator is not connected to a static threshold anymore, hence it must bear a larger input range; however, it is worth remembering that the comparator's precision is not critical to the design;
- During ϕ_2, the two integrators are connected in series, hence the dynamic performance will be that of a two-pole system with similar time constants, which has a larger settling time. However, as long as the integrators behave linearly, settling errors don't significantly affect the ADC conversion [11].
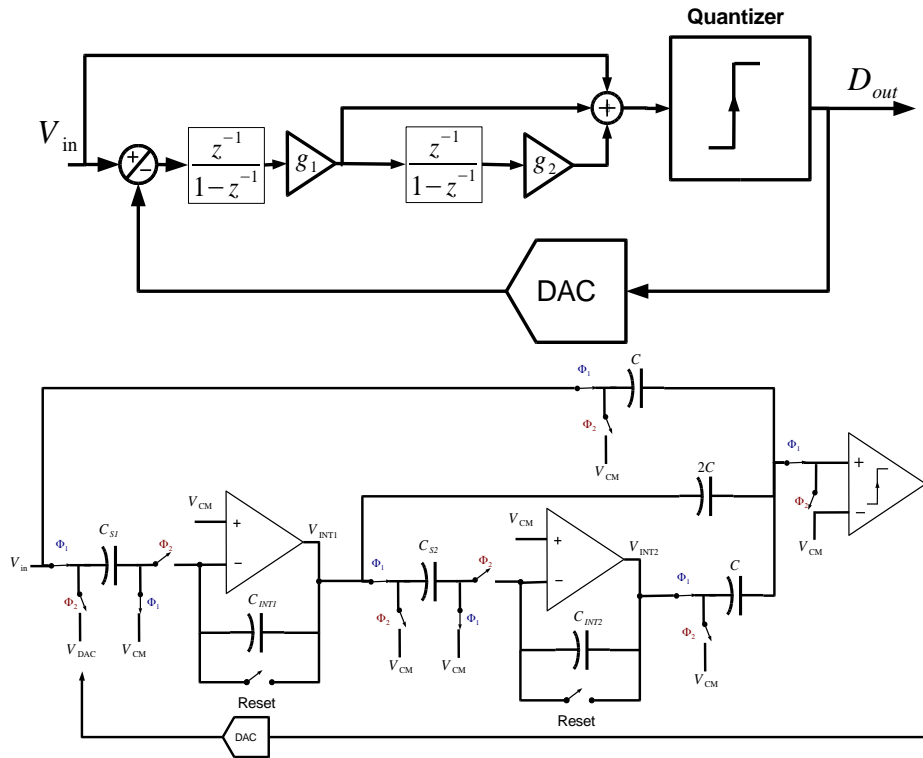
Figure 4.10 Silva-Steensgard feed forward configuration (InFF). Block diagram (top) and circuit schematic (bottom)
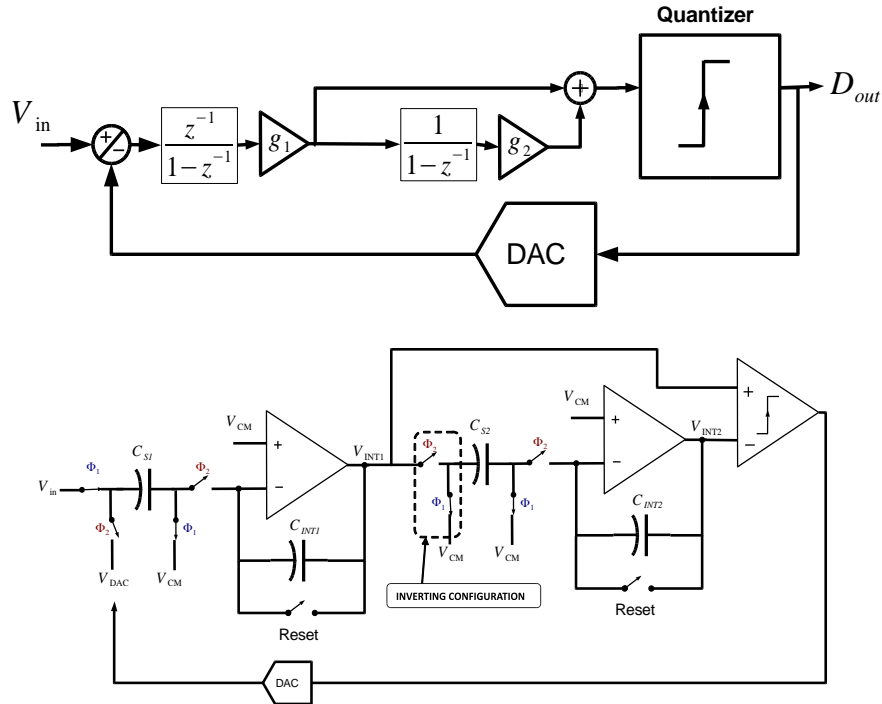
Figure 4.11 Simplified feed forward configuration (DiFF). Block diagram (top) and schematic (bottom). The input feed forward branch has been removed to allow the elimination of the summing capacitors

4.5.3   Comparison through behavioural simulations

Prior to schematic level simulation and design, behavioural simulations were carried out using MATLAB® Simulink® which, although less precise than an IC simulator, is fast and allows for easier storage and elaboration of the results.
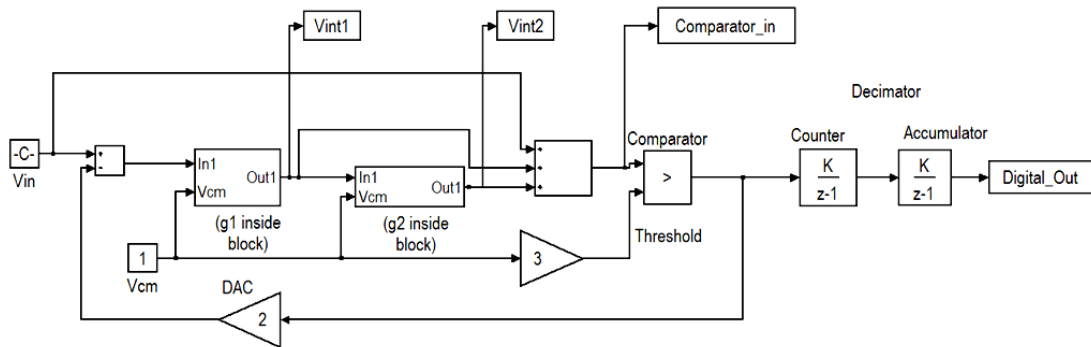
Figure 4.12 Behavioural simulations architecture block diagram - Input Feed Forward configuration

Using this tool, the I/O transfer-curves of the two candidate ADCs for the design were extracted and the effect of non-idealities was estimated.

In the results reported here the coefficients g_1 and g_2 were set to give the best conversion, and were found through extensive simulations. See Table 4.4 below. For DiFF g_2 was chosen equal to 0.25 to give the best compromise between DNL, INL and the number of glitches in the transfer characteristic, as shown in Figure 4.14.

65

Table 4.4 Best coefficients for DiFF and InFF architectures

	DiFF	InFF
g_1	(irrelevant)	0.5
g_2	0.25	0.25

The differential and integral non-linearities were extracted as described in Chapter 1. For completeness, although not explicitly in the specification, the effective number of bits (ENOB) is also reported, measured as the number of bits for which the measured rms of the quantization error equals that of an ideal ADC, as seen in Eq. (2.2).

The comparison between the two architectures is summarised in Table 4.5: the two performances are similar overall, but the DNL of the Silva-Steensgaard architecture is remarkably better. This suggests a higher vulnerability of the DiFF architecture to limit cycles and dead zones, which was confirmed by including in the simulations the impact of the finite DC gain of the integrators' amplifiers. The negative effect of poor DC gain was introduced by modifying the transfer functions of the integrators to have a pole $p < 1$, as in Eq. (3.26); the corresponding DC gain can be calculated using the relation derived in Chapter 5.5:

$$p = \frac{1}{1 + \frac{g}{1 + G_{DC}}} \xrightarrow{G_{DC} \gg 1} 1 - \frac{g}{G_{DC}} \tag{4.18}$$

$$\varepsilon_p = \frac{g}{1 + G_{DC} + g} \sim \frac{g}{G_{DC}} \tag{4.19}$$

Looking at Figure 4.13 we can see that the InFF structure is very robust against integrators with low DC gain; DiFF on the other hand is more sensitive to this parameter and its DNL at relatively low values of G_{DC} is dominated by dead zones (appearing at precisely 1/3 and at 2/3 of the FSR).
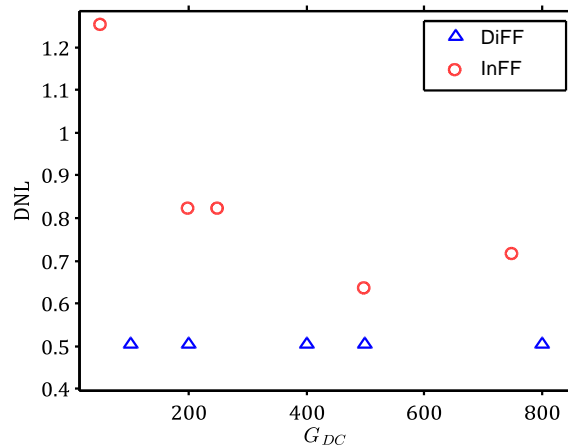
Figure 4.13 Maximum DNL (in bits) of DiFF and InFF architectures as a function of the amplifiers' DC gain

Table 4.5 Comparison between the two considered architectures of Sigma-Delta.

	DiFF	InFF
DNL (bits)	0.68	0.48
INL (bits)	2.13	2.39
ENOB	11.63	11.68

Despite InFF being insensitive to the amplifiers' DC gain, DiFF was chosen as the definitive architecture for the ADC in an attempt to save area. Moreover, acceptable DNL (DNL < 1 according to the specification set in Chapter 2.4) can be obtained with $G_{DC} > 200 = 46$dB, a value very easily achievable in amplifiers – especially if cascode configurations are used.
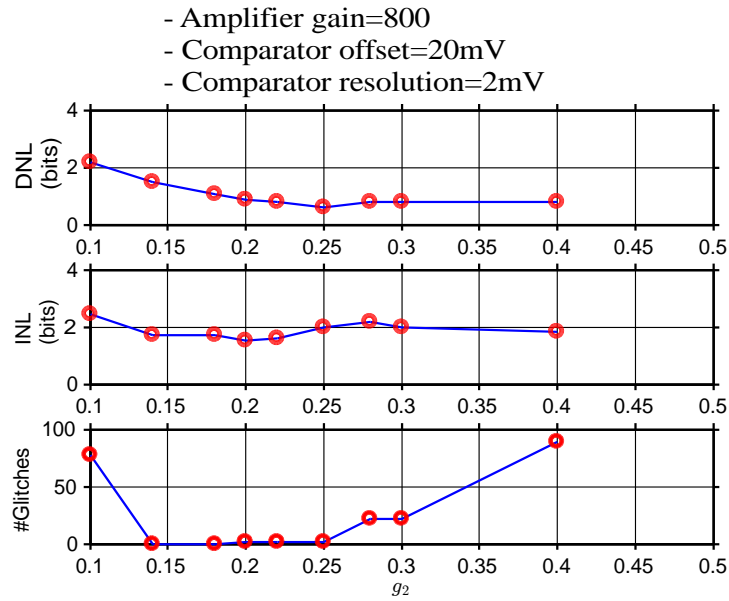
Figure 4.14 DNL, INL and total number of glitches in the transfer curve as function of g2. Results obtained simulating a non-ideal SDM, with finite DC gain of OTAs and finite offset and resolution of the comparator

4.6   Deriving analogue specifications

4.6.1   OTA gain

Based on the results shown in Figure 4.13, it was decided to set the minimum value of G_{DC} at 200, so as to ensure a good DNL for g_2 equal to the nominal value; moreover, it was simulated that DNL < 1 is thus ensured even for $g_2 = 0.2$, which is a drastic deviation from the nominal value.
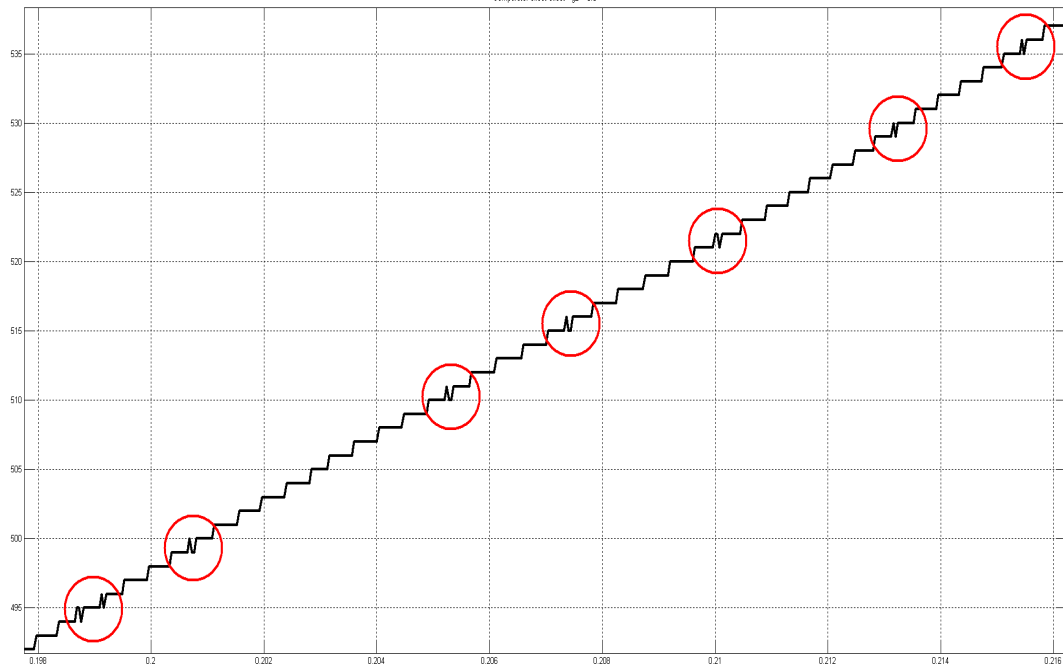
Figure 4.15 Glitches in the ADC I/O caused by the comparator's offset

4.6.2   Comparator's offset and resolution

A faulty comparator was simulated including the effect of finite offset $compOffset$ and limited resolution (or hysteresis) $compRes$: the hysteresis was modelled assuming the comparator would always make the wrong decision if the difference between its inputs was in absolute value lower than a set threshold.

Conversions with $compOffset$ up to $100mV$ were simulated: as expected, noise-shaping kills most of its effect at the output; however, near the lower edge of the transfer-curve, some glitches (such as those shown in Figure 4.15) could be observed (occasionally appearing in the first ~ 25 steps of conversion) even for very small offsets. These are anyway too few to give a consistent reduction of the ENOB; moreover, in that range the noise of the sensor will be dominated by the shot noise from the signal itself (large incoming signal means both larger noise and lower output voltage of the pixel, as explained in Chapter 1), hence these glitches wouldn't be significant in the target application.

In order to ensure that the comparator had no relevant effect on the ENOB, the value of $compRes$ was set to $2mV$ (for $g_1 = 0.2$, which is the nominal value, chosen so that the signals would fit the amplifiers' output range – as explained in Chapter 6; if g_1 was increased, $compRes$ can be increased by the same factor): for greater values a large number of glitches started to emerge, appearing first at the edges of the transfer-curve and then progressively moving towards the centre as the simulated resolution was worsened. The input common mode range at which this specification must be met is a small fraction of the FSR: for g_1 equal to the nominal value 0.2, it was observed to be contained within $250mV$ of V_{CM}.

4.6.3   Noise

White noise sources were added at the inputs of the 1st and 2nd stage of the ADC and the corresponding standard deviation of the output for fixed input signal was measured. It was verified that the worsening factor for the input noise W is 4/3, as described in Section 3.1.4. The noise of the second stage was simulated in an ideal SDM to have an impact ~ 550 times smaller than that of the input noise (the reduction factor is a ratio between variances), thanks to noise shaping (coefficients used in the simulation were $g_1 = 0.2$, $g_2 = 0.25$).

Relying on noise-shaping, no specifications were set in terms of noise for the 2nd stage and the comparator's performance. The OTA of the 2nd stage, in fact, will have an input transconductance comparable to that of the 1st stage due to speed performance, and hence similar noise performance; the comparator, on the other hand, is expected to have no influence at all on noise due to the improved noise shaping. These assumptions were confirmed *a posteriori* with circuit-level simulations, as will be shown in Chapter 8.2.

Given the short time between two consecutive resets of the ADC, $1/f$ noise was assumed not to be relevant and thus wasn't simulated; in Chapter 6.4 it is explained how it was accounted for at circuit level design, while in Chapter 8.2 it is demonstrated that $1/f$ noise is indeed negligible.

Table 4.6 Analogue specifications derived from behavioural simulations

G_{DC}	> 200
$compOffset$	$< 100mV$
$compRes$	$< 2mV$ (for input range within $250mV$ of V_{CM})
Input noise - 1st stage	$< 100\mu V_{rms}$
Input noise – other stages	Irrelevant

Chapter 5

Switched capacitors circuits

One of the outcomes of the analysis of complex $\Sigma\Delta$ architectures that was carried out in Chapter 4 is that the ADC will need to be time-discrete, i.e. to operate in cycles: this is achieved in analogue systems with switched capacitors (SC). SC systems are time-discrete circuits which exploit conservation of charge and the virtual ground of an amplifier to emulate the resistance of a time-continuous filter. Before moving on to the circuit design it is convenient to introduce the SC technique and analyze some of its constraints, in order to lay the groundwork for the discussion regarding analogue design in Chapter 6.

5.1   Principle of operation

Figure 5.2 shows the simplest configuration of a SC cell.

Switched capacitor circuits always work in two non-overlapping phases, here called Phi1 (or Φ_1) and Phi2 (or Φ_2), clocked at a frequency f_s:

1- Charge is stored on the sampling capacitance C_S;
2- C_S releases the charge to the virtual ground VG
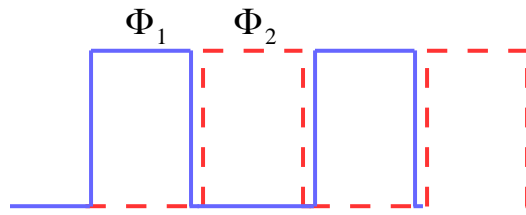
Figure 5.1 Non-overlapping clocks

The combination of these two cycles makes the SC cell behave, on average, like a two terminal component crossed by a current:

$$I_{cycle} = V_{in} \cdot C_S f_s = \frac{V_{in}}{R_{eq}} \qquad (5.1)$$

$$R_{eq} = \frac{1}{C_S f_s} \qquad (5.2)$$
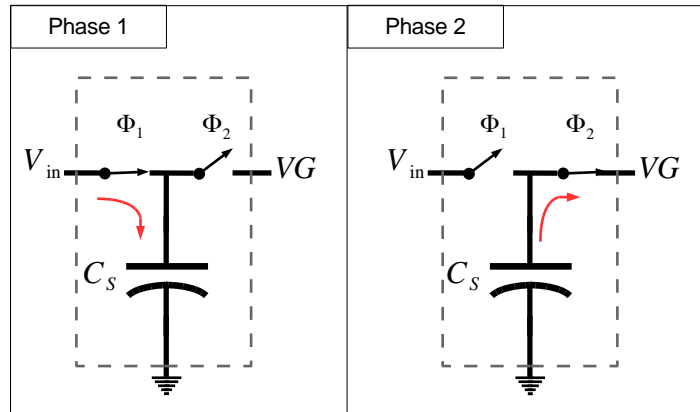
Figure 5.2 Phases in a basic switched capacitor cell

As shown in Eq. **Errore. L'origine riferimento non è stata trovata.**, this time-discrete operating cell emulates the action of a time-continuous circuit with an equivalent resistance R_{eq} in place of the switched capacitor, and can thus be used to implement the same functions as those of amplifiers in a time-continuous feedback configuration – such as an integrator.

The simple structure in Figure 5.2, however, has the major drawback that the transfer is directly affected by any stray capacitance (see Figure 5.3 - left): the effective capacitance would be $C_{eq} = C_S + C_{stray}$. The consequent lack of control over the transfer gain is an issue that must be corrected. To do this, the stray-insensitive configuration shown in Figure 5.3 (right) is commonly adopted: in this configuration, the charge is indeed stored also on one parasitic capacitor (C_{stray1} in figure) during phase 1, but it won't be released to the VG during phase 2, thus preserving the correctness of the transfer function.
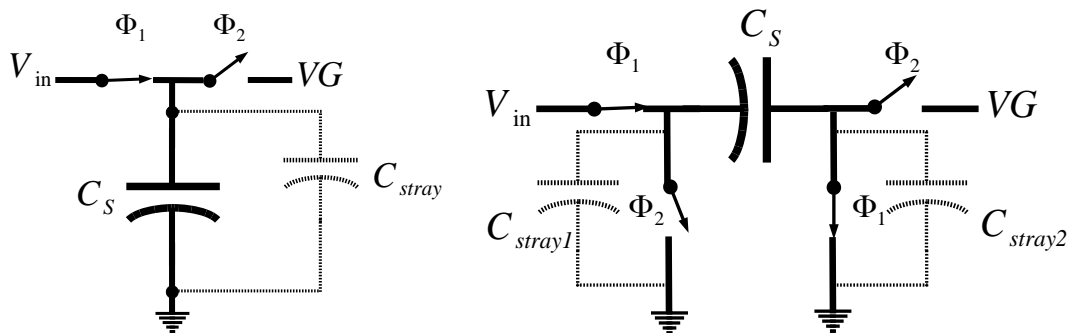
Figure 5.3 Stray capacitances in a simple SC cell (left) and in a stray-insensitive SC cell (right)

Since we are interested in the use of SC only to realize an integrator, this configuration will be analyzed in detail in the following section.

5.2 Switched Capacitor Integrator

Figure 5.4 shows a SC integrator stage: it features an amplifier (without loss of generality, we will consider it to be an operational trans-conductance amplifier – OTA), a sampling capacitor C_S and an integration capacitor C_{INT}. Note that the SC-cell containing C_S is now

considered to be a four-terminal cell, the terminals being respectively V_a, V_b, V_c and the virtual ground; the common mode of the amplifier is here called V_d.

The phases will now be described in detail. As a reference, we will consider the end of the nth cycle to coincide with the end of phase 2, thus the end of the preceding phase 1 will be the cycle $n - 1/2$.

1- During phase 1 (Figure 5.5), a charge $Q_S = Q_{start} = C_S\left[V_a\left(n - \frac{1}{2}\right) - V_c\left(n - \frac{1}{2}\right)\right]$ is stored on the sampling capacitor C_S; the OTA output V_{INT} instead, provided any leakage is negligible, will stay fixed at the voltage $V_{INT} = V_d\left(n - \frac{1}{2}\right) + V_{C_{INT}} = V_d\left(n - \frac{1}{2}\right) + \frac{Q_{INT}(n-1)}{C_{INT}}$;

2- During phase 2 (Figure 5.6), the OTA will react by injecting or drawing current to or from its output node (depending on the sign of its differential input) to restore the virtual ground. Throughout this process the capacitor C_{INT} receives exactly the same charge as C_S (since the input pins of the ideal OTA don't draw any current), hence will have experienced the same charge variation, regardless of the specific voltages' behaviour during the transient:

$$-\Delta Q_{C_S} = \Delta Q_{C_{INT}}$$

$$-C_S\left[V_b(n) - V_d(n) - V_a\left(n - \frac{1}{2}\right) + V_c\left(n - \frac{1}{2}\right)\right] \qquad (5.3)$$
$$= C_{INT}[V_{INT}(n) - V_{INT}(n - 1)]$$

Hence:

$$V_{INT}(n) = V_{INT}(n - 1) + \frac{C_S}{C_{INT}}\left[V_a\left(n - \frac{1}{2}\right) - V_b(n) - V_c\left(n - \frac{1}{2}\right) + V_d(n)\right] \qquad (5.4)$$

$$V_{INT}(z) = \frac{C_S}{C_{INT}} \cdot \frac{\left(z^{-\frac{1}{2}}V_a - V_b - z^{-\frac{1}{2}}V_c + V_d\right)}{1 - z^{-1}} \qquad (5.5)$$
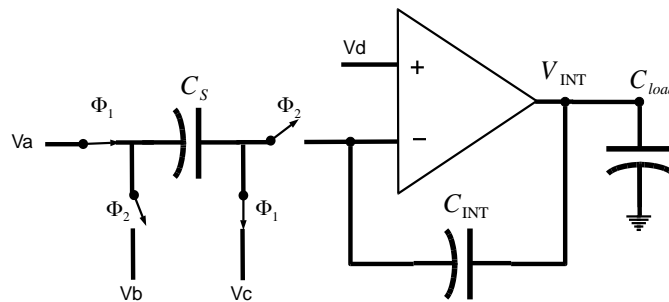
Figure 5.4 SC integrator
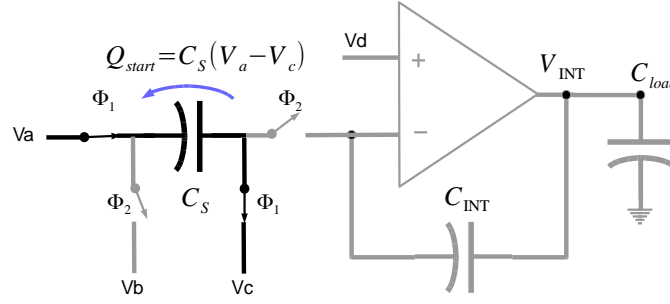
Figure 5.5 SC integrator: sampling phase (phase 1)

$$\Delta Q_{INT} = C_S(V_a - V_c - V_b + V_d)$$
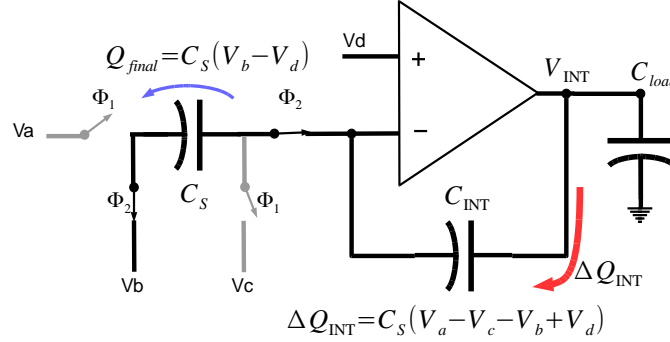
Figure 5.6 SC integrator: integrating phase (phase 2)

From Eq. (5.4) - (5.5) we can see that the stage behaves indeed like an integrator with gain $g = C_S/C_{INT}$; it should further be noted that if C_{INT} integrates the difference between two signals (for example, $V_a - V_b$), their common mode won't matter as it will cancel out. Therefore, the OTA can work with a common mode V_{CM} different from that of its inputs: this feature was exploited to shift the common mode of the designed SDM to a value different from the middle of the input range.

In the first stage of a $\Sigma\Delta$ modulator, the voltages can then be: $V_a = V_{in}$, $V_b(n) = V_{DAC}(n-1)$, $V_c = V_d = V_{CM}$. Note that the DAC voltage fed at node V_b is delayed by one sample, since the value to be used at cycle n will be obtained at cycle $n-1$.

In the second stage, in order to perform the inverting configuration necessary for the architecture chosen in Chapter 4.5 and shown in Figure 4.11, we need $V_b = V_{INT1}(n-1/2)$, $V_a = V_c = V_d = V_{CM}$ (inverting configuration).

Since the output of an integrators doesn't change during phase 1, we can write: $V_{INT}(n-1/2=VINTn-1$. This allows us to set the term $z-12$ in the transfer function equal to $z-1$:

$$V_{INT1} = g_1 \cdot \frac{z^{-1}}{1-z^{-1}} (V_{in} - V_{DAC}) \tag{5.6}$$

$$V_{INT2} = -g_2 \cdot \frac{1}{1-z^{-1}} \cdot (V_{INT1} - V_{CM}) \tag{5.7}$$

Note that the inverting configuration gives a non-delayed transfer function, hence the absence of the term z^{-1} at the numerator in Eq. (5.7).

5.3 Settling error

Two types of settling error (ε_s) must be distinguished: input-dependent settling error (or non-linear ε_s) and settling error independent of the input (or linear ε_s) [11]. The former can be caused by slewing of the OTA or the non-linearity of the MOSFET switches, which change region of operation throughout the transient. It is a possible cause of distortion and, therefore, of increase in INL and DNL. In our case, the input of the first stage throughout a conversion can have two values, $(V_{in} - V_{DAC}^L)$ and $(V_{in} - V_{DAC}^H)$, hence the largest settling error-related distortion can be expected to occur for values of V_{in} that give the maximum and minimum values of the ratio $\frac{V_{in} - V_{DAC}^L}{V_{in} - V_{DAC}^H}$. This will occur for V_{in} closest to V_{DAC}^H or V_{DAC}^L.

Linear settling error, on the other hand, is simply related to the finite switches resistance and op-amp bandwidth, and can be accounted for as a modification of the effective gain of the integrator [11]. This is shown in Eq. (5.8), where the transfer function is assumed to have a single pole τ_p and T_{Φ_2} is the duration of phase 2:

$$g_{eff} = g \cdot (1 - \varepsilon_s) = g \cdot \left(1 - e^{-\left(\frac{T_{\Phi_2}}{\tau_p} \right)} \right) \tag{5.8}$$

5.4 Slewing

At the beginning of phase 2 (instant $t = 0^+$), the capacitor is connected to V_b and to virtual ground: assuming that the switches are ideal, and in particular that their resistance is much shorter than the OTA's transconductance, the output of the OTA can be considered at high impedance during the instant immediately following the closing of the switches. Hence, C_S will retain its charge: this will cause the virtual ground $V_g|_{0^+}$ and the output V_{INT} to quickly move together with the bottom plate of C_S:

$$\Delta V_g|_{0^+} = \Delta V_{INT}|_{0^+} = V_b(n) - V_a\left(n - \frac{1}{2} \right) \tag{5.9}$$

It should be noted that this variation has the opposite sign compared to the relation in Eq. (5.5), with respect to inputs V_a and V_b: this entails that the voltage swing to be covered by V_{INT} during phase 2 to reach complete settling is larger than Eq. (5.5) alone suggests, and is given by:

$$V_{INT}(n) - V_{INT}(0^+) = V_{INT}(n) - V_{INT}(n-1) - \Delta V_{INT}|_{0^+}$$

$$= (1 + g)\left[V_a\left(n - \frac{1}{2} \right) - V_b(n) \right] = \frac{\left[V_a\left(n - \frac{1}{2} \right) - V_b(n) \right]}{\beta} \tag{5.10}$$

Where the feedback factor was introduced:

$$\beta = \frac{C_{INT}}{C_S + C_{INT}} = \frac{1}{1 + g} \tag{5.11}$$
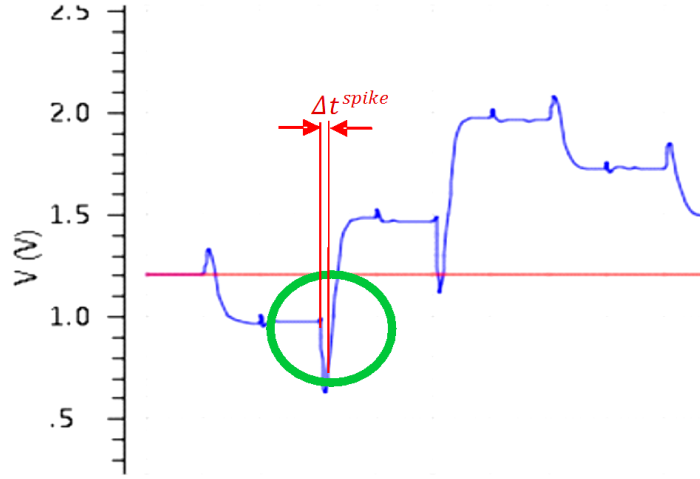
Figure 5.7 Negative spikes - caused by inability of C_{sample} to instantaneously release its charge - increase minimum SR specification

The result obtained in Eq. (5.10) indicates that the slew-rate SR of the OTA (the maximum rate at which its output voltage can move) must be large enough to counteract this negative spike; however, Eq. (5.10) gives a much worse result than simulated, since the assumption of switches with resistance negligible with respect to the OTA transconductance does not hold in practice. Moreover, it doesn't take into account charge injection and clock feed-trough (explained later in Section 5.6), which change the initial condition for V_{INT} in a way difficult to predict: therefore, the relation derived in Eq. (5.10) is only indicative of the order of magnitude of the slewing, and the designer should ultimately rely on transient simulations to assess the minimum current necessary to overcome it.

The demand for higher SR is worsened by the fact that the spike seen takes a finite portion Δt_{spike} of the duration of phase 2 (T_{Phi2}) to occur, thus reducing the effective time available for settling. In simulations with $T_{Phi2} = 5ns$, the observed delay was $\Delta t_{spike} \cong 1ns$.

5.5   Finite op-amp gain

Until now the OTA was assumed to have infinite DC gain G_{DC}. However, an OTA with low G_{DC} will not behave as a perfect integrator: it is sometimes referred to as "lossy" or "leaky" integrator, because it won't be able to integrate all the charge from capacitor C_S: a residual amount $C_S \cdot V_{INT}(n)/G_{DC}$ will be left on it at the end of every cycle. Since this charge is not directly related to the input but, instead, to the output, it cannot be accounted for in terms of simple offset or gain modification: it will affect the pole p of the discrete-time transfer function. The effects of a lossy integrator on the performance of a ΣΔ have been analyzed in Chapters 3.3.1.

Referring to Figure 5.4-Figure 5.6 and assuming for simplicity and clarity that $V_b = V_c = V_d = 0$, the charge balance of Eq. (5.3) must be rewritten to account for the non-nil OTA differential input $v_\varepsilon = V_{INT}(n)/G_{DC}$:

$$-C_S \left[V_a \left(n - \frac{1}{2} \right) - v_\varepsilon(n) \right]$$
$$= C_{INT} [V_{INT}(n) + v_\varepsilon(n) - V_{INT}(n-1) - v_\varepsilon(n-1)]$$

(5.12)

The effective gain g_{eff} and pole p of the transfer function can be derived to be:

$$p = \frac{1}{1 + \frac{g}{1 + G_{DC}}} \xrightarrow{G_{DC} \gg 1} 1 - \frac{g}{G_{DC}}$$

(5.13)

$$g_{eff} = \frac{g}{1 + \frac{1}{\beta G_{DC}}}$$

(5.14)

5.6 Charge injection and clock feed-through

The switches can be another source of distortion, causing non-linear clock feed-through and charge injection.

Clock feed-through occurs when clock Φ_1 or Φ_2 commutes. Figure 5.8 shows qualitatively its dynamic: the coupling between capacitor C_{gd} and C_S will cause additional charge to be deposited on the sampling capacitor at every cycle, thus introducing an offset in the transfer. This offset, however, can be input-dependent, because most of the charge will be injected when the transistor is off (since it shows high impedance) and will hence depend on the switching off voltage $V_{in} + V_{thr}$ (assuming that the switch connected to the input is turned off first, as shown in Figure 5.8), where V_{thr} is the threshold voltage of the transistor. If the transition of the clock from 1 to 0 is assumed to be very slow compared to the speed at which the switch can recollect the injected charge[2], a rough quantitative estimation of the effect of clock feed-through can be:

$$\Delta V_{C_S} \cong (V_{in} + V_{thr}) \cdot \frac{C_{gd}}{C_{gd} + C_S}$$

(5.15)

If the clock transition is instead assumed to be fast, the switch can always be considered at high impedance, hence:

$$\Delta V_{C_S} \cong V_{dd} \cdot \frac{C_{gd}}{C_{gd} + C_S}$$

(5.16)

The portrait just given is very simple compared to reality (a better mathematical treatment of its effect is given in [29]), hence Eq. (5.15) is not accurate: it does, however, highlight the two main characteristics of clock feed-through:

1. The larger the switch size, the larger will C_{gd} be, and the feed-through effect will thus worsen.

---

[2] Assuming that the resistance is dominated by the switch which is turning off, the characteristic time of charge recollection by the switch can be estimated to be $\tau \cong R_{sw}(C_S + C_{gd})$.
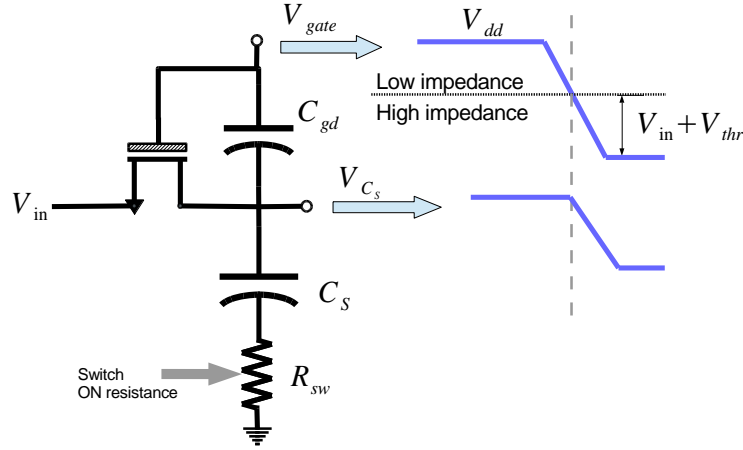
Figure 5.8 Clock feed-through: qualitative visualization in the case of slow clock transition

2. Clock feed-through can depend on the input V_{in}, since its value affects the on resistance of the switch and its switch-off point.

Charge injection, on the other hand, is related to the periodic collection and release of channel charge by the MOSFET switches. Referring to Figure 5.9, let's consider the two switches connected to Φ_1: they are both connected on one side to a voltage source, so their channel charge will be provided by that source; however, when they turn off the evenly distributed charge in the channel will be released on both sides, thus flowing also on C_S. For a fast transition of the gate, half of it will flow on one side and half on the other [30]; on each side, the charge will further be shared between the stray capacitance of the switch and whichever external capacitance C_{ext} is connected to it (see Figure 5.9 (b)). C_{ext} is thus the capacitance seen from drain towards ground by each switch, and depends on which one is turned off first: for the first one to turn off $C_{ext} = C_S$ (since C_{stray} is short-circuited by the other switch), while for the second one $C_{ext} = C_S C_{stray}/(C_S + C_{stray})$.

The portion of charge that flows on $C_{ext}/(C_{ext} + C_{stray})$ will hence be deposited also on C_S, for a total of:

$$Q_{C_S} = \frac{Q_{ch}}{2} \cdot \frac{C_{ext}}{C_{ext} + C_{stray}} = \frac{C_{OX}(V_{dd} - V_{thr} - V_{source})}{2} \cdot \frac{C_{ext}}{C_{ext} + C_{stray}} \qquad \textbf{(5.17)}$$

In Eq. (5.17) V_{source} is 0 for the switch to the right and equal to V_{in} for the switch to the left; As we have seen for clock feed-through, this periodic injection introduces an offset in the transfer, which depends on the input in a similar manner.
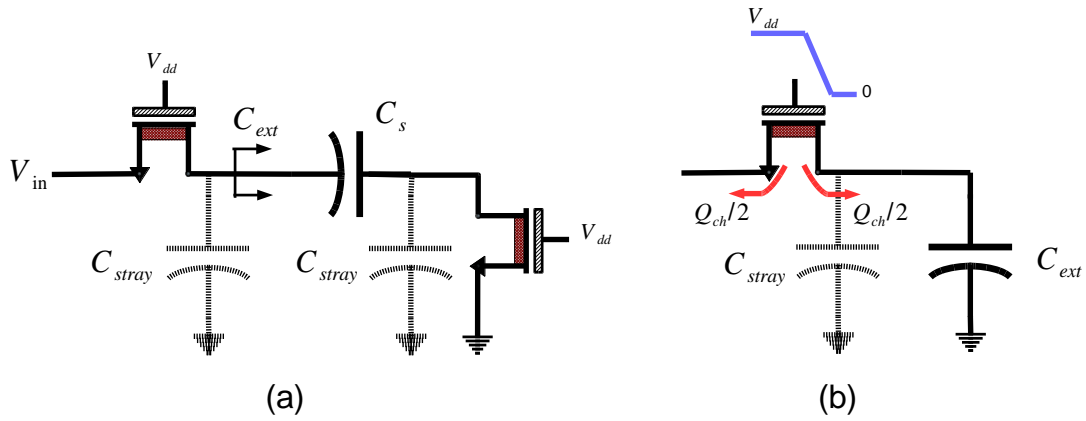
Figure 5.9 MOSFET switches connected to the sampling capacitance with channel charge Q_{ch} in evidence (a) and charge injection to the external capacitance C_{ext} (b)

Charge injection and clock feed-through can be reduced by using delayed clocks such as shown in Figure 5.10: in this way the switch connected to the input will always be the last to turn off and C_S will be disconnected from ground, thus showing high impedance: the input dependent capacitive partition of clock feed-through will hence be lessened and a smaller fraction of the input dependent channel charge will flow on C_S, according to Eq. (5.17).

In order to compensate these effects the switches are often replaced with transfer gates (a pMOS and an nMOS in parallel, clocked with complementary phases), or the series of an nMOS switch and a dummy nMOS with source and drain short-circuited [31]: the pMOS in the former solution and the dummy nMOS in the latter serve the purpose of recollecting the charge injected by the nMOS switch. The use of fully differential amplifiers also helps the compensation.

In our case, given the loose linearity requirements, simple nMOS switches were employed, but clocked with delayed phases shown in Figure 5.10.
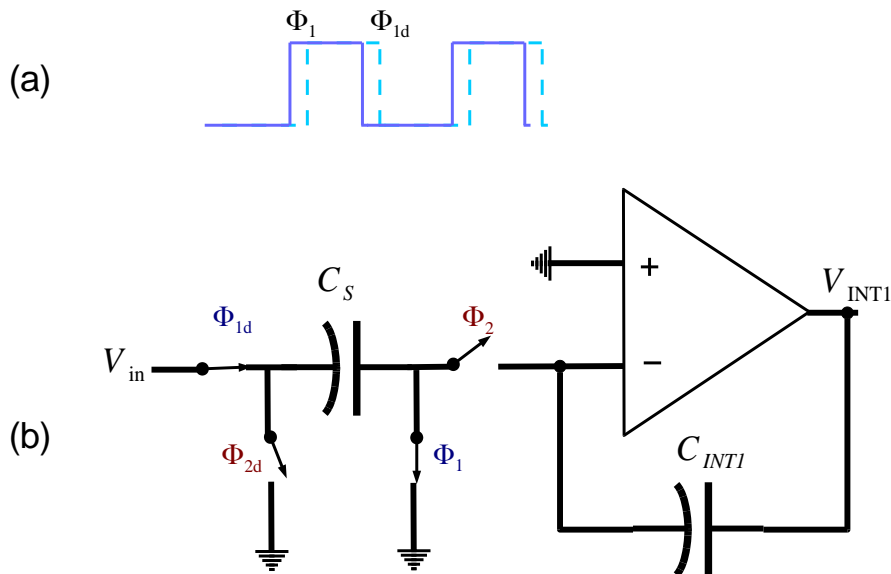
Figure 5.10 Delayed clocks for phase 1 (a) and relative connections in a switched capacitor integrator

5.7   White noise in SC circuits

The several noise sources in a SC circuit affect the transfer in the form of noise charge sampled on either C_S or C_{INT}. In particular, the spurious charge injected on C_S will only affect the transfer if it manages to be transferred to C_{INT} before being "flushed" away by a connection to low impedance nodes (voltages V_a, V_b, V_c in Figure 5.4) through the switches.

A thorough mathematical treatment of noise transfer would need to be set in the frequency domain and to use basic signals' theory concepts such as aliasing and convolutions. In the case of white noise however, since noise samples from the same source are all uncorrelated, the treatment can be carried out in the samples' domain, evaluating the standard deviation σ_{sample} of the noise charge injected by every source at each cycle and then using Eq. (3.15).

Before beginning the analysis we remember that the standard deviation of a white noise source with spectral density S at the output of a single pole system with time constant τ is:

$$\sigma^2 = S \cdot |T(\omega)|^2 = S \int_0^\infty \frac{1}{1 + (\omega\tau)^2} \, d\left(\frac{\omega}{2\pi}\right) = \frac{S}{4\tau} \tag{5.18}$$

and that the unilateral white spectral density of a resistor is given by the Johnson-Nyquist theorem:

$$S_R = 4kTR \tag{5.19}$$

Where k is Boltzmann's constant, T the temperature and R the resistance. The white noise input spectral density of the OTA was instead derived in Appendix A.

During phase 1, the bottom plate of C_{INT} is only connected to the OTA input, thus it won't be able to accept nor release any charge. Noise charge can instead be deposited on C_S by the switches' resistance R_{sw} and by the buffer driving the input node of the SC stage: the charge remaining on C_S after the switches turn off and become high impedances will be transferred to C_{INT} during the following phase 2. Noise introduced in this way is usually referred to as reset noise.

If the buffer at the input is ideal in terms of both noise and bandwidth, then it can be excluded. The standard deviation of the noise charge due to the switches only will then be:

$$\sigma_{sw1}^2 = C_S^2 \sigma_{V_{Cs}}^2|_{switch} = C_S^2 \, 2 \cdot \frac{4kTR_{sw}}{4 \cdot (2R_{sw}C_S)} = kTC_S \tag{5.20}$$

The result from Eq. (5.20) is typical of the situation just depicted, where the resistor $2R_{sw}$ is both what causes the noise and what sets the bandwidth. In reality, the situation is different: let's now suppose to have a noisy buffer; for simplicity, we will assume that the input noise is only given by the input transistor pair, hence it will have a spectrum: $S_{buff} = 2 \cdot 4kT\gamma \cdot 1/g_{m_{buff}}$. We further neglect the presence of a parasitic load capacitance connected to the output of the buffer. Let $R_{out}|_{openloop}$ be its output resistance at open loop, and $r_{out} = R_{out}|_{openloop}/G_{DC}$ be its output resistance at closed loop: if the amplifier is an OTA, $G_{DC} = g_{m_{buff}}R_{out}|_{openloop}$, hence $r_{out} = 1/g_{m_{buff}}$; otherwise, if an OpAmp is used, $R_{out}|_{openloop}$ and thus r_{out} will be much lower than $1/g_{m_{buff}}$. Referring to Figure 5.11, which shows the Thevenin equivalent of the buffer, the standard deviation of the deposited noise charge due to the switches and the driver will be:

$$
\begin{aligned}
\sigma_{buff}^2 = C_S^2 \sigma_{V_{Cs}}^2|_{buff} &= C_S^2 \cdot \frac{S_{buff}}{4[C_S(2R_{sw} + r_{out})]} \\
&= kTC_S \cdot \frac{2\gamma}{r_{out}g_{m_{buff}} + 2R_{sw}g_{m_{buff}}} = kTC_S \cdot F_{buff}
\end{aligned}
\tag{5.21}
$$

$$
\sigma_{sw1}^2 = C_S^2 \sigma_{V_{Cs}}^2|_{switch} = kTC_S \cdot \frac{2R_{sw}}{2R_{sw} + r_{out}} = kTC_S \cdot F_{sw1}
\tag{5.22}
$$

From equations (5.21)-(5.22) it is clear that a driver with a very small output resistance r_{out} such as an OpAmp actually degrades the noise performance: compared to an OTA with same input transistors, it will have the same noise but larger bandwidth, hence more aliasing. In an OTA, instead, bandwidth and noise are set by the same transconductance, hence giving a performance more similar to that of the simple reset noise with a single noisy resistor. If that is the case, it will be:

$$
F_{buff} = \frac{2\gamma}{1 + x_{buff}}
\tag{5.23}
$$

$$
F_{sw1} = \frac{x_{buff}}{1 + x_{buff}}
\tag{5.24}
$$

Where we have defined: $x_{buff} = 2R_{sw}g_{m_{buff}}$.
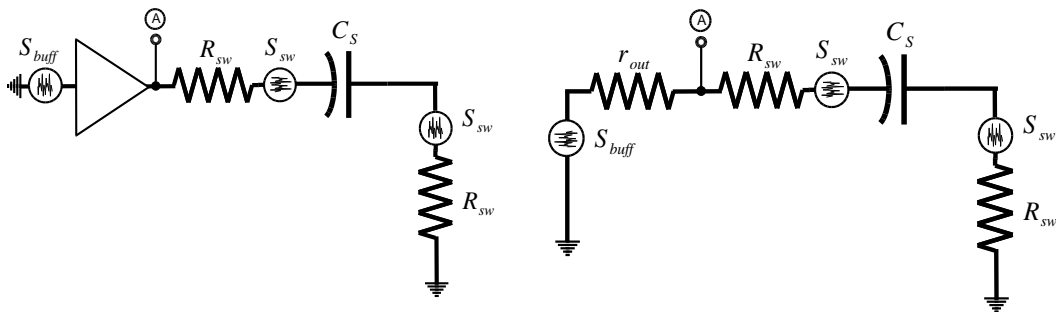
Figure 5.11 Noise sources during phase 1: ideal buffer (left) and its Thevenin equivalent (right)
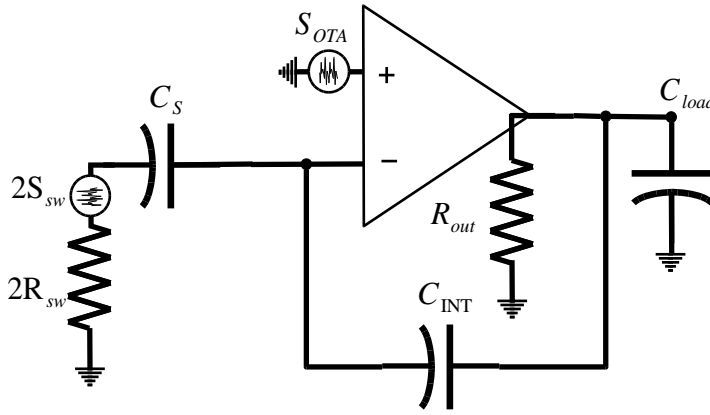
Figure 5.12 Noise sources during phase 2

During phase 2 (see Figure 5.12): two noise sources come into play, the switches connected to Phi2 and the OTA, and the bandwidth of the closed-loop system is set by a combination of their respective resistance and transconductance. Noise will come from the charge deposited on C_{INT}: whatever remains on C_S after the end of phase 2 will be removed during phase 1. The presence of a load capacitor C_{load} is also included for completeness: in fact, in our design, it has a value comparable to that of C_S. The loop gain of the system is:

$$G_{loop} = -\beta G_{DC} \cdot \frac{1 + s\tau_z}{as^2 + bs + 1} =$$

$$\cong -\beta g_{m_{OTA}} R_{out} \cdot \frac{1 + 2R_{sw}C_S}{2R_{sw}R_{out}\beta C_S C_{load}s^2 + (\beta C_S + C_{load})R_{out} \cdot s + 1}$$

(5.25)

The zero can be found by inspection; the coefficients a and b can be found using the time constants method:

$$b = \sum_i C_i R_{open}$$

(5.26)

$$\frac{b}{a} = \sum_i (C_i R_{closed})^{-1}$$

(5.27)

Where R_{open} and R_{closed} are the resistances seen by capacitor C_i when the other capacitors are replaced by open circuits or short circuits respectively.

Considering the output to be the charge on C_{INT}, both S_{OTA} and S_{sw} will have the same transfer function (with opposite sign) $T(s)$, given by:

$$T(s) = T|_{G_{loop} \to \infty} \cdot \frac{-G_{loop}(s)}{1 - G_{loop}(s)} =$$

$$\cong \frac{C_S}{\frac{a}{\beta G_{DC}}s^2 + \left(\frac{b}{\beta G_{DC}} + \tau_z\right)s + 1}$$

(5.28)

We would like now to approximate this system to have a single pole at $\tau_p = b/(\beta G_{DC}) + \tau_z$ $\cong \left(2R_{sw} + \frac{1}{g_m}\right)C_S + \frac{C_{load}}{\beta g_m}$; this is legitimate if:

$$\frac{4a}{\beta G_{DC}\tau_p^2} = \frac{4C_S C_{load}}{\left[C_S \cdot (x_{OTA} + 1) + \frac{C_{load}}{\beta}\right]^2} \ll 1 \qquad (5.29)$$

where we defined the term $x_{OTA} = 2g_{m_{OTA}}R_{sw}$. Condition (5.29) is satisfied if one between C_S and C_{load} dominates - which, as will be shown in Chapter 6.4, was not the case in our design, since $C_S = 30fF$ and $C_{load} \cong 50fF$; for simplicity, however, we will still consider to be dealing with a single pole system. At this point, the noise charge on C_{INT} due to both the switches and the OTA can be derived:

$$\sigma_{sw2}^2 = C_S^2 \cdot \frac{S_{sw}}{4\tau_p} = kTC_S \cdot \frac{x_{OTA}}{x_{OTA} + 1 + \frac{C_{load}}{\beta C_S}} = kTC_S \cdot F_{sw2} \qquad (5.30)$$

$$\sigma_{OTA}^2 = C_S^2 \cdot \frac{S_{OTA}}{4\tau_p} = kTC_S \cdot \frac{2\gamma\left(1 + \frac{g_{m_{MIRROR}}}{g_{m_{OTA}}}\right)}{x_{OTA} + 1 + \frac{C_{load}}{\beta C_S}} = kTC_S \cdot F_{OTA} \qquad (5.31)$$

We can now derive the equivalent input noise of the integrator, defined as the input voltage necessary to deposit on C_S a charge equal to the noise charge:

$$\begin{aligned}\sigma_{sample}^2 &= \frac{\left(\sigma_{sw1}^2 + \sigma_{buff}^2 + \sigma_{sw2}^2 + \sigma_{OTA}^2\right)}{C_S^2} \\ &= \frac{kT}{C_S} \cdot \left(F_{sw1} + F_{buff} + F_{sw2} + F_{OTA}\right)\end{aligned} \qquad (5.32)$$

5.8 $1/f$ noise in periodically reset SC circuits

A complete, rigorous analysis of $1/f$ noise would be lengthy and complex - especially in S&H circuits where aliasing is involved – and is beyond the purpose of this dissertation. The purpose of this paragraph is to show that, in the case in exam, where the integrators are periodically reset and form part of a ΣΔ ADC, $1/f$ noise is not a major cause of SNR degradation and that its effect can be estimated with simple (although not rigorous) equations.

$1/f$ noise is introduced by the OTA, and can be represented with an equivalent input noise source with spectrum $S_{OTA}^{(1/f)} = A_{1/f}/f = S_{OTA}^{white} \cdot f_c/f$, where f_c is called corner frequency. There are thus two ways in which it affects the integrator's transfer:

1) As a random, slow drift of the output common mode, hence as a time-dependent output offset.
2) As random additional charge sampled on C_S and transferred to C_{INT}, hence as time-dependent input offset.

Contribution 1) is taken out by noise-shaping, so it doesn't represent a problem; contribution 2) instead, being introduced at the input, could affect the noise of the system. However, we have to take into account that in an Incremental Sigma-Delta the capacitors will be periodically reset, thus an intrinsic high-pass filtering is present in the system. The power spectrum of $1/f$ noise is therefore limited by the cut-off at a frequency $1/T_{ADC}$ at the lower end and by $1/2\pi\tau_p$ - the closed loop bandwidth of the amplifier – at the higher end. An analytical expression for the total noise power in this case of filtering doesn't exist, but we can assume that the order of magnitude will be the same of the case of filtering by a CR-RC filter with similar cut-off frequencies. The result will therefore be in the order of:

$$\sigma_{1/f}^2 \sim K \cdot S_{OTA}^{white} f_c \cdot \ln\left(\frac{T_{ADC}}{2\pi\tau_p}\right) = K \cdot S_{OTA}^{white} f_c \cdot \ln(\alpha \cdot M) \tag{5.33}$$

In the equation above M is the number of cycles, while K and α are correction factors in the order of units (the values used for a rough estimation were in our case $K = 2$ and $\alpha = 1$). The key point is that the total power of $1/f$ does not diverge, and it can be made negligible with respect to white noise thanks to the fast conversion frequency of the ADC.

Chapter 6

Analogue design – the modulator

In Chapter 4 a choice for the architecture of the Sigma-Delta was made, and in Chapter 5 the operation of SC integrators was analysed: it is now time to deal with the design of the $\Sigma\Delta$ modulator at transistor level.

Firstly, the main characteristics of the process used will be reviewed, followed by a description of the power supplies needed for the chip to work. Then we shall give an overview of the modulator and the behaviour of its components during the two phases of operation. The design of its blocks will subsequently be exposed in detail, starting with the integrator stages and finishing with the comparator and DAC.

6.1   Characteristics of the process

The test chip developed in this project uses TowerJazz$^{®}$ 0.18µm process for CMOS Image Sensor (CIS). This process allows transistors of different oxide thickness to be used in the same design. Thick oxide (or "high voltage", HV) transistors have a higher threshold voltage V_{thr}, a larger minimum length ($L^{(hv)} = 0.35\mu m$) and normally work with 3.3V as nominal supply. Thin oxide transistors (or "low voltage", LV) have lower V_{thr}, their minimum length is $0.18\mu m$ and nominally work with a supply of 1.8V - although they can work at up to 2.2V supply. The former type of transistor was used for analogue stages for its higher gain, while the latter is more suited for digital operation since it is faster (low threshold) and will dissipate less dynamic power (lower supply voltage).

(a)   (b)
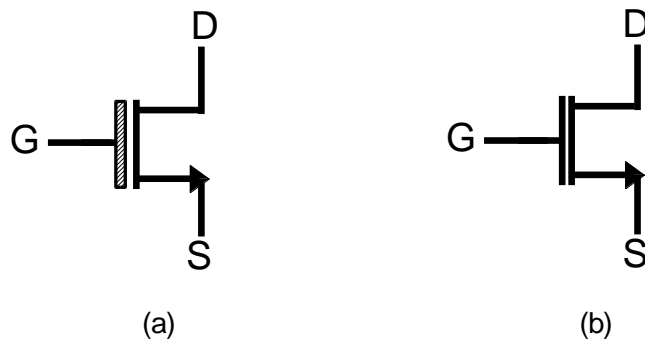
Figure 6.1 Symbols used for thick oxide, HV MOSFET (a) and thin oxide, LV MOSFET (b)

6.2   Supplies used

The system works in total with 5 supplies (plus two grounds, one for the digital and one for the analogue stages), summarized in the table below. The numbers in the name of the supply tell its value, while the suffix "A" or "D" distinguishes whether they were dedicated to analogue or digital stages.

Table 6.1 Supplies used in the design and blocks supplied

VDD_A33	OTAs and comparator's input stage
VDD_A2	Comparator's pMOS track branches $\rightarrow V_{DAC}^{H}$
VDD_D33	Non overlapping clocks generator
VDD_D2	All other digital signals
VDD_D18	DLPF (decimator)

6.3   Modulator overview

Figure 6.2 re-proposes the top level schematic of the modulator introduced in Chapter 4.5.2: we remember that the peculiarity of this architecture is that the feed forward operation is not given by charge sharing between capacitors (as is the common practice), but is instead achieved by first inverting the output of the second stage V_{INT2} and then using the comparator to compare it with the output of the first stage V_{INT1}. This allows to save the area that would be taken by the feed forward capacitors.

Figure 6.3 shows the operation of the modulator during its two phases: during phase 1 V_{in} and V_{INT1}, being the inputs of the first and second stage, are sampled on C_{S1} and C_{S2} respectively; at the same time the integrators' outputs V_{INT1} and V_{INT2}, which are at all times connected to the comparator's positive and negative input (respectively called C_OutPos and C_OutNeg in the figure), are being compared to compute the DAC output. This will then be fed out at the beginning of phase 2, during which the integrators will integrate the previously sampled charge on C_{INT1} and C_{INT2} and update their outputs.

The phases of operation are scanned by non-overlapping clocks named Phi1 and Phi2; in order to reduce non-linearity from charge injection and clock feed-through, as explained in Section 5.6 their delayed versions Phi1d and Phi2d are connected to the input switches.

Figure 6.2 Schematic of the designed Sigma Delta Modulator
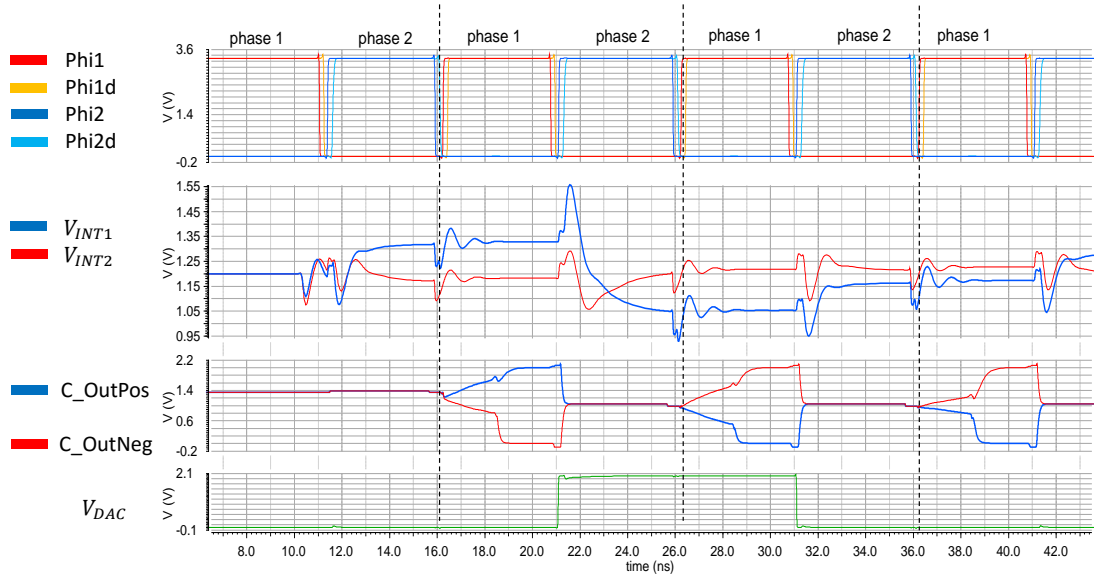
Figure 6.3 Waveforms of clocks, integrators and comparator's inverting and non inverting outputs in the developed Sigma Delta Modulator.

6.4 Integrator stages

6.4.1 OTA architecture

For compactness and containment of the power consumption, the amplifier was chosen to be a single stage OTA, in particular a telescopic cascode OTA: this architecture in fact ensures high DC gain G_{DC} and good tolerance to disturbances coupled to the power supply (such as resistive voltage drops caused by switching in a nearby ADC) [27]. A further advantage for high speed operation of this architecture compared to a simple differential pair is that, while in the latter the load is directly affected by the size of the input transistors through their drain-to-substrate parasitic capacitance, in a cascode configuration the size of the input pair has a lower impact on the load (not negligible though, since their gate capacitance is in the path of the loop), and can thus be made larger to increase the bandwidth.

In literature, other options have also been suggested. In particular, non-differential or single branch OTAs - such as the one shown in Figure 6.4 - are used, which dissipate half the power to operate at the same frequency (since the bias current is not split between the left and the right branch). The settling value of the virtual ground in these amplifiers is the driving voltage of the input transistor $V_{GS} = V_{OV} + V_{thr}$, and it hence varies throughout the conversion, since the overdrive V_{OV} depends on the output voltage: $V_{OV} \cong \overline{V_{OV}}|_{V_{out}=V_{CM}} + (V_{out} - V_{CM})/G_{DC}$ (in a linear approximation). Unless the specification for G_{DC} is substantially increased for the whole output range, V_{OV} won't be constant and, in particular, will behave non-linearly, thus degrading the ADC performance. In order to circumvent this problem, autozeroing techniques can be used which, at the price of adding one capacitor in the integrator stage, cancel the differential error voltage at the input of the OTA, thus allowing to directly set the common mode of the transfer and furthermore to reduce the effect of finite G_{DC} [32]. Using this technique, it was demonstrated that even CMOS

inverters used as OTAs are suitable for a ΣΔ [15]. However, it was noted through simulations that the efficiency of the voltage error cancelation in SC integrators that implement this technique is prone to be highly affected by the input parasitic capacitances of the OTA and, for this reason, in the end it was decided that the amplifier should be a simple and reliable differential OTA.

The input transistors are pMOS. This choice, which is not ideal in terms of speed (since the lower mobility of holes with respect to electrons makes it necessary to use pMOS larger by a factor of ~ 2.5 to have the same performance as nMOS), was driven by the architecture of the comparator, which also has pMOS inputs, as explained in Section 6.5.2.
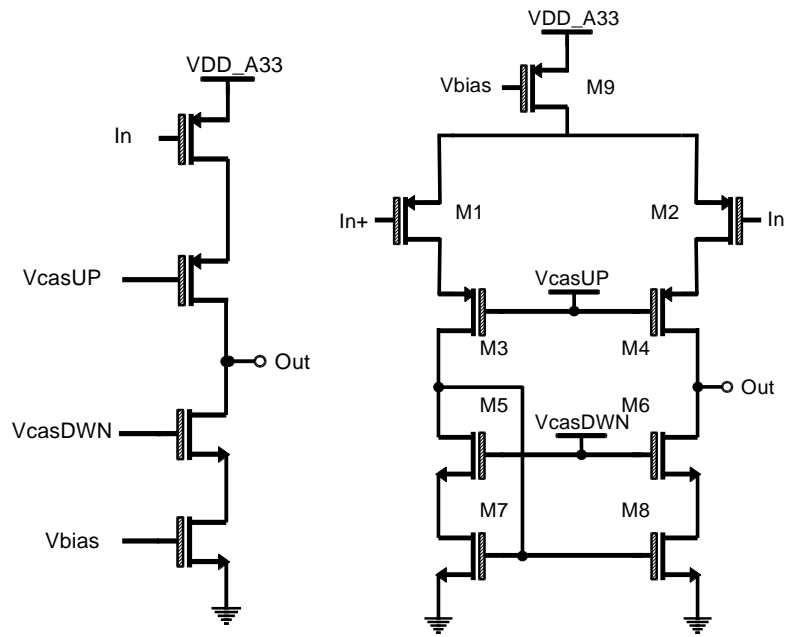
Figure 6.4 Telescopic cascode OTA: single branch (left) and differential (right)

6.4.2 Sizing of the integrator stages

Since the aim of the project is realizing an ADC suitable for an implementation in a stacked chip sensor, which might host tens of thousands of ADCs, the lengths and widths of the switches were set to be larger than the minimum allowed by the technology for yield purposes. Note that for the switches this is not ideal as, normally, minimum size is preferred to reduce charge injection and clock feed-through. The switches' corresponding large signal resistance was then estimated to be $R_{sw} \sim 4k\Omega$.

A first order estimation of the sizing and biasing of the first stage was done using hand calculations and running DC or ac simulations. Subsequently, transient simulations were run to have a closer look at the actual waveforms and see the effects of parasitics and charge injection. The process was then repeated. The design flow for the hand calculations is as follows:

1) Knowing that it won't have a large impact, start by assuming that $1/f$ noise has no contribution;

2) Make a guess for the OTA transconductance noise factor $\left(1 + g_{m_{MIRROR}}/g_{m_{OTA}}\right)$ and for the value of the noise parameters $x_{OTA} = 2g_{m_{OTA}}R_{sw}$ and $x_{buff} = 2g_{m_{buff}}R_{sw}$ introduced in Section 5.7. in order to estimate factor $F = F_{sw1} + F_{buff} + F_{sw2} + F_{OTA}$. For example, assuming $x_{OTA} = x_{buff} = 1$ and $\left(1 + g_{m_{MIRROR}}/g_{m_{OTA}}\right) = 1.5$ one gets $F = 2.67$;

3) Assuming that the beneficial effect of C_{load} on the noise is negligible, use Eq. (3.15) and Eq. (5.32) to derive the minimum sampling capacitance C_S which ensures that the noise specification is met:

$$C_S^{min} = W_2 \cdot \frac{kT}{M \cdot (\sigma_{in}^2 - \sigma_1^2)} \cdot \left(F_{sw1} + F_{buff} + F_{sw2} + F_{OTA}\right)$$
$$= \frac{4}{3} \cdot \frac{kT}{M \cdot (\sigma_{in}^2 - \sigma_1^2)} \cdot F \qquad \textbf{(6.1)}$$

4) After making an initial assumption for $g_1 = C_{S1}/C_{INT1}$ and thus $\beta_1 = 1/(1 + g_1)$, find the minimum current I_{OTA} that gives high enough SR for the worst case transition, which corresponds to an input equal to either $V_{in}^{min} - V_{DAC}^H$ or $V_{in}^{max} - V_{DAC}^L$; this is done using Eq. (5.10), to which it follows that I_{OTA} should be given by solving:

$$SR = \frac{I_{OTA}}{\beta C_S + C_{load}} > SR^{min} = \frac{\left(\dfrac{V_{in}^{max} - V_{DAC}^L}{\beta}\right)}{T_{Phi2}} \qquad \textbf{(6.2)}$$

The computation can be made, for example, assuming $C_{load} \cong \beta C_S$ (given the high speed of the system, all capacitances will be kept as low as possible and thus comparable to each other). The actual current should be chosen higher than the minimum by a margin factor; in our case, the over-sizing factor was ~1.4.

It is interesting to note that, if C_{load} is negligible compared to βC_S (which however is not our case), Eq. (6.2) can be simplified in such a way that I_{OTA}^{min} depends only on the general ADC specifications (neglecting $1/f$ noise):

$$I_{OTA}^{min}|_{C_{load}=0} = C_S^{min} \frac{(V_{in}^{max} - V_{DAC}^L)}{T_{Phi2}} = 2 \cdot M C_S^{min} \cdot \frac{(V_{in}^{max} - V_{DAC}^L)}{T_{ADC}}$$
$$= \frac{8}{3} \cdot \frac{kT}{\sigma_{in}^2} \cdot F \cdot \frac{(V_{in}^{max} - V_{DAC}^L)}{T_{ADC}} \qquad \textbf{(6.3)}$$

5) As noted in Chapter 5.2, the common mode V_{CM} can be set independently of the input range. Hence, after having made a guess for the size of all transistors, derive V_{CM}, V_{casUP} and V_{casDWN} which maximize the output range, which is done solving the system of Eq. (6.4)-(6.8) (refer to Figure 6.4):

$$V_{out}^{max} = V_{CM} + V_{thr}^{(M2)} - V_{OV}^{(M4)} \qquad \textbf{(6.4)}$$

$$V_{out}^{min} = V_{OV}^{(M8)} + V_{OV}^{(M6)} \tag{6.5}$$

$$V_{CM}^{optimum} = \frac{V_{out}^{max} + V_{out}^{min}}{2} \tag{6.6}$$

$$V_{casUP} = V_{out}^{max} - V_{thr}^{(M4)} \tag{6.7}$$

$$V_{casDWN} = V_{out}^{min} + V_{thr}^{(M6)} \tag{6.8}$$

6) Re-evaluate g_1 to adapt the output ranges of the two amplifiers to the working range necessary for ΣΔ operation.

7) Size the OTAs to ensure large enough gain-bandwidth-product (GBWP). In principle, as long as it is controlled, the settling error could be large, since it can simply be accounted for as a modification of the effective capacitor ratio g_{eff} (as explained in Chapter 5.3); however, the exponential relation between g_{eff} and $g = C_S/C_{INT}$ suggests that this factor might be affected by large spread if complete settling is not allowed; for the first stage, this shouldn't be a problem by itself, since the SDM loop conversion is independent of g_1; however, non-linearities related for example to the finite SR could affect the performance if the settling is not complete. For this reason, it was decided to aim for an almost complete settling (i.e. within 3-4%), although this might entail overdesigning and overestimating the necessary current and size of the OTA. A thorough assessment of the minimum necessary bias current will be easily done on the test chip.

In order to obtain the transfer function of the system of two integrators in series, the two were supposed to work independently from one another for all frequencies. This simplification gives:

$$\text{T(s)} = \frac{1}{(1 + s\tau)} \cdot \frac{1}{(1 + s\tau)} = \frac{1}{(1 + s\tau)^2} \tag{6.9}$$

where τ has the expression derived in Section 5.7, under the approximation of single-pole system:

$$\tau \cong \left(2R_{sw} + \frac{1}{g_m}\right) C_S + \frac{C_{load}}{\beta g_m} \tag{6.10}$$

The step response is then:

$$\frac{V_{INT}(t)}{V_{step}} = 1 - e^{-\frac{t}{\tau}} \left(1 + \frac{t}{\tau}\right) \tag{6.11}$$

To have settling error $< 4\%$, it must be $\tau < \frac{T_{clock}}{2} \cdot \frac{1}{5}$, hence:

$$g_m \sim \frac{C_S + \frac{C_{load}}{\beta}}{\frac{T_{clock}}{10} - 2R_{sw}C_S} \tag{6.12}$$

The size of the input transistors can then be estimated using the well known relation between MOSFET bias current and small signal transconductance when in

saturation: $W/L = g_m^2/(\mu_n C_{OX} I_{OTA})$, where μ_n is the electron mobility in the channel and C_{OX} the gate capacitance.

8) Once the definitive value of g_m is found, estimate $\sigma_{1/f}^2$ using Eq. (5.33); the values of K and α need to be guessed ($K = 2$ and $\alpha = 1$ were used in our case) while, with the appropriate simulations, S_{OTA}^{white} and the corner frequency f_c can be extracted, and thus the flicker noise coefficient $A_{1/f} = S_{OTA}^{white} \cdot f_c$ can be derived.

9) Reiterate from point 2) until the values found for g_m and I_{OTA} don't change too much.

10) If necessary, now increase the lengths of the transistors contributing to noise to definitely reduce $1/f$, and increase the widths by the same factor. Note that, since Eq. (5.33) is not rigorous and the factor K in it needs to be guessed, a fairly large margin of error should be allowed for this estimation - this anyway doesn't significantly complicate the design, since the impact of $1/f$ is easily overwhelmed by white noise.

11) Obtain the DC gain G_{DC} for all regions of operations: in particular make sure that, even when V_{INT} is close to the extremes of its range, the MOSFETs in the OTA are still well in saturation and G_{DC} is still meeting the requirements. Figure 6.5 plots G_{DC} as a function of the integrators' output voltage.

In case the DC gain obtained at this point failed to meet the specifications, make the length of the transistors longer to increase the output load resistance (and then increase their width by the same factor in order to maintain the g_m found in the previous passages unaltered).
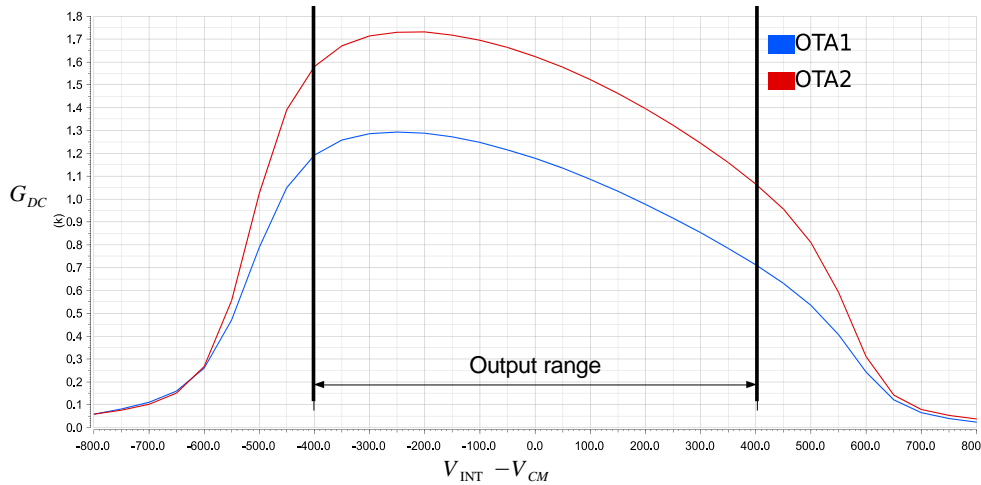
Figure 6.5 Linear plot of the DC gain of each OTA as a function of its output

The procedure described above, followed by transient simulations for adjustments, was used for the sizing of the first stage; given the resulting low size of the capacitor - in the order of tens of fF - and the requirement for its linearity, it was chosen that C_S and C_{INT} would be realized as metal-insulator-metal (MIM) capacitors, whose structure is shown in Figure 6.6 (another option to implement linear, small capacitors is usually metal-fringe capacitors, also shown in Figure 6.6, however this was not available in the run where the test chip would be fabricated; moreover, these capacitors are not area-efficient for larger capacitances – in the order of $\sim 100 fF$).

For the second stage a similar procedure was used, except that V_{CM} was now already set and that, not caring about the noise of this stage, the value of C_{S2} was automatically set to be the lowest considered acceptable for yield purposes.

The final capacitors and transistors sizes, bias current and small signals parameters are given in Table 6.2 and Table 6.3.
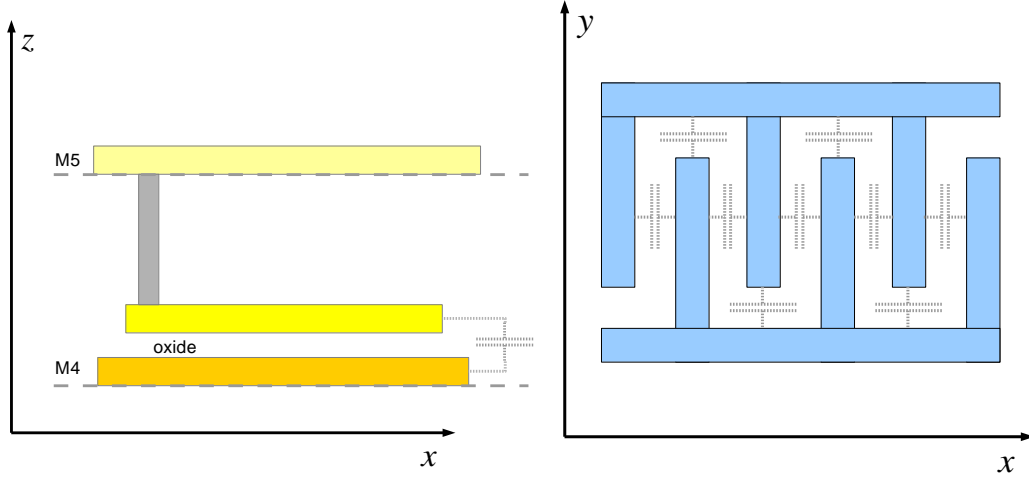
Figure 6.6 MIM capacitor cross-section (left) and metal-fringe capacitor top view (right)

Table 6.2 Parameters of the two integrator stages

V_{CM}	$1.2V = \dfrac{V_{DAC}^{H} + V_{DAC}^{L}}{2} + 200mV$
g_1	0.2
I_{OTA1}	$45\mu A$
G_{DC1}	1178
$g_{m_{IN1}}, g_{m_{MIRR1}}$	$206\mu A/V$, $120\ \mu A/V$
C_{S1}, C_{INT1}	$30fF$, $150fF$
g_2	0.25
I_{OTA2}	$27\mu A$
G_{DC2}	1624
$g_{m_{IN2}}$	$96,5\mu A/V$
C_{S2}, C_{INT2}	$30fF$, $120fF$

Table 6.3 Transistors sizes in the two OTAs

M1-2	10/0.4 (OTA1) , 5/0.4 (OTA2)
M3-4	4.2/0.4
M5-6	1.7/0.4
M7-8	1.5/0.6
M9	3/0.4 (OTA1) , 1.8/0.4 (OTA2)

6.4.3   Under-damping issue

The linear analysis carried out in Section 5.7 considered the contribution to the singularities in the transfer function of C_S and C_{load} only. The implicit assumption there was that poles and zeroes introduced by other parasitic capacitances were at a frequency reasonably higher than the GBWP and hence would not constitute a measurable presence. This is however not the case: given the high frequency at which the circuit has to work, the design was oriented towards containing the capacitive loads while making the size of the OTA transistors large enough to give high GBWP (for the input MOSFETs) and maximize the output range (for the cascode and mirror MOSFETs): this made internal parasitics comparable to the loads and hence to the GBWP. As a result, the phase margin of the OTAs can be lower than 45°: in particular, after phase 2 ends and C_{S1} and C_{S2} are disconnected from their respective OTAs, both the open loop gain and the bandwidth are increased (see Figure 6.7, where the Bode plot of the loop gain in the two different phases is shown), thus the GBWP will approach the poles at high frequency and it will increase by a factor:

$$\frac{(GBWP)^{phase1}}{(GBWP)^{phase2}} \sim \frac{G_{DC}}{\beta G_{DC}} \cdot \frac{(\beta C_S + C_{load})R_{out}}{C_{load}R_{out}} = \left(1 + \frac{C_S}{C_{load}} + g\right) \tag{6.13}$$

$$G_{loop}$$
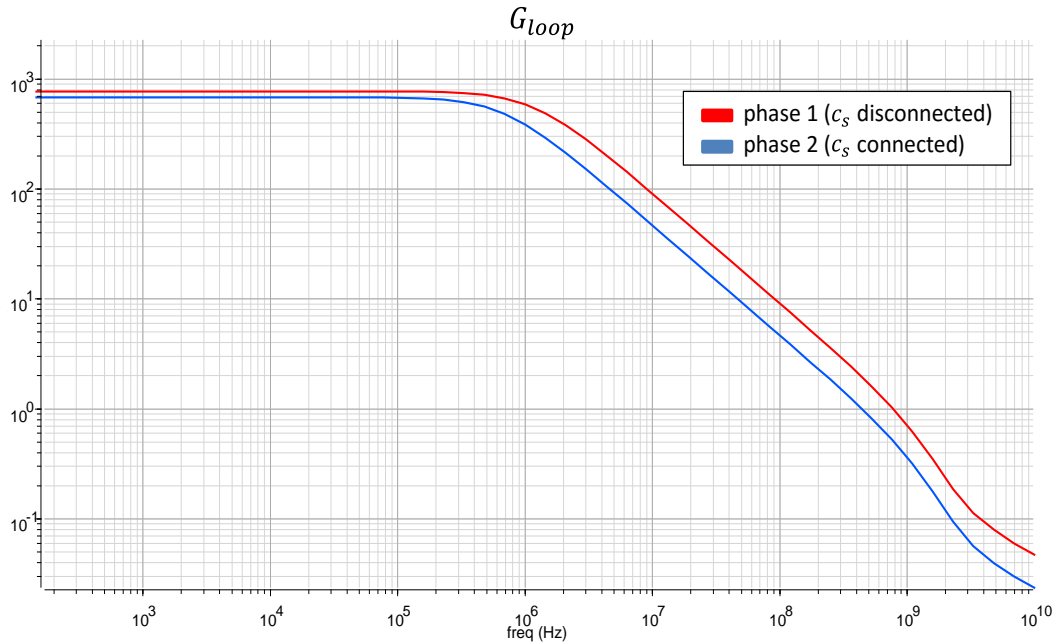
Figure 6.7 Bode plot of first OTA's loop gain: comparison between phase 1 and phase 2

Table 6.4 Effect of sampling capacitance on the gain-bandwidth-product and on the OTAs compensation

		OTA1	OTA2
GBWP	With C_S (phase 2)	$443MHz$	$281MHz$
	Without C_S (phase 1)	$777MHz$	$593MHz$
Phase margin φ_m	With C_S (phase 2)	57°	67°
	Without C_S (phase 1)	34°	46°

Table 6.4 gives the simulated values of the gain-bandwidth-product and of the phase margin of the OTAs during the two phases. The effect of low phase margin on the OTA behaviour can be observed in Figure 6.8 and Figure 6.9: at the start of phase 1, the output oscillates for a few ns before settling; during this time, the comparator is in the Track phase and is therefore comparing the OTA outputs V_{INT1} and V_{INT2}: if these were to be continuously crossing each other, the correct behaviour of the loop would be compromised.

Figure 6.8 OTA switching and oscillations. The different traces correspond to different process corners; the nominal is in red.

Simulations on Cadence® Virtuoso® (specifically: pole-zero, "pz", and stability, "stb", analysis) suggested that this unwanted shift in the phase margin might be given by the combined effect of the large input transistors gate-to-drain capacitance C_{gd} and the upper cascode transistors' gate-to-source capacitance: since it is caused by parasitic components, this issue is complicated to tackle at design-level without sacrificing the amplifiers' bandwidth, and conventional techniques (such as introducing a nulling resistor in a compensated, two-stages OTA) can't be applied; the only choice was thus to assess whether the oscillations would effectively represent a problem and eventually compensate the amplifiers by increasing their capacitive load.
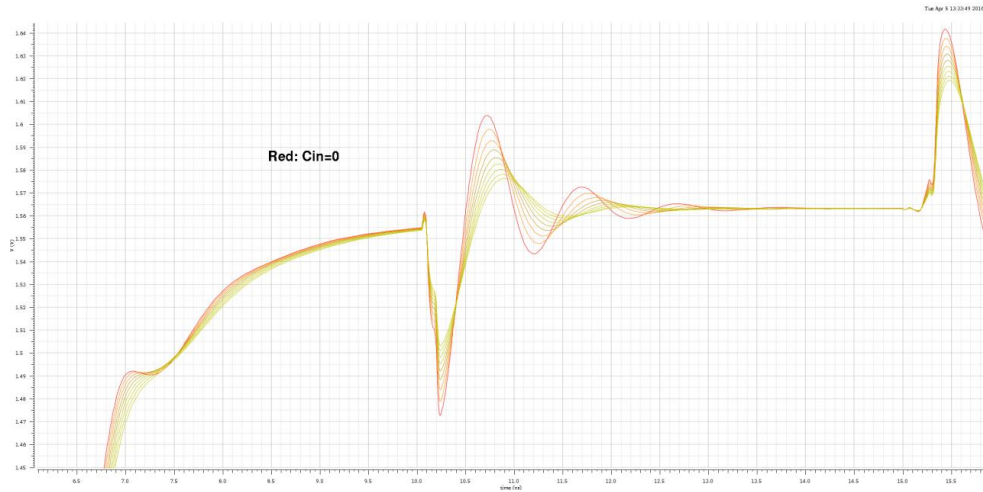
Figure 6.9 Detail of oscillations after the end of phase 2 for different values of a compensating capacitor C_{in} connected to the input. The red curve corresponds to $C_{in}=0$.

Figure 6.10 shows V_{INT1} and V_{INT2} switching in the worst case, i.e. finishing phase 1 being as close as $2mV$ (which corresponds to the required resolution of the comparator): the two waveforms do indeed cross each other, hence the comparator is not guaranteed to give the correct output. To assess the impact of the reciprocal crossing of the comparator's inputs, complete conversions of the ADC were simulated and its transfer characteristic in a defined range extracted. The effect of sporadic mistakes in the decision taken by the comparator on the overall ADC performance would be expected to be an appearance of glitches near the edges of the transfer characteristic, where noise-shaping is less efficient and each decision of the comparator has a larger weight on the output; however, no glitches were observed as a result of the simulations.
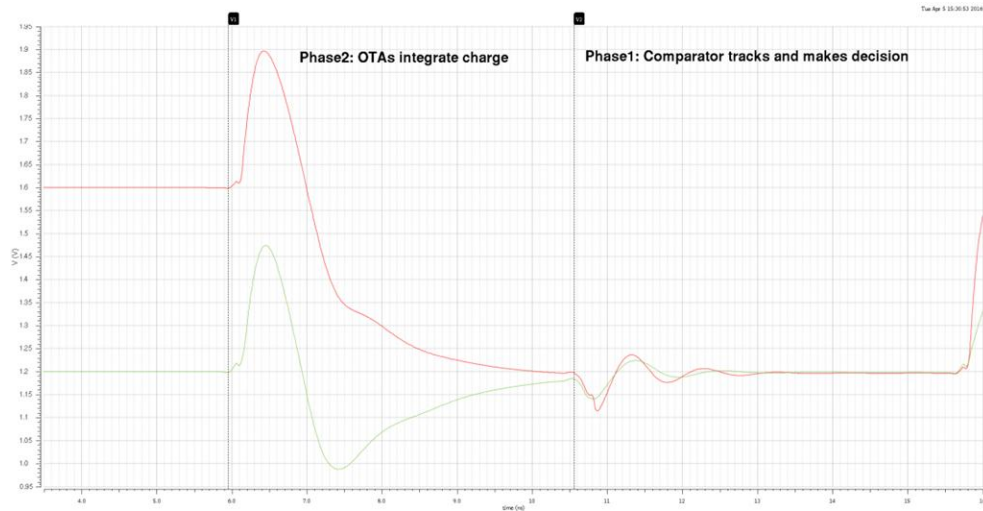
Figure 6.10 OTAs' outputs coming close together and oscillating after phase 2

To make sure that the low phase margin of the OTAs would not constitute an issue, it was ultimately decided for the test structure that, while the nominal ADC would be designed without extra load, there would be a separate group of ADCs which would have

compensating capacitors connected to the OTAs inverting input, of values of $20fF$ and $10fF$ for the first and second OTA respectively (see Chapter 7.4).

6.4.4   Impact of charge injection and clock feed-through

Both switched capacitors stages make use nMOS switches only, hence no counter-measure against charge injection and clock feed-through is taken apart from employing clocks with delayed phases. The reason why it was preferred to avoid using other types of switches such as transfer gates is that these would need additional clocks (which would be Phi1, Phi1d, Phi2, Phi2d inverted, so as to have non-overlapping '1's), which would make the routing channel wider (see Chapter 7.3), thus reducing the effective space available.

These two spurious mechanisms will therefore degrade the ADC integral linearity by introducing an input-dependent offset. Figure 6.11 shows the simulated input offset of the first integrator stage as a function of its input, while in Chapter 8.1.1 the overall INL degradation of the ADC due to the switches is discussed.
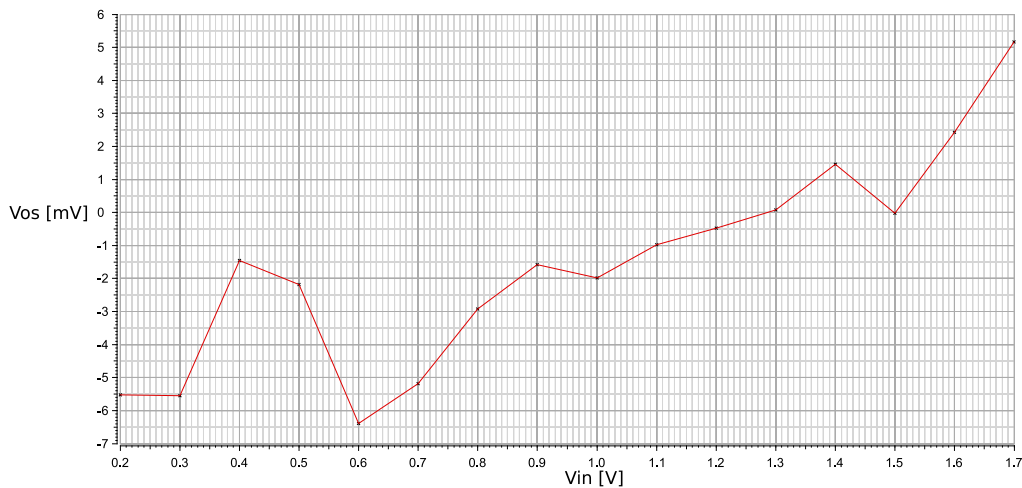
Figure 6.11 Measured input offset of integrator's first stage as a function of its input (all nodes except for the input were kept at the common mode).

The degradation of the ADC's INL is however not a concern given the ADC application. The offset introduced by the switches can be a problem if its spread is large. Monte Carlo simulations were run to assess variation of the ADC offset with and without the contribution of charge injection; the two values extracted for the standard deviation were $7.94mV$ when charge injection was included and $7.35mV$ when it was excluded; these very similar results confirm that the offset spread is mainly given by mismatched transistor pairs in the OTA.

6.4.5   Spread of second integrator's gain

Monte Carlo simulations were run to check that the spread of the second stage's integrator gain g_2 (mainly caused by capacitors mismatch) would not constitute a problem. Behavioural simulations with Simulink® showed appreciable difference in the ADC performance for variations of g_2 of approximately 10% from its nominal value: the results from the Monte Carlo simulation, shown in Figure 6.12, prove that the actual spread - roughly equal to 0.4% - is more than acceptable.
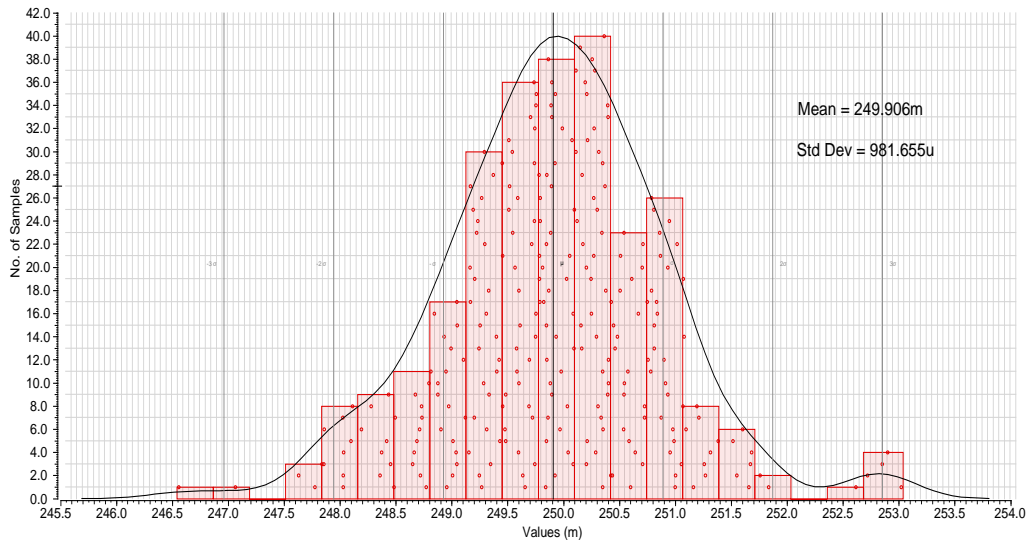
Figure 6.12 Statistics of measured second stage's integrator gain g_2 from Monte Carlo simulations.

6.5   Comparator/DAC

6.5.1   Overview

The clocked nature of the Incremental Sigma Delta demands that the comparator should be a clocked comparator, i.e. reset at every cycle. Since it has to compare the integrators' outputs during phase 1, it makes sense that the reset is performed during phase 2. This makes it necessary to memorize the result of the comparison of phase 1 and buffer it before the comparator is reset. In our case, as shown in Figure 6.13, the buffer is a flip flop supplied between $V_{DAC}^L = 0$ and VDD_D2=V_{DAC}^H. The flip flop was not custom designed: instead, a standard cell from the foundry's fast logic library (i.e. employing thin oxide transistors) was used to ensure good reliability and yield. Inside this flip flop, however, there is an inverter driven by the input: since the outputs of the comparator are reset at every cycle at $\sim V_{DD}/2$, if one of these were directly connected to the flip flop it would cause cross-conduction current in the inverter (since both its transistors would be on), which must be avoided since it would drastically degrade the power consumption and would increase the risk of cross-talk between ADCs. For this reason, a NOR gate was placed in the chain: its output, connected to the flip flop input, is only allowed to be different from 0 after the comparator has made a decision and its outputs are restored to the full scale. Cross-conduction power dissipation from the flip flop is thus avoided at all times.

The signal which disables the NOR forcing its output to ground is also the same that clocks the flip flop, and it is the complement of the signal that scans the Decide phase of the comparator (explained in Section 6.5.3).
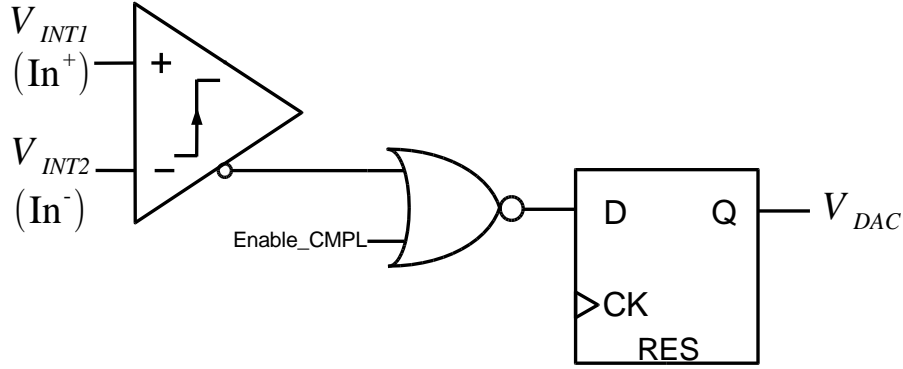
Figure 6.13 Comparator and DAC buffers

6.5.2 Architecture

In general, the DAC reference voltages can either be distributed at chip level or connected to buffers inside each column. While it can be challenging to realize a global buffer with the current capacity necessary to drive many ADCs, resorting to the latter option will give the problem of spread in both V_{DAC}^H and V_{DAC}^L (due to the spread in the column-level offset), thus it will produce offset (if the delivered voltages are different from V_{DAC}^H and V_{DAC}^L but their difference FSR is preserved) and gain (if the delivered voltages are offset by different amounts, thus changing the ADC FSR) pattern noise. Some works (see, for example, [27]) propose to use in-column buffers while using a continuous calibration running in background. In this work, one of the aims was to realize an ADC which didn't need any calibration, therefore the global reference option was preferred; moreover, in order to avoid the implementation of a large on-chip buffer, it was decided that the DAC reference voltages could be delivered directly by a supply line, and that a level-shifter (see Figure 6.14) would be used to increase the range from $V_{DD\_18} = 1.8V$ to $V_{DAC}^H = 2V$.
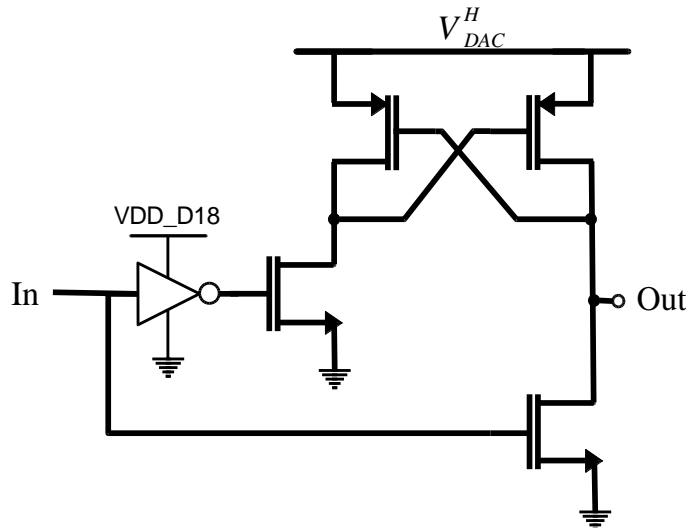
Figure 6.14 Structure of a level-shifter from VDD_D18 to V_{DAC}^H

The structure of the level-shifter uses a cross-coupled pair which is similar to those that implement the positive feedback in a comparator. For this reason, and considering that the DAC voltage could be delivered as a supply, it is possible to include the level-shifter/DAC in the comparator itself, as its decision stage. This stage would thus be supplied at 2V, rather than 3.3V like the rest of the modulator: this allows to use LV transistors, which are faster than HV. The only care that should be taken in adopting this solution is to avoid having the gate of a low voltage transistor driven to a high voltage by the 3.3V transistors. The resulting electric field arising in the thin oxide would be strong enough to break the dielectric, thus compromising the whole ADC. For this reason, LV transistors should only be connected to HV transistors in a pull-down configuration, preferably featuring nMOS.

Two possible architectures to implement the block in object are shown in Figure 6.15, using nMOS and pMOS inputs respectively: in both cases, only HV nMOS are connected to LV transistors. The difference in the input voltages leads to a difference in the currents generated in each branch and hence a difference in the outputs $\Delta V_{out} = \Delta I \cdot R_{pull}$. This is then amplified by the positive feedback, generating the outputs V_{DAC}^H and V_{DAC}^L.
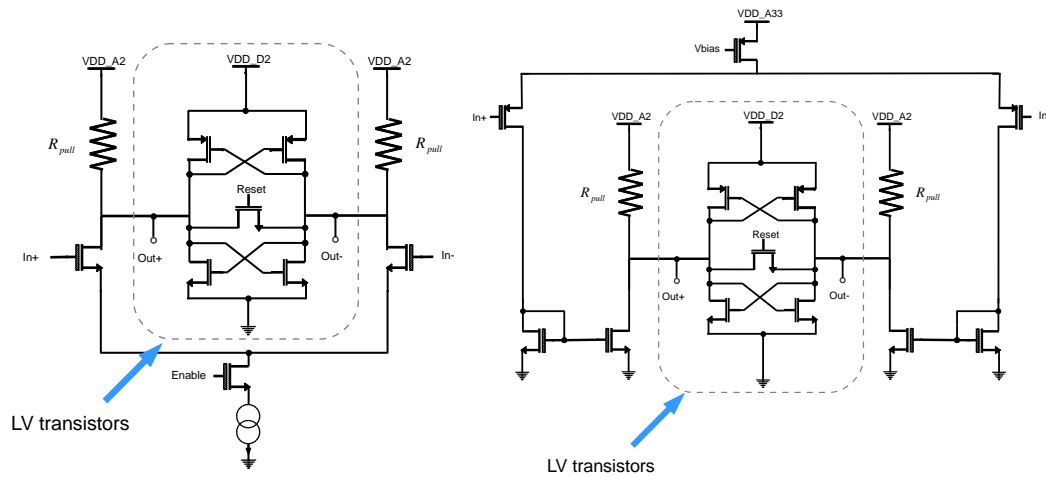
Figure 6.15 Two ways of connecting the LV transistors to HV transistors

Of the two alternatives, the one featuring nMOS inputs was discarded because of the concern that rapidly moving inputs might still drive the outputs connected to LV transistors to a high voltage through capacitive coupling. The further separation between input and output stage introduced in the other option eliminates this possibility. Furthermore, the introduction of a mirroring gain (achievable by making the inner transistors of the mirror wider than the outer ones) would help meet the resolution specification. However, this feature was not exploited, and the mirror ratio was set at 1: this allows the widths to be kept as small as possible in order to have the shortest settling time for a given bias current.

Given the clocked operation of the comparator, it was possible to further modify the structure in Figure 6.15 (right), getting rid of R_{pull} to implement a dynamic push-pull stage. This resistance sets a trade-off between resolution and speed, since one would want the quantity $\Delta I \cdot R_{pull}$ to be large for resolution purposes while the time constant $R_{pull}C_{out}$ should be small for high speed. In a push-pull stage instead, the outputs diverge during the

inputs' comparison, thus the desired resolution can be achieved with a small current, provided the comparison time is long enough.

The final configuration of the comparator is therefore shown in Figure 6.16.

As done for the OTAs, the dimensions of the transistors were chosen above the minimum for yield purposes: a minimum length of $0.6\mu m$ was set for HV transistors and $0.4\mu m$ for LV transistors. All transistors' widths were kept at the minimum allowed by this argument: exception is made only for the input transistors, which have a width of $2\mu m$, chosen to give a meaningful load capacitance to the integrators rather than for resolution purposes, to contain the under-damping of V_{INT1} and V_{INT2} which is explained in Section 6.4.3.

Table 6.5 Transistors' sizes in the comparator

Input pair (HV)	2/0.6
HV nMOS	0.6/0.6
LV transistors (all)	0.42/0.4
Bias pMOS (HV)	0.6/0.6
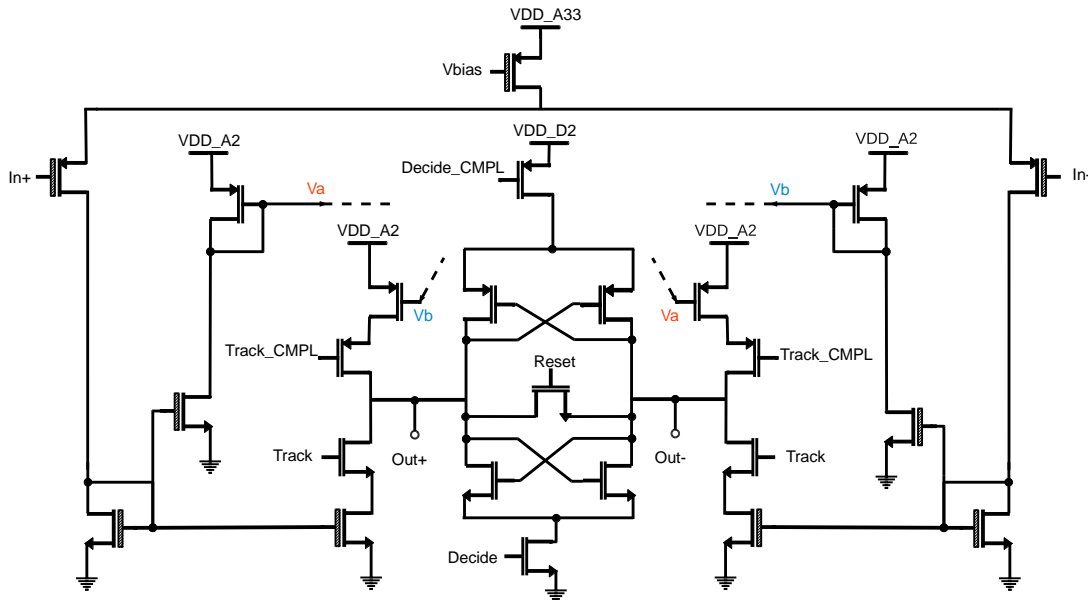
Figure 6.16 Comparator's configuration. Nodes Va and Vb are connected to the node with the same name. Signals Track_CMPL and Decide_CMPL are the inversion of Track and Decide, respectively

6.5.3   Operation

The comparator works in three phases: Reset, Track and Decide, timed as shown in Figure 6.17.
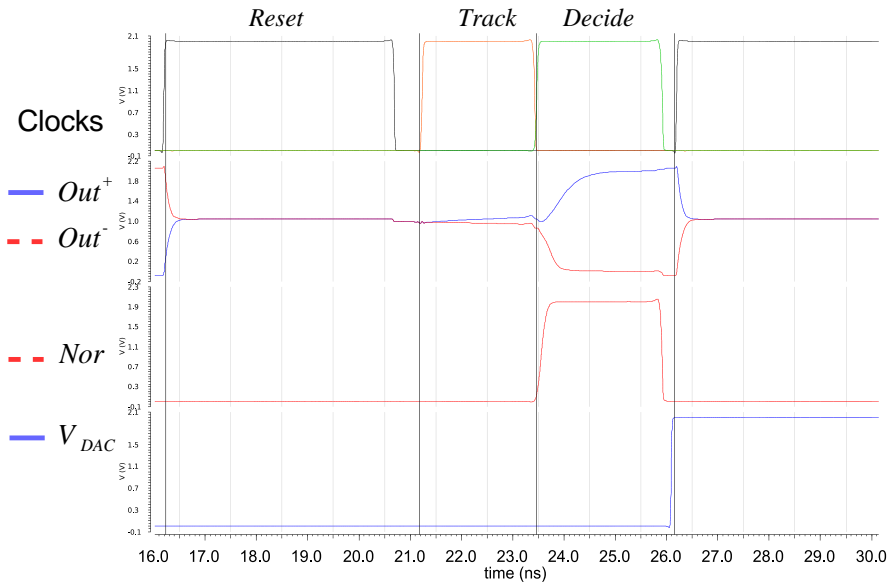
Figure 6.17 Phases and operation of the comparator

- *Reset* ($T_{cycle}/2 = 5ns$): a switch short-circuits the outputs, which are floating, together: the final voltage is ~$V_{DD}/2$. Since the comparator needs to make its decision during Phase1 (when the OTAs outputs are stable), the Reset phase corresponds to Phase2.
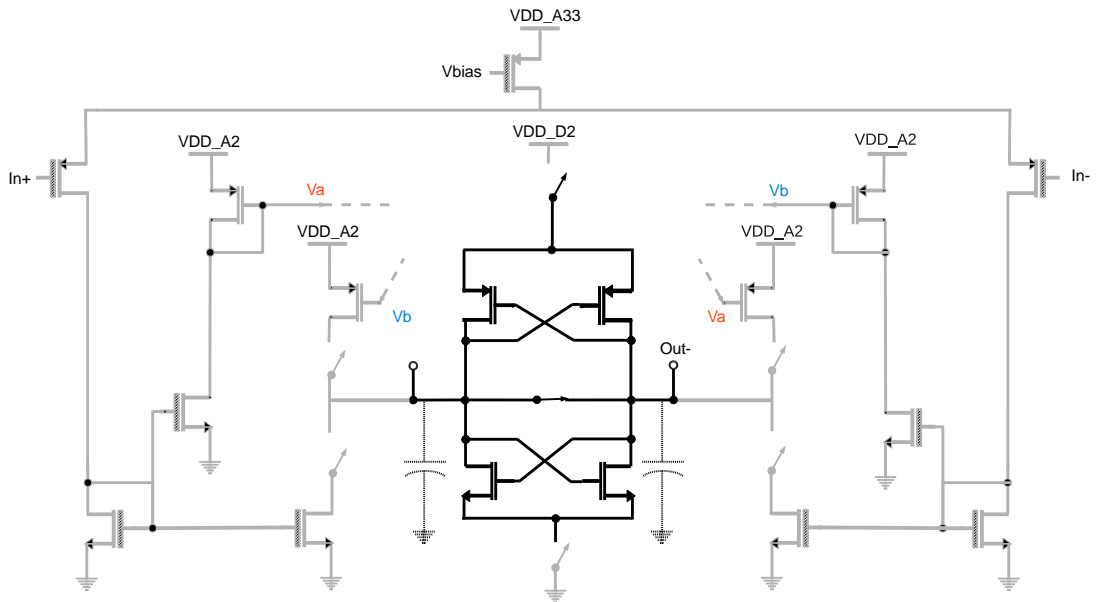
Figure 6.18 Reset phase

- *Track* ($T_{cycle}/2 = 2.5ns$): the cross-coupled inverters are still disconnected from supply, so the outputs show high impedance; the push-pull stage is hence connected to the outputs and lets their voltages diverge according to the difference of the inputs.
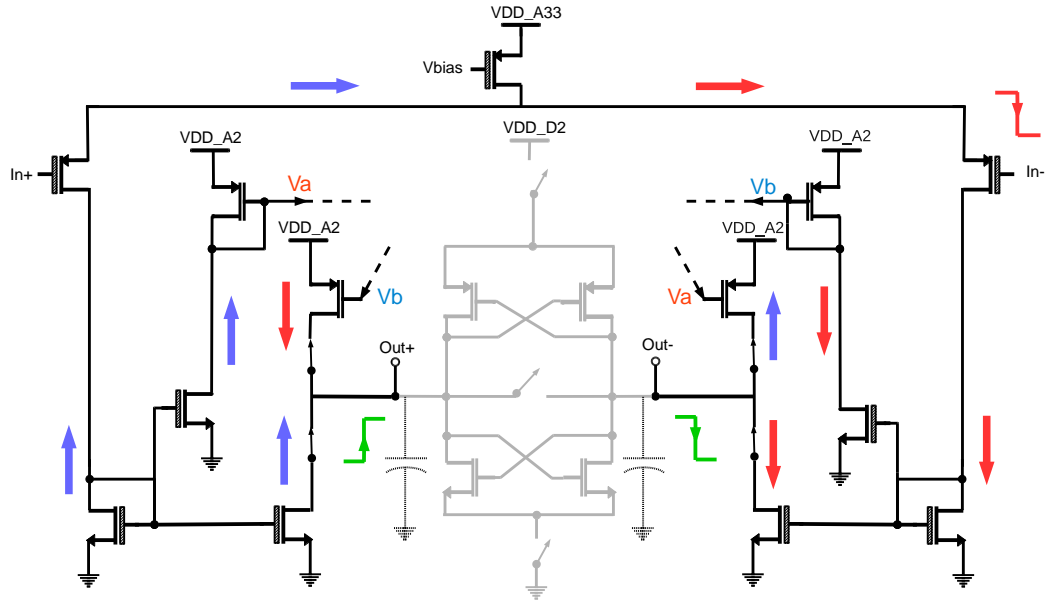
Figure 6.19 Track phase

- *Decide* ($T_{cycle}/2 = 2.5ns$): the cross-coupled inverters are now connected to supply, and all other switches are open: the positive feedback restores the outputs to V_{DD} and ground, respectively. At the end of this phase, signal the flip flop after the comparator is triggered, and the DAC voltage is thus fed to the modulator's input at the beginning of phase 1.
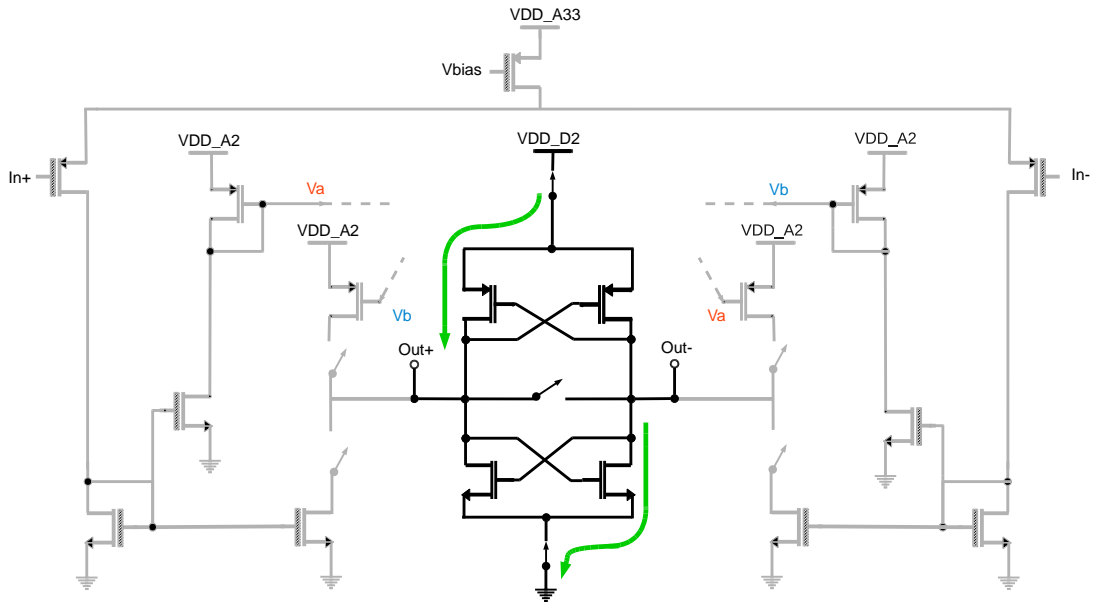
Figure 6.20 Reset phase

6.5.4 Power consumption and simulated performance

The current drawn by each branch is $2.75\mu A$, for a total of $I_{Comp} = 11\mu A$. This was the minimum current that ensured quick enough restoring of the internal voltages after one

branch passed from being turned off at one cycle (i.e. the corresponding input pMOS had its gate high enough to turn it off) to suddenly being turned on at the following cycle.

Resolution and offset were extracted with Monte Carlo simulations where the switching points of the comparator for rising and falling input were compared as shown in Figure 6.21. The hysteresis was measured to be small enough to meet the specification of $2mV$ without the need for a further increase of the bias current (it was in fact lower than $250\mu V$ - which was the minimum measurable with the setup used for the simulation - for all Monte Carlo iterations). The extracted standard deviation of the offset is $20mV$, i.e. 5 times smaller than the specification of $100mV$: thus, even in a chip hosting thousands of ADCs, the probability that a comparator will affect the performance is extremely small.
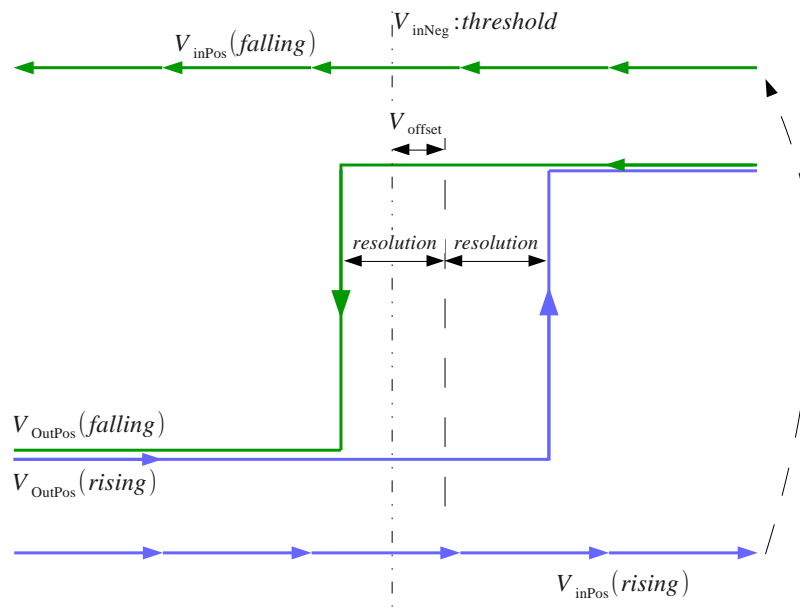
Figure 6.21 Diagram of measurement of comparator's offset and resolution

Despite there not being a specification for its contribution to noise, transient noise simulations were run to have rough assessment of the comparator's performance in this sense. The input difference was kept at $2mV$, and the number of incorrect decisions made was counted. The probability of failure in this condition was simulated to be 0.4%: considering that the comparator noise also undergoes a 2nd order noise-shaping, this was more than enough to consider that the noise performance of the ADC will not be affected by the comparator. This was subsequently confirmed by system-level noise simulations, which will be presented in Chapter 7.

Chapter 7

Digital design and layout

After having examined the design of the analogue modulator in Chapter 6, in this chapter the rest of the system developed is examined. We will first look at the design of the digital sections of the ADC, starting with the synthesis of the clocks and then discussing the realisation of the digital filter. Then, some considerations about the layout of the ADC will be done and lastly, the core architecture of the test chip will briefly be reviewed.

7.1   Timing signals generator

The operation of the modulator is scanned by several clocks running at $100MHz$: four 3.3V clock signals - Phi1 (Φ_1), Phi1d (Φ_{1d}), Phi2 (Φ_2), Phi2d (Φ_{2d}) – control the integrators' switches as shown in Figure 6.2; five 2V signals - Comp_Reset, Comp_Track, Comp_Decide and their complements Comp_DecideCMPL, Comp_TrackCMP - scan the phases of the comparator. In addition to these, reset signals at both 3.3V and 2V are necessary, and they need to be synchronized with the rest of the clocks. This section deals with the generation of these signals.

The timing signals generator block has two inputs: a clock running at 200MHz and an external asynchronous reset. The clock works at twice the frequency of operation of the system: this allows to easily generate signals with duty-cycles of 25% and 75% like that of Comp_Track, Comp_Decide and their inverted equivalents; the following section will explain how this is done. In addition to synthesizing the signals, the main clock serves to synchronize the external reset with the rest of the circuit.

In the design, particular attention was posed to the synchronization between 2V and 3.3V signals, which are driven by transistors of different oxide thicknesses (thin oxide transistors are supplied at 2V while thick oxide transistors at 3.3V) and are therefore generated at different paths.

The cells composing the timing signals generator are for the most part standard logic gates which are provided by one of the foundry's libraries.

The output signals of the block are listed in Table 7.1. Currently, the block only relies on an external reset, whereas in future versions it will be able to automatically generate an end-of-conversion reset signal every M cycles. To generate this signal, it is necessary to implement

a counter similar to the one necessary for the digital filter of the SD. However, in the test chip, the reset will be provided externally, which allows to vary the number of cycles and thus change the bit depth.

Table 7.1 Outputs of the timing signals generator

3.3V supply	Sys_resetANLG	Sys_resetSYNCHED level-shifted to 3.3 V
	Phi1	Scans phase 1 of the integrators
	Phi1d	Phi1 delayed
	Phi2	Scans phase 2 of the integrators
	Phi2d	Phi2 delayed
2V Supply	Sys_resetSYNCHED	Synchronizes the external reset to the rising edge of the global clock
	Sys_resetSYNCHED_CMPL	Sys_resetSYNCHED inverted
	Comp_Reset	Scans the Reset phase of the comparator
	Comp_Track	- Scans the Track phase of the comparator - Clocks the digital filter (its rising edge corresponds to the start of phase 1)
	Comp_TrackCMPL	Track inverted
	Comp_Decide	Scans the Decide phase of the comparator
	Comp_Decide_CMPL	Decide inverted. Other functions: - it works as an inverted CLEAR signal for the NOR between the comparator and the flip flop, to avoid DC power consumption in the flip flop - when rising, it triggers the flip flop which delivers the DAC output to the first analogue stage
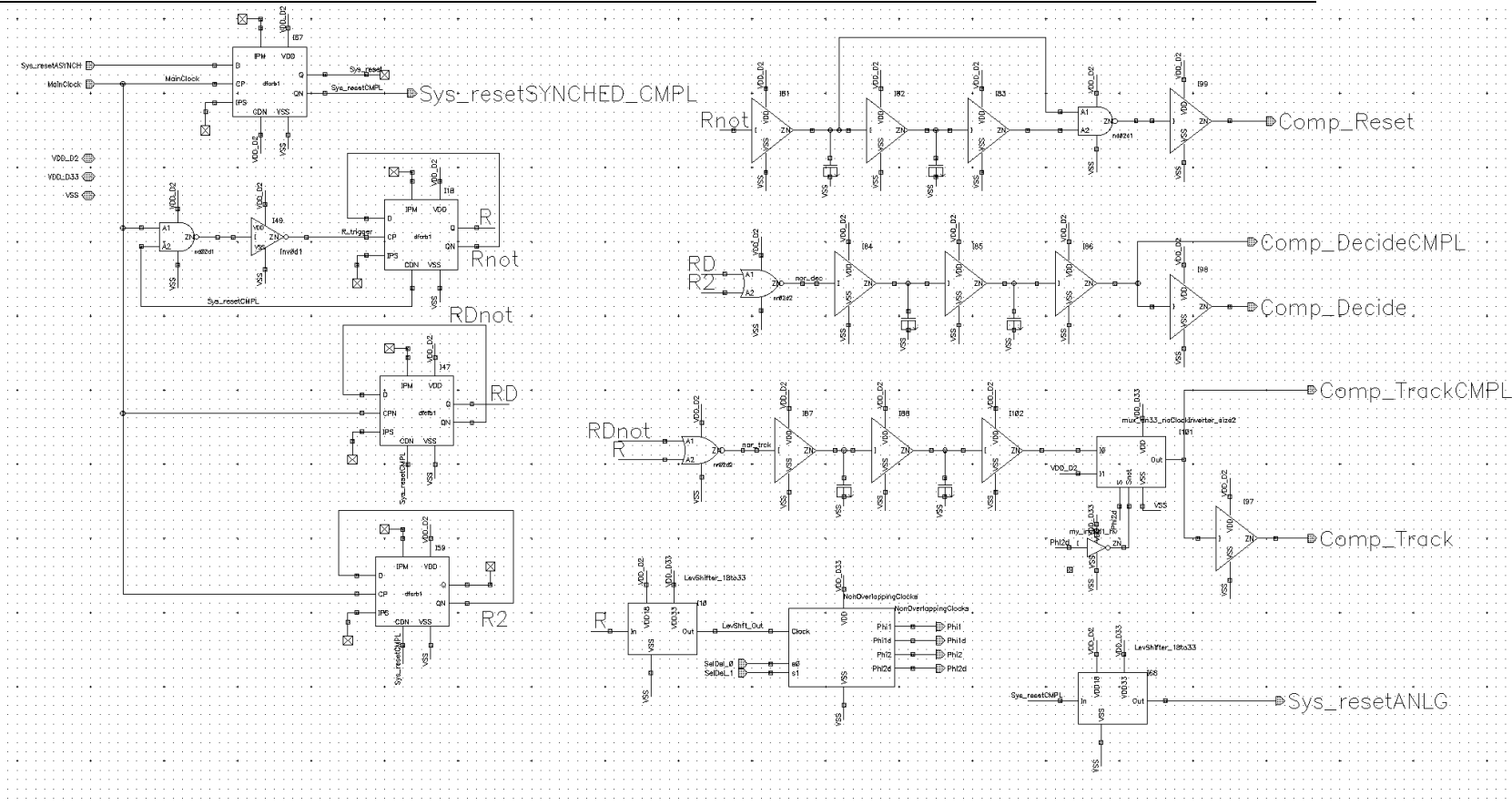
Figure 7.1 Schematic of the timing signal generator. To the left, the four flop flops that synchronize the reset and generate signals R, RD, R2 and their complements. To the bottom, R is level-shifted to 3.3V to drive the non-overlapping clocks generator with outputs Phi1, Phi1d, Phi2, Phi2d. To the right, three delay chains are used for the comparator clocks to make their phases match the delay of the non-overlapping clocks generator.

7.1.1   Signals synthesis

2V signals

The two inputs - the global clock and the asynchronous reset - are supplied at 2V.

The global clock controls four flip flops. One is a positive edge-triggered flip flop which synchronizes the external asynchronous reset with the rising edge of the clock. The other three flip flops are in the toggle configuration (see Figure 7.2), so that at every triggering edge their outputs commute: their switching frequency is hence half that of the main clock and equal to the oversampling frequency of the $\Sigma\Delta$, i.e. $100MHz$. Their three outputs are R, RD and R2; unlike R and R2, RD is generated by a negative edge-triggered flip flop as seen in Figure 7.2, and it is thus shifted with respect to the others by 1/4 of their period: signals with 25% or 75% duty-cycle can then easily be generated by exploiting the partial overlap of these signals. Moreover, the T_{CQ} delay (the time between the rising edge of the clock and the switching of the output) of all flip flops is almost identical, which means that the relative phases of the signals are well controlled.
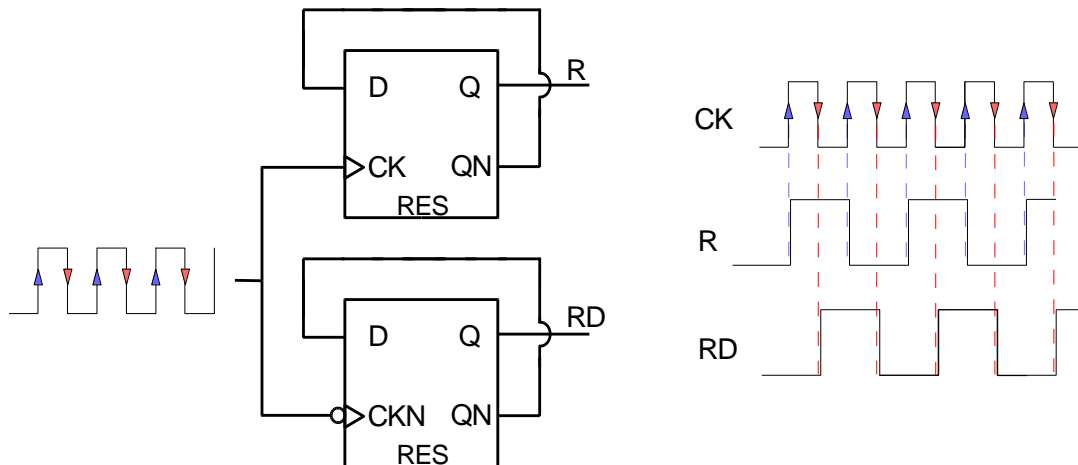
Figure 7.2 Toggle flip flops and their output waveforms

R2 is just like R except that, when reset is on, R2='1' and R='0'. R2 is used for the generation of Comp_Decide and Comp_Decide_CMPL, and it forces them respectively to '0' and '1' during the global reset, thus ensuring that the output of the NOR gate after the comparator is pulled to ground and the following flip flop does not introduce DC power dissipation (refer to Figure 6.13, where signal Enable_CMPL is Comp_Decide_CMPL).

Table 7.2 and Figure 7.3 show how the 2V outputs of the block are generated using R, R2 and RD (in the table, $\bar{a}$ denotes logical negation of a). Note that $\overline{RD}$ is given by the negative output of the flip flop which generates RD, hence no delay is added.
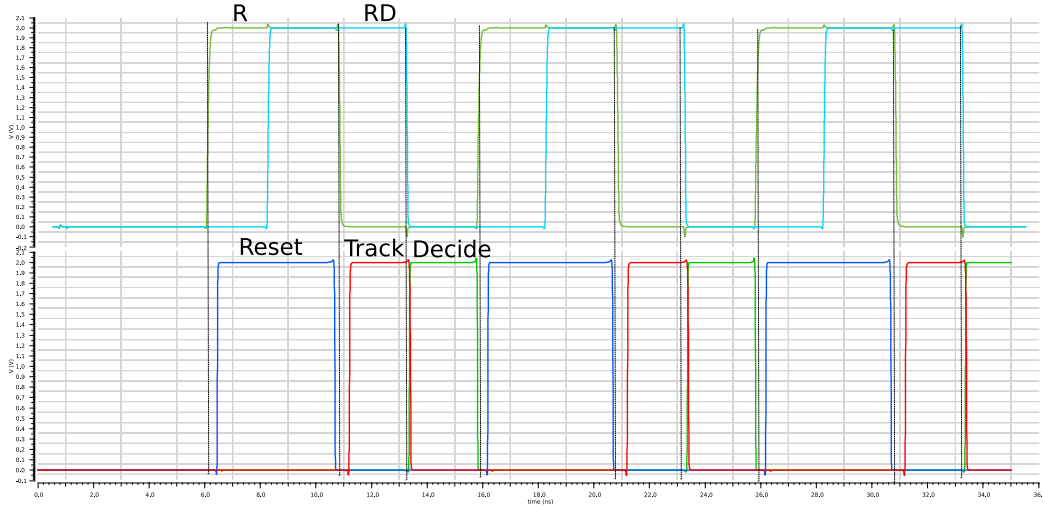
Figure 7.3 Generation of Reset, Track and Decide signals using phase-shifted signals R and RD.

Table 7.2 Logical synthesis of the comparator's clocks

$Comp\_Reset = R$ (with reduced duty-cycle)	
$Comp\_Track = \overline{R + \overline{RD}}$ (gated by Phi2)	$Comp\_Track\_CMPL = \overline{Comp\_Track}$
$Comp\_Decide = \overline{R2 + \overline{RD}}$	$Comp\_Decide\_CMPL = \overline{Comp\_Decide}$

3.3V signals

The 3.3V outputs of the block are Sys_resetANLG (obtained simply by level-shifting the synchronized reset to 3.3V), which at every end of conversion resets the capacitors C_{INT1} and C_{INT2}, and signals Phi1, Phi1d, Phi2 and Phi2d, which scan phase 1 and phase 2 of the integrators, obtained by level-shifting R and feeding it to a non-overlapping clocks generator. Nominally, Phi1d and Phi2d are like Phi1 and Phi2 but delayed to reduce the distortion due to charge-injection and clock feed-through (see Chapter 5.6). In order to be able to test experimentally whether the use of delayed clocks had a significant influence on non-linear charge injection and clock feed-through at the frequency of operation, the non-overlapping clock generator can select the relative delay between Phi1 and Phi1d (and their counterparts). To this purpose, the block is composed of two stages:

- Core (see Figure 7.4): generates signals p1, p1d_A, p1d_B and their complementary p2, p2d_A, p2d_B.
- Multiplexer (see Figure 7.5): receives two select signals s0 and s1 and, using transfer gates, outputs the signals Phi1, Phi1d, Phi2, Phi2d according to the four options summarized in Table 7.3.
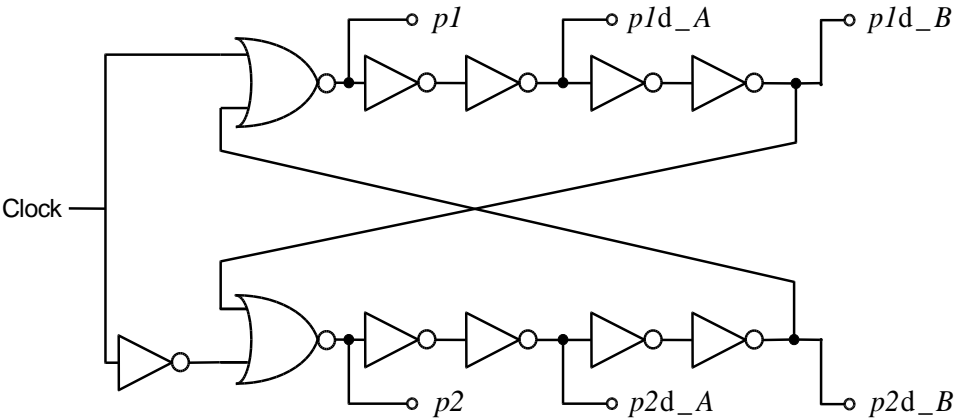
Figure 7.4 Non overlapping clock generator - core
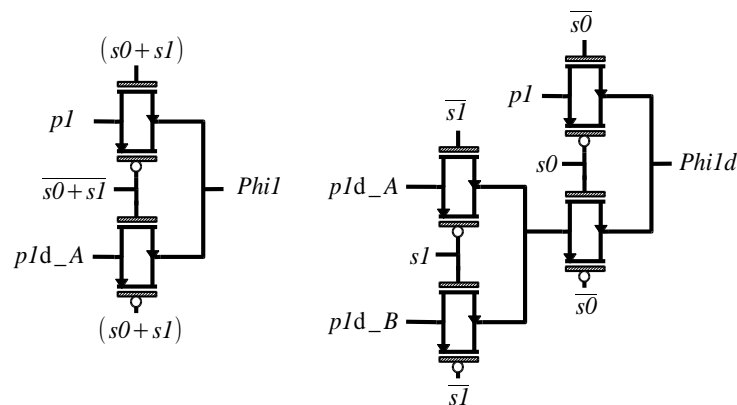
Figure 7.5 Non-overlapping clocks generator: multiplexer stage for Phi1 and Phi1d

Table 7.3 Four possible combinations for Phi1 (Phi2) and Phi1d (Phi2d)

s1 s0	Phi1 (Phi2)	Phi1d (Phi2d)	Configuration
0 0	p1d_A (p2d_A)	p1 (p2)	Swapped Phi-Phid
0 1	p1 (p2)	p1 (p2)	Phi1==Phi1d
1 0	p1 (p2)	p1d_A (p2d_A)	Nominal Delay $\sim 170ps$
1 1	p1 (p2)	p1d_B (p2d_B)	Delay $\sim 270\, ps$
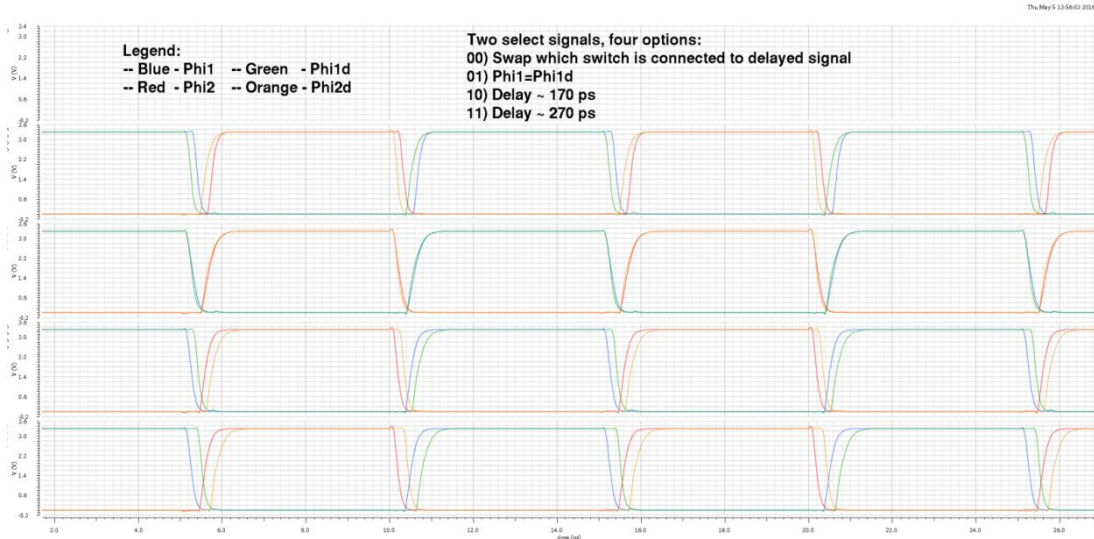
Figure 7.6 Signals Phi1, Phi1d, Phi2, Phi2d in the four selectable options

Synchronization

The most important potential timing issues are:

- Overlap of the comparator's Reset phase with Decide. Signal Comp_DecideCMPL triggers the flip flop responsible for delivering the DAC to the input: if the outputs of the comparator are reset before the triggering (at least $280ps$ earlier than the triggering to be precise, which is the setup time of the flip flop), the incorrect DAC voltage could systematically be fed to the input. In order to avoid any overlap of the Reset phase with either Track or Decide, Comp_Reset has a duty cycle slightly lower than 50%: this duty-cycle shrinkage was obtained by delaying R and using a AND gate as depicted in Figure 7.8. Furthermore, Monte Carlo simulations were run to make sure that the variability of the edges' position in time would be small enough not to constitute a problem.
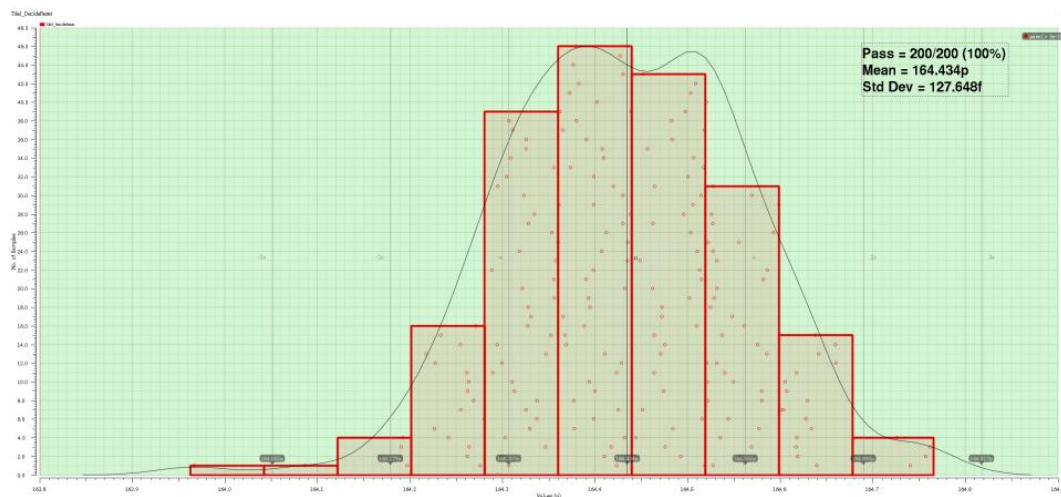
Figure 7.7 Monte Carlo simulation measuring the delay between the falling edge of Decide signal and rising edge of Reset. The extracted standard deviation, 128fs, is several orders of magnitude lower than the delay between the edges.
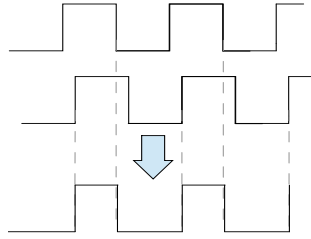
Figure 7.8 Shrinking the duty cycle with a delay and a AND gate

- The non-overlapping clocks generator synthesizing the 3.3V signals has some delay: in order to compensate for this and match the timing of signals at different supplies, the 2Vsignals are delayed using a chain of inverters slowed by MOS capacitors. This delay naturally has some variability due to process variations: this could become an issue if it caused the Track phase of the comparator to start early with respect to the end of Phi2, since the integrator outputs might not have settled. This problem was eliminated at design level by gating Comp_Track and its complement with a 3.3V-transistors transfer gate controlled by Phi2d.

Transition from reset state to running conversion

The signals are arranged so that during global reset (Sys_resetSYNCHED = '1') Phi1=Phi1d='1',Phi2=Phi2d='0' and at the end of the reset phase Phi1 switches to '0' and Phi2 to '1'. This ensures that the reset phase coincides with phase 1 of the first conversion cycle of the ΣΔ, and no time is wasted between the end of a conversion and the start of the following one. Moreover, Comp_Decide_CMPL is forced to '1' during global reset – whereas normally it would be 0 during the second half of phase 1. This is done in order to force to ground the output of the NOR it is connected to, thus avoiding cross-conduction power consumption in the DAC flip flop at all times (see Chapter 6.5.1).

7.2   Decimator

The block diagram of the implemented decimator is shown in Figure 7.9. As already stated in Chapter 4.5.1, the output is coded with 17 bits, so that an ENOB of up to approximately 16 bits can be obtained.

An appealing option for the implementation of the logic was to use dynamic gates. Dynamic gates use capacitors to store the digital information rather than switches permanently connected to V_{DD} or ground and for this reason they don't need a pull-up network, resulting in reduced area occupation and – in the case of dynamic flip flops – reduced capacitive load for the clock. This type of logic is avoided in low frequency circuits, since leakage currents would slowly discharge the capacitor and thus risk to corrupt the information stored. A rough estimation of a high enough threshold for leakage not to matter for this technology node gives $f = 10kHz$ ($100\mu s$ are needed to cause a decrease of $1V$ on a $10fF$ capacitor with $1pA$ leakage current), several orders of magnitude lower than our case of $100MHz$. This type of logic could therefore be used in the decimator- and has indeed been employed by other designers, see [33]; however, it also needs careful design and thorough simulations to make sure that the circuit is robust against other degrading phenomena, e.g. clock feed-through (described - although for a different context - in Chapter 5.6).

V_{in} — Modulator

d

1st stage

2nd stage

HA_FB_0 — $D[0]$ HA_FB_0 — $D_{out}[0]$

$c_1[0]$ $D[1]$ $c_2[0]$ $D_{out}[1]$

HA_FB_1 — $D[1]$ FA_FB_1 — $D_{out}[1]$

$c_1[1]$ $D[2]$ $c_2[1]$ $D_{out}[2]$

HA_FB_2 — $D[2]$ FA_FB_2 — $D_{out}[2]$

$c_1[2]$ $D[3]$ $c_2[2]$ $D_{out}[3]$

HA_FB_3 — $D[3]$ FA_FB_3 — $D_{out}[3]$

$c_1[3]$ $D[4]$ $c_2[3]$ $D_{out}[4]$

HA_FB_4 — $D[4]$ FA_FB_4 — $D_{out}[4]$

$c_1[4]$ $D[5]$ $c_2[4]$ $D_{out}[5]$

HA_FB_5 — $D[5]$ FA_FB_5 — $D_{out}[5]$

$c_1[5]$ $D[6]$ $c_2[5]$ $D_{out}[6]$

HA_FB_6 — $D[6]$ FA_FB_6 — $D_{out}[6]$

$c_1[6]$ $D[7]$ $c_2[6]$ $D_{out}[7]$

HA_FB_7 — $D[7]$ FA_FB_7 — $D_{out}[7]$

$c_1[7]$ $D[8]$ $c_2[7]$ $D_{out}[8]$

HA_FB_8 — $D[8]$ FA_FB_8 — $D_{out}[8]$

$c_2[8]$

HA_FB_9 — $D_{out}[9]$

$c_2[9]$

HA_FB_10 — $D_{out}[10]$

$c_2[10]$

HA_FB_11 — $D_{out}[11]$

$c_2[11]$

HA_FB_12 — $D_{out}[12]$

$c_2[12]$

HA_FB_13 — $D_{out}[13]$

$c_2[13]$

HA_FB_14 — $D_{out}[14]$

$c_2[14]$

HA_FB_15 — $D_{out}[15]$

$c_2[15]$

HA_FB_16 — $D_{out}[16]$

HA_FB_X

$c_1[x-1]$

HA — $D[x]$

$c_1[x]$

Q   D

CK

RES

FA_FB_X

$c_2[x-1]$

$D[x]$ → FA → $D_{out}[x]$
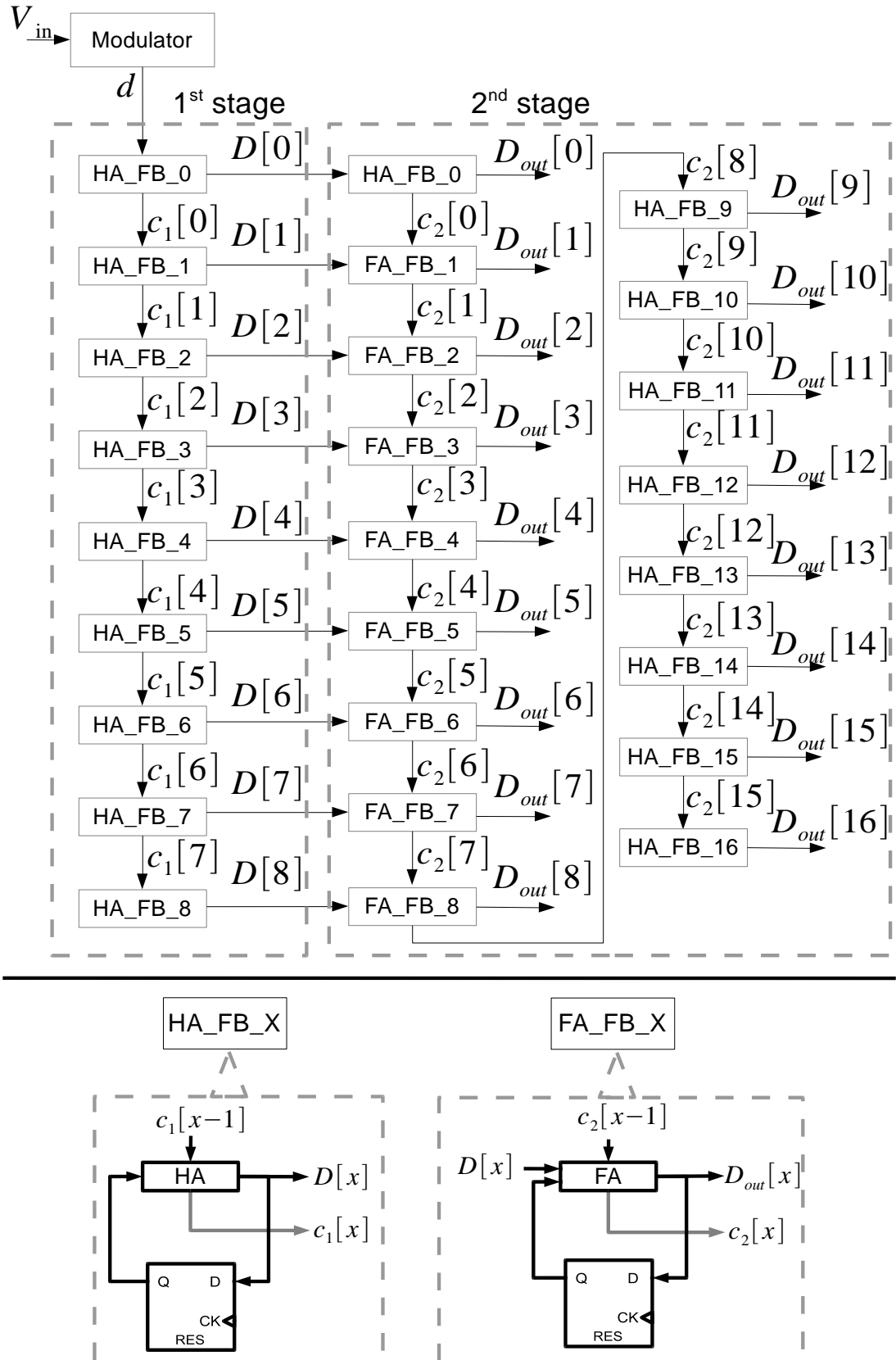
$c_2[x]$

Q   D

CK

RES

Figure 7.9 Decimator block diagram (top) and detail of adder blocks (bottom) . HA denotes a half adder, FA a full adder.

113

In the system developed for this project it was decided that static gates would be used, since they are in general more reliable and high priority was given to avoiding all risks of malfunctioning in the digital section. Moreover, a library of static IP logic gates was available, which allowed to use the CAD tool of Cadence® Encounter® to automatically perform the synthesis and layout of the filter starting from a Verilog® script which described the functional behaviour of the filter. Dynamic gates remain an interesting improvement to the design, which might be included in future developments if the interest in further area and power reduction is strong.

7.3   Layout and dimensions

Except for the decimator, all blocks in the ADC – both digital and analogue – are full custom layouts. The main dimensions are summarised in Table 7.4, while Figure 7.14 shows the floor plan of the ADC (except the digital filter).

The timing signals generator was included in the structure of the ADC and placed between the modulator and the decimator. This choice, which would be appealing in a complete sensor since it would simplify the global routing and eliminate the problem of out-of-phase signals caused by spread in the delays, is however only temporary. It was made to ensure that all ADCs were well timed and that the failure of one generator would not compromise the whole chip. In future developments, the block will need to be excluded (or at most be shared by many ADCs), as its large power consumption - caused by the high frequency of operation and a design not optimized from this point of view – is unsuitable for a large sensor.

The ADCs on the chip have a column-parallel disposition, with the pitch set to 15 μm and a total resulting length of $640\mu m$: the area specification set in Chapter 2.4 is therefore met, especially if the length of the timing signal block – which will be taken out of the ADC structure in further developments – is subtracted (the total length would thus be $\sim 540\mu m$) and if we consider that the final version of the ADC will not include the timing signals generator and will have a 12-bits output rather than 17.
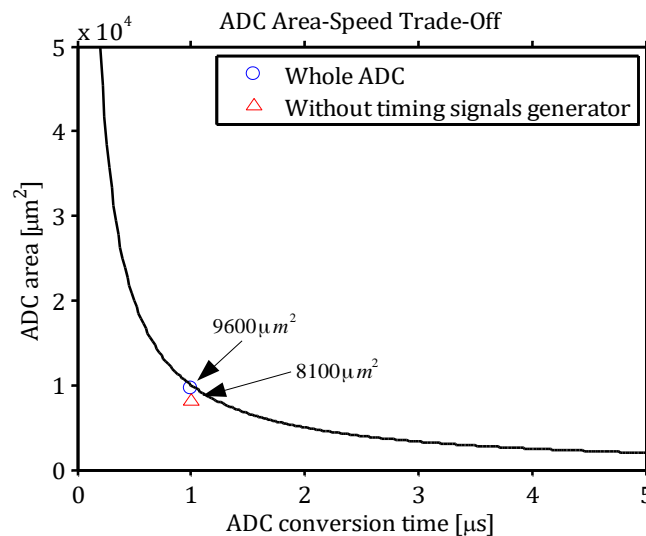

Figure 7.10 ADC area specification and position of developed ADC (including all blocks)
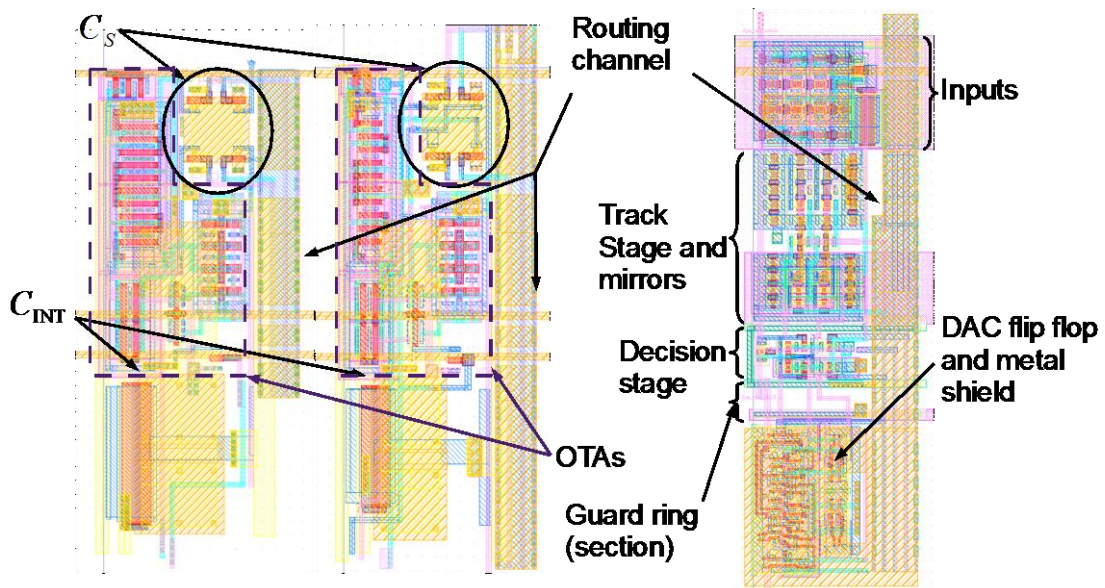
114

Figure 7.11 Layout of the three modulator's blocks. From left to right, shown is the first switched capacitor stage, the second switched capacitor stage and the comparator with buffers
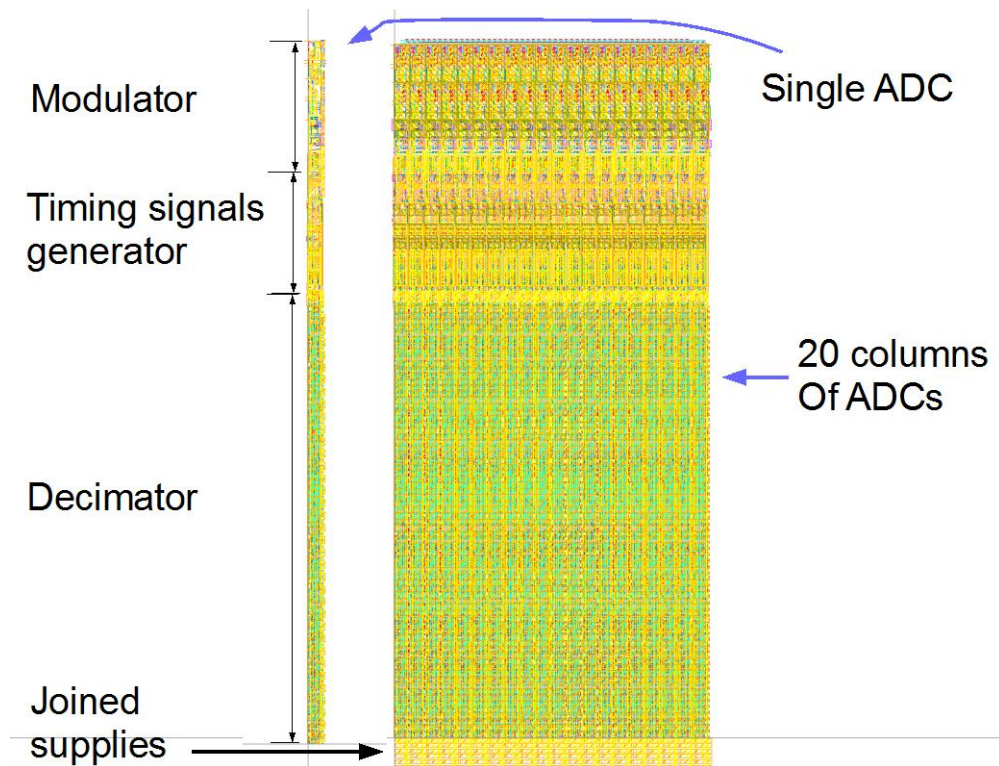
Figure 7.12 ADC top view. Single ADC (left) and 20 column-parallel ADCs (right)

Pitch	15 μm
Modulator - Stage 1 (OTA1)	34 μm
Modulator - Stage 2 (OTA2)	35 μm
Modulator - Stage 3 (Comparator and buffers)	46.5 μm
Timing signals generator	98 μm
Decimator	410 μm

In order to limit their coupling to the analogue signals, the clocks were distributed inside a shielded routing channel (its cross-section is shown in Figure 7.13), which ran beside the analogue blocks, thus further limiting the space available. The shield is realised with metal layers and vias, whose connection to the digital ground filters most of the coupling of the digital signals to the external surroundings.

The supplies are routed vertically on the top metal and, in order to limit their interaction with the digital gates, a metal shield connected to digital ground (VSS_D) lies below the supply lines.
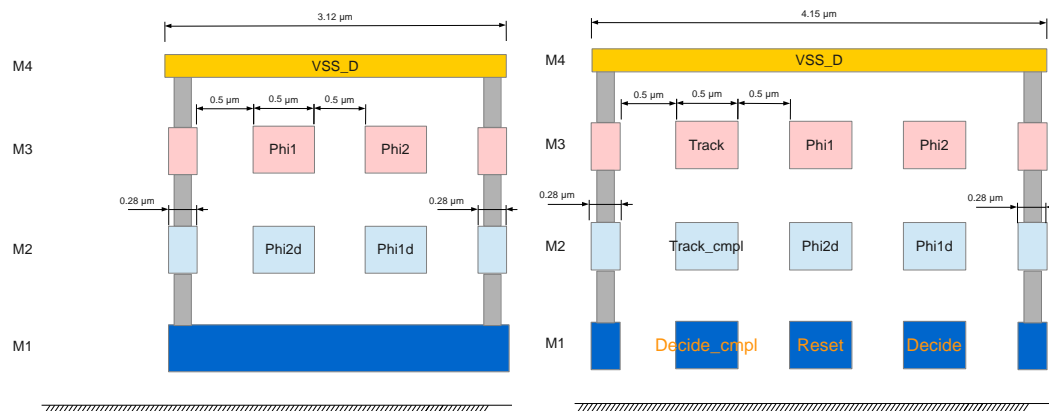
Figure 7.13 Routing channel cross section. Beside the OTAs (left) and beside the comparator (right)

Non-uniformities in the properties of the substrate (such as doping) can cause mismatches in nominally identical pairs – such as the input transistors of an OTA or its mirror transistors – thus introducing offset and spread. This effect is partially due to a gradient in the substrate properties and partially due to their random distribution. To limit the contribution arising from the gradient, the common centroid technique shown in Figure 7.15 was adopted: each transistor is decomposed into four smaller transistors, and the symmetry in the placement is such that the average overall effect of non-uniformities' gradient is the same for both transistors of the pair.
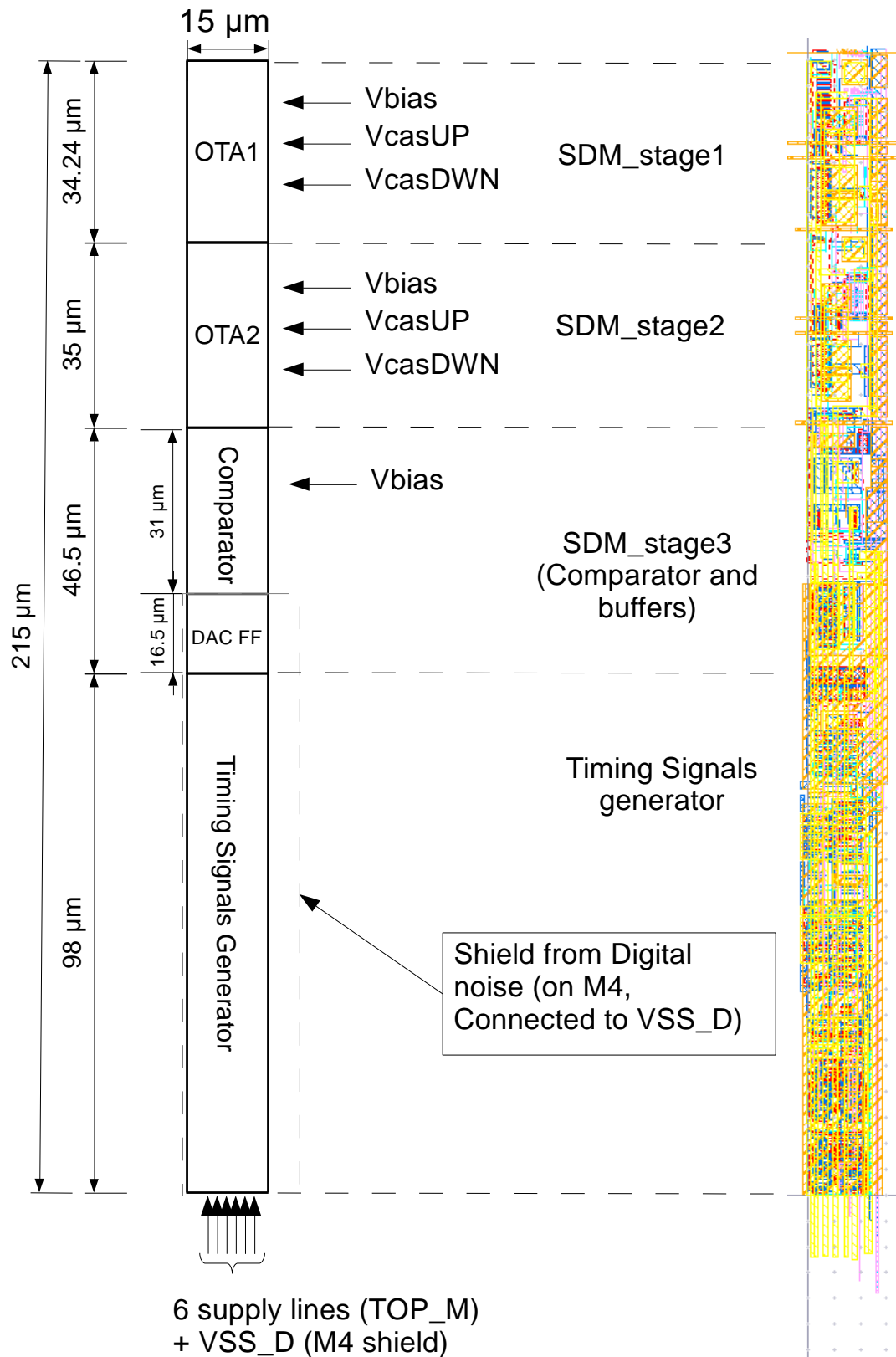
Figure 7.14 Diagram (left) and layout (right) of modulator and timing signals generator's blocks and dimensions – dimensions scaled
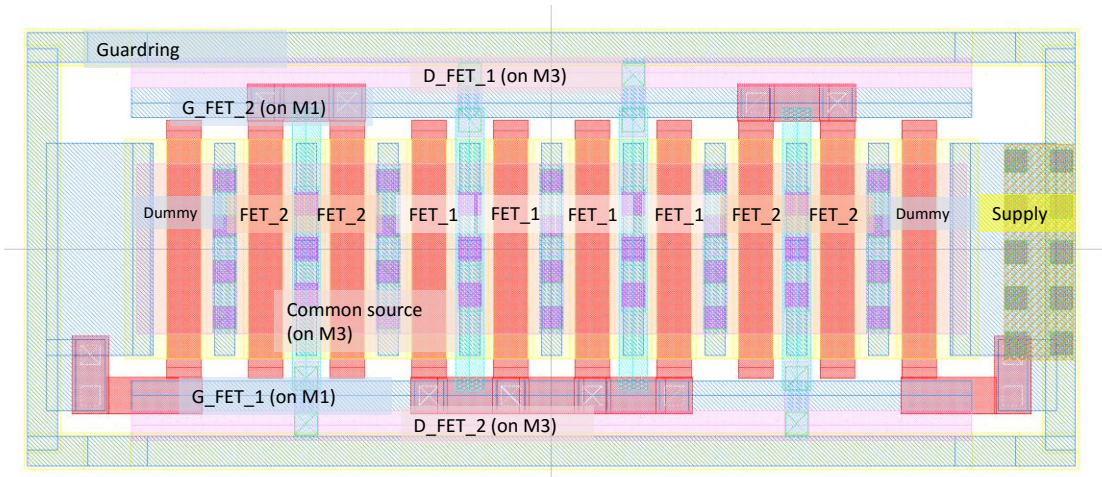
Figure 7.15 Common centroid of type "AABBBBAA" top view

In a mixed signal circuit the fast switching activity of the digital gates involves a displacement of charge in the substrate which could reach the analogue section and thus affect its performance. This has to be avoided, especially for the operation of the comparator, whose outputs are at high impedance during the Track phase. To prevent this, two guard rings – one connected to the p-doped substrate and one connected to an n-well, as shown in Figure 7.16 - separate the analogue section from the digital: any charge flowing towards analogue components will drift following the electric field between the two arrays and thus be recollected.
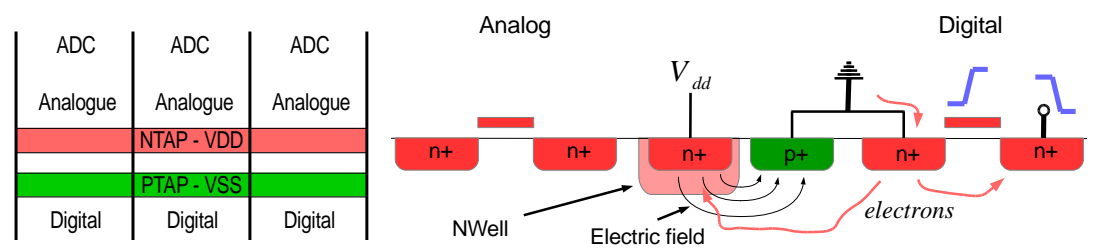
Figure 7.16 Two arrays of substrate contacts filter the substrate noise towards analogue transistors. Top view (left) and cross-section (right)

7.4   Splits

For testing purposes, the ADCs hosted on the chip are not all identical: they have been divided in 8 groups of 20 ADCs, each group differing from the others for the value of some parameter (e.g. a capacitance) or some process-related design choice. This subdivision in "splits" was done to be able to understand through testing how the deviation of some parameters from the nominal value would impact the transfer characteristic.

The components that were subject to change in the splits, shown in Table 7.5, include (referring to Figure 6.2):

- Integrator capacitance of the 2nd stage C_{INT2}: changing this parameter means changing the integrator gain g_2, which – as discussed in Chapter 4.5.2 and 4.5.1 – directly influences the modulator's stability. Apart from the nominal value of 0.25, in some splits

g_2 will have the values 0.2, 0.33, 0.5. According to the trends shown in Figure 4.14, a particularly bad performance is expected for $g_2 = 0.5$.

- Sampling capacitance of the 1st stage C_{S1}: this capacitance affects two relevant quantities:
  o White input noise, for which – according to Eq. (5.12) - high capacitance is preferred.
  o The gain of the 1st integrator g_1: as explained in Chapter 4.5, this parameter acts as a scaling factor for the signals which ideally has no effect on the conversion; however, for large values of g_1 the integrators' output voltages will occasionally exceed the OTAs' output range, thus drastically decreasing the DC gain of the amplifiers. For this reason, integrators with g_1 higher than the nominal are expected to have a larger quantization noise. It will be possible to discriminate this effect from other potential sources of SQNR degradation (such as hidden spread in other parameters) by decreasing the OTA currents (so that the overdrive voltages of the transistors decrease and the OTAs' output range increases, according to Eq. (6.4)(6.5)), modifying the cascode bias voltages accordingly and working at a slower frequency than nominal (so that the subsequent worsening in bandwidth doesn't affect the performance).
- Extra load capacitances C_{load1} (1st stage) and C_{load2} (2nd stage). These capacitances have been implemented as MOS capacitors in accumulation and are connected to the negative input of the amplifiers in order to increase their phase margin, which – as explained in Chapter 6.4.3 - is lower than 45° during phase 1, thus possibly hindering the correct operation of the comparator and consequently causing a degradation of the overall ADC performance. The extra added load will also decrease the slew-rate of the OTAs and could hence also increase the distortion. Note that, from a layout point of view, MOS capacitors don't occupy any extra area, since they can lie below MIM capacitors C_{INT1} and C_{INT2}.
- Use of "special capacitors" – available with the technologic process used, the details of which can't be revealed in this dissertation - to be used in place of the nominal MIM capacitors C_{S1}, C_{S2}, C_{INT1}, C_{INT2} for increased area density.

Table 7.5 List of all splits included in the test chip

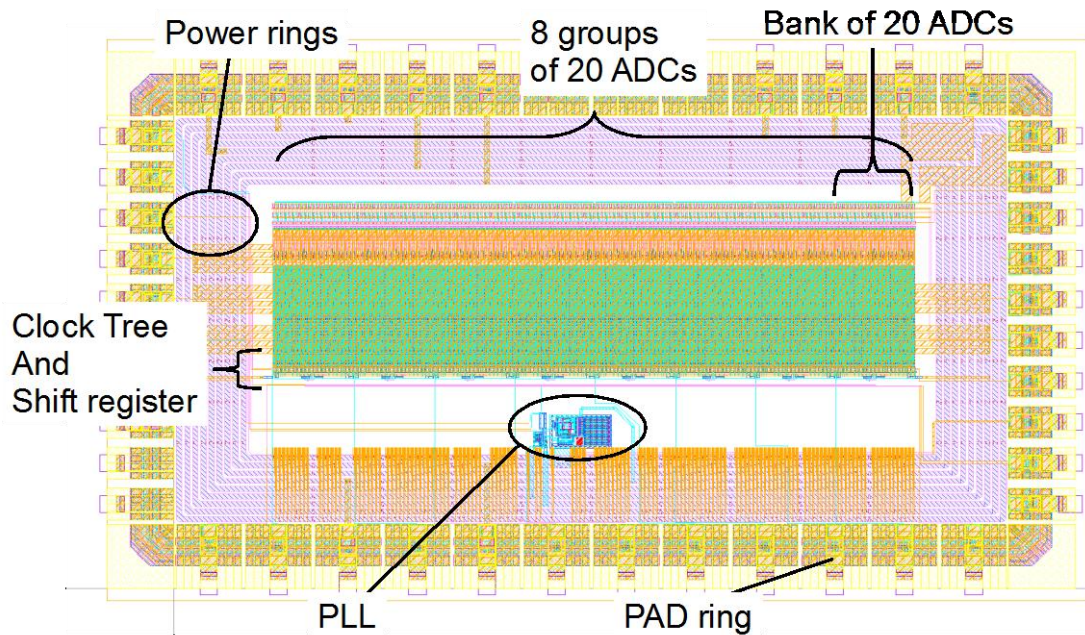| Parameter	C_{INT2}	C_{S1}	C_{load1} , C_{load2}
Influences…	g_2 →Stability	- g_1 → Signals' magnitude - Noise	- Phase margin - Settling
1 (nominal)	$120fF$	$30fF$	0,0
2	$60fF$	$30fF$	0,0
3	$90fF$	$30fF$	0,0
4	$150fF$	$30fF$	0,0
5	$120fF$	$45fF$	0,0
6	$120fF$	$60fF$	0,0
7	$120fF$	$30fF$	$20fF, 10fF$
8	Special capacitors (values as nominal)		

7.5   Test chip core architecture

Figure 7.17 Test chip layout top view

7.5.1   Clocks generation

The $200MHz$ clock necessary for the ADC operation will be internally generated starting from a slow clock at $12.5MHz$ fed to a phase-locked loop (PLL) which multiplies its frequency by 16 (note that the PLL block was not developed in this project). There is also the possibility to use a "backup" clock externally generated at $200MHz$; a multiplexer controlled by an external signal decides which of the two clocks will reach the ADCs.

7.5.2   Clocks distribution

Given the reasonably small number of ADCs (160) and dimensions of the chip, the relative phase-shift of the travelling clocks shouldn't be significant; in any case, clock trees were laid out to distribute the $200MHz$ clock to the ADC, the global reset and the shift register signals used for readout (see following section).

7.5.3   Readout

In order not to compromise the speed of the image sensor it will be hosted in, the ADC should be able to perform a new conversion while data from the old conversion is read out, i.e. conversion and readout should be pipelined – as described in Chapter 2.2.1. To this purpose, a shift register was placed beside the second stage of the decimator: once one conversion is completed, the 17 output bits are copied to the shift register and a new conversion can start immediately.

In the case of the test chip developed, however, this feature will not be exploited, since it would require an output rate too high to be handled by the testing apparatus: the 17 bits of all ADCs would have to be read out in $1\mu s$, thus the minimum readout frequency would need to be $17MHz \cdot n_{ADCs}/n_{parallel} = 340MHz$, where n_{ADCs} is the total number of ADCs on chip

(160 in our case) and $n_{parallel}$ the number of bits that can be read out of the chip at the same time (8, since there is one output pad per split).

Chapter 8

Simulated performance and future developments

The previous chapters were dedicated to the exposition of the ADC design, starting from its system-level architecture (Chapter 4) down to schematic and transistor-level design of its separate parts (Chapter 6 and Chapter 7). In this chapter, the simulated performance of the converter as a whole will be exposed, in relation to the specifications set in Chapter 2.

8.1   Non-linearity

8.1.1   INL

INL was measured considering the deviation from an end-point line (a straight line connecting the two extremes of the transfer characteristic).

The result obtained, INL=18.75 LSBs, is within specification (0.5% of the total FSR, or 20 LSBs) but remarkably worse than what was predicted with behavioural simulations on Simulink® (see Figure 4.14). Figure 8.1 clearly shows why: the figure plots INL extracted in two different cases: in case (a) the real ADC was simulated, while in case (b) the nMOS switches were replaced by ideal relays; the resistance of the ideal switches was set to $4k\Omega$, similar to that of nMOS switches; therefore, the only major difference between the two cases was the absence of charge injection and clock feed-through effects in case (b) compared to case (a). Charge injection is thus a major cause of non-linearity; reduction of this effect could be included in future developments, but might also not be necessary given the loose linearity specifications of ADCs for image sensors.
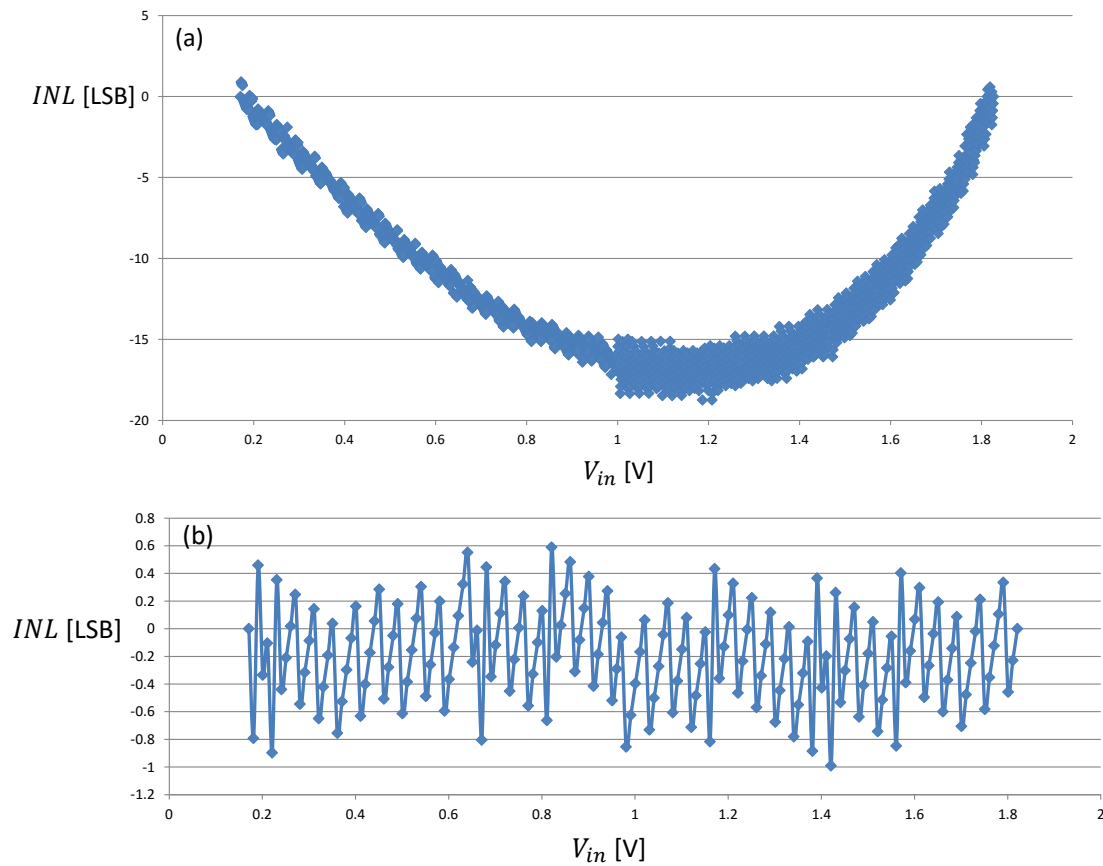
Figure 8.1 Simulated INL against ADC input. With real nMOS switches (a) and with ideal switches (b)

8.1.2 DNL

A complete assessment of DNL would require measuring the width of all the steps in the transfer curve, which would take a long simulation time. This was dropped in favour of a thorough noise performance assessment, which is described later; DNL was instead measured in restricted regions: near the edges of the input range (where overloading starts to reduce the noise shaping efficiency) and where dead zones could appear (at 1/3 and at 2/3 of the full scale, according to the behavioural simulations carried out in Simulink$^{®}$ – see Figure 8.2).
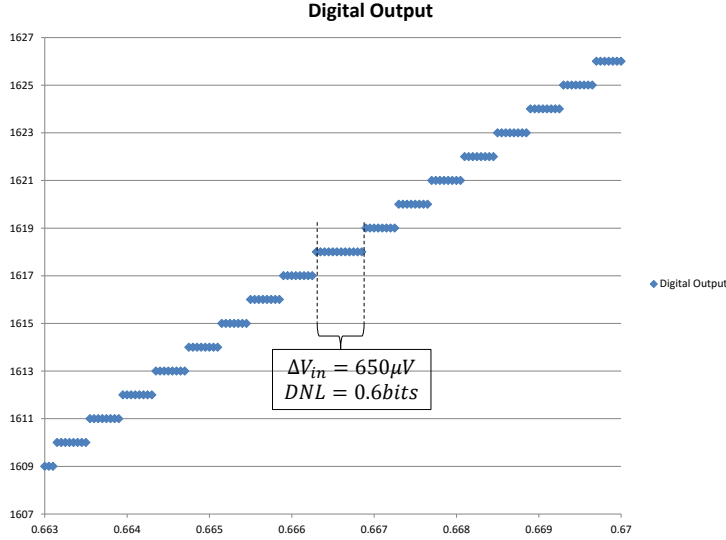
Figure 8.2 Transfer curve near the dead zone at 1/3 of the full scale. Maximum extracted DNL is where shown in figure

Results show that DNL is indeed higher in the dead zone area, with a maximum value of $0.6\ LSBs$, in good accordance with what derived from simulations on Simulink® in Chapter 4.5.3.

8.2 Noise performance

The system was designed to have an input-referred *rms* noise $\sigma_{in} < 100\mu V$, while the LSB is $404\mu V$; since $\sigma_{in} \cong LSB/4$, the noise is hardly detectable: assuming a Gaussian distribution, the recurrence of an incorrect code should be $\sim 1/16000$. Getting meaningful statistics with the nominal settings ($M = 100,\ n_{bits} = 13$) would hence require a very large amount of time and simulations. To get around this problem it was decided to exploit the fact that, changing the number of cycles, the LSB decreases more quickly than σ_{in}, hence for M large enough it will be $\sigma_{in} > LSB$ and noise will more often manifest itself, making the measurements significantly easier. In particular, assuming that the main noise contribution is white and recalling Eqs. (3.15) and (4.10), we have:

$$\frac{\sigma_{in}(M)}{LSB(M)} \propto M^{3/2} \qquad \textbf{(8.1)}$$

In order to be able to use very large values of M, the code for an "upgraded" digital filter with 21-bits output was scripted in Verilog®, thus allowing to simulate conversions with M up to 2000 cycles to be simulated.

The graphs in Figure 8.3 show how output and input referred noise change with M. From those graphs – especially looking at the logarithmic plot - two conclusions can be drawn:

- It is confirmed that the dominant noise contribution is white: the slope m of the straight line extrapolated from the log-log plot is in fact $m \sim -0.5$, thus suggesting that $\sigma_{in} \propto 1/\sqrt{M}$. If $1/f$ noise were significantly affecting the measurements, its

impact would increase with M, hence making the trend line more horizontal: the measured slope would thus be, in absolute value, $|m| < 0.5$, which is not the case.

- By extrapolation, the input-referred noise at $M = 100$ can be estimated to be $\sigma_{in} = 79 \mu V$, thus below specification.
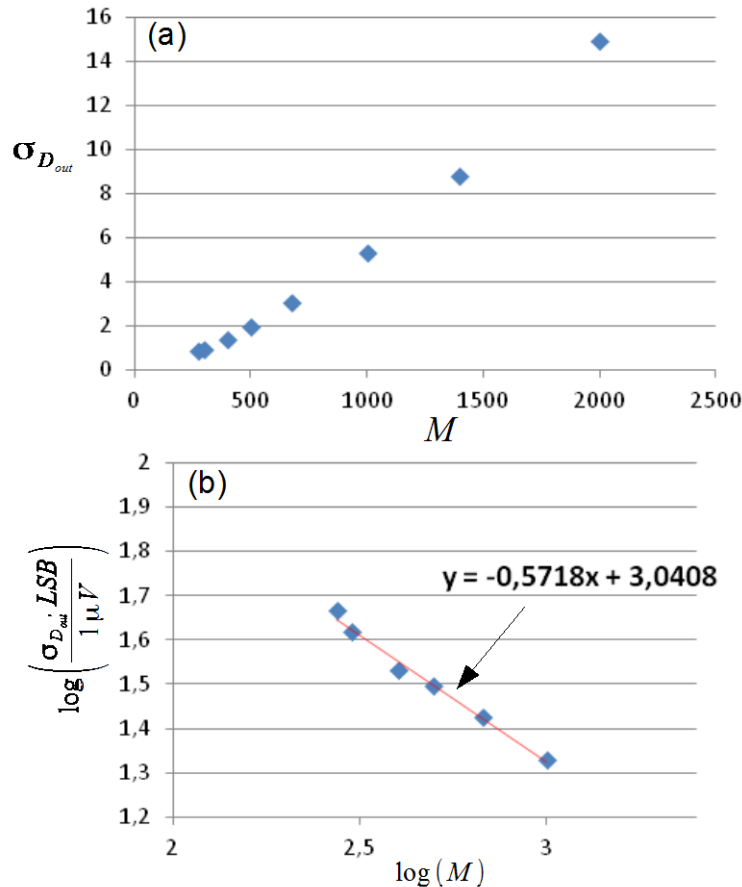
Figure 8.3 Noise performance *vs* number of cycles *M*. In (a) the plot is linear and the standard deviation is calculated on the output code; in (b) the plot is bi-logarithmic, and the noise is referred to the input

8.3   Power consumption

The power drawn by all the supply lines is summarized in Table 8.1. A distinction is made between the power drawn by VDD_D2 and VDD_D33, which are almost exclusively connected to the timing signals generator (VDD_D2 also supplies the NORs and the flip flop after the comparator, but these give a minor contribution) and the other supplies, which are only connected to the ADC. The values portrayed were obtained from simulations run including the effect of parasitics extracted from layout (which add capacitive load to the digital gates, thus increasing their power consumption).

From the results shown it is clear that the contribution of the decimator (supplied by VDD_D18) is similar to that of the analogue section, thus breaking the power consumption constraint. This is a consequence of the high speed at which the system needs to be clocked,

which causes the digital dynamic power dissipation of each gate to increase. The individual contribution to the power consumption can be estimated by [34]:

$$P_{dynamic} = V_{DD} \cdot \left(Q_{supply}\right)_{average} f_S = \alpha_{0\to1} \cdot C_{out} V_{DD}^2 \cdot f_S \qquad \textbf{(8.2)}$$

In Eq. (8.2) Q_{supply} is the average charge drawn from supply, $\alpha_{0\to1}$ is the switching activity of the gate - i.e. its probability of transitioning from '0' to '1' - C_{out} is the output capacitance (comprising of all parasitics) and f_s the frequency of operation.

The most power-hungry components amongst the digital cells were observed to be the static flip flops, due to the relatively large number of transfer gates employed – which increases the capacitive load of the clock – and the fact that each one hosts an inverter to obtain the negative clock (necessary to drive the pMOS in the transfer gate), while it would be more efficient to distribute both the positive and negative clock globally. For this reason, future versions of the ADC will surely feature dynamic flip flops in place of static - which give a lower capacitive load to the clock.

It should furthermore be noted that the outputs of the two stages of the decimator in the ADC developed for this test structure are represented with 9 bits and 17 bits respectively, whereas in the final version of the ADC they will have 7 and 12 bits respectively (the choice of increasing the number of bits of the outputs was explained in Chapter 4.5.1): because of this, the capacitive load of the clock is larger, thus the power consumption reported in Table 8.1 is an overestimation.

Table 8.1 Power consumption summary

			Power [μW]		
			Average	RMS	Peak
ADC	Analog	VDD_A33	252.78	253.11	344.52
		VDD_A2	13.4	16	80
	Digital	VDD_D18	253.8	851.4	7200
Timing signals generator		VDD_D33	838.2	1610.4	5435.1
		VDD_D2	516	764	3360
Total (ADC only)			519.98	1120.51	7624.52
Total (with Timing Signals generator)			1874.28	3509.31	21219.62

Conclusions

In this master's thesis project a column-parallel $\Sigma\Delta$ ADC for high data-rate image sensors was designed using TowerJazz® 0.18µm process.

The ADC was required to achieve 12 bits of resolution in the competitive conversion time of $1\mu s$. Other design specifications include a constraint on the maximum input noise, which had to be less than $100\mu V_{rms}$, and on the average power consumption, to be contained within $330\ \mu W$. The converter, which was laid out in a column-parallel topology with $15\mu m$ pitch, was also required to occupy an area smaller than $10000\mu m^2$ (hence its length should be smaller than $670\mu m$). Meeting this specification makes the ADC suitable to be implemented in a stacked chip in future developments, which would push further the limit of achievable frame-rate.

The specification on the number of bits was met implementing a 2nd order Incremental Sigma-Delta operating at a frequency of $100MHz$ for 100 cycles per conversion. The implemented ADC is moreover able to give an output of up to 17 bits: this will allow the use of larger oversampling ratios and thus to test how noise and other characteristics change with the number of cycles.

Despite its fast operation, the ADC size is limited, with the modulator being only $100\mu m$ long and the whole converter occupying $15\mu m \cdot 540\mu m = 8100\mu m^2$ on silicon.

The realisation of the ADC required a thorough research into the field of Sigma-Delta Modulation, with particular care for the needs of ADCs for image sensors, where noise, size and variability of the parameters must be contained. The 2nd order Incremental Sigma-Delta architecture (which uses two cascaded integrators in both the analogue section - i.e. the modulator - and the digital section - i.e. the decimator) was considered to give the best compromise between area and spread. In particular, a Cascaded Integrator Feed Forward configuration was adopted, which limits the power demand of the amplifiers compared to a Cascaded Integrator Feed Back configuration. It was moreover modified at schematic level to eliminate the capacitors normally used to implement the feed forward, thus resulting in a better exploitation of the area budget.

The design was carried out in four distinct phases: system-level behavioural simulations, analogue design of the modulator, digital design of the decimator and of the clock signals' generator and, lastly, top level design for the test structure.

Behavioural simulations with Simulink® allowed to understand and investigate the features of noise shaping and were used to compare two alternative feed forward architectures and to subsequently derive the analogue components' specifications for the design.

The modulator design featured two switched capacitor integrators and a comparator. In the switched capacitor stages, high amplifiers' gain (necessary to limit DNL) is ensured by the adoption of a cascode configuration. The sizing and biasing of these stages were set with particular attention to the noise performance, which set a lower limit to the value of the capacitors and hence to the current consumption of each stage. The comparator was designed to have a resolution lower than $2mV$, and its decision stage uses fast, thin oxide transistors whose supplies are the two DAC reference voltages, thus ensuring quick and full restoring of its outputs within the short time available. The delivery of the DAC reference voltages to the ADC through the supply has the further advantage of avoiding any spread in the references' distribution, thus potentially eliminating the need for calibration.

The digital design of the clock signals generator was carried out at schematic level and using a full custom layout. The two stages of decimator (which produce at the output a 9 bits and a 17 bits number respectively) were instead first coded in Verilog and subsequently realised using Cadence® Encounter® Digital Implementation System, which entailed a careful study of the potentials of this tool. While the timing requirements were not a challenge in the realisation of the decimator ($100MHz$ frequency of operation for the two cascaded stages is easily achievable in the technology employed), the design of the digital signals generator required careful synchronization of clocks with different duty-cycles and supplied at different voltages, with control of the spread of their relative delays. Where necessary, the risk of overlapping signals was eliminated at design level.

The finished test chip has been submitted for fabrication, and it includes the nominal ADC and 7 variations of its design, to get more insight on the impact of some parameters on its performance.

Results from simulations show that the noise specification was met: specifically, the ADC manifests only $79\mu V_{rms}$ of input equivalent noise.

The power specification was not met, due to the significant contribution of the digital blocks of the decimator at the high frequency of operation. Therefore, developments for the immediate future shall include a replacement of the static registers in the decimator with dynamic flip flops, in order to allow a reduction of the power consumption within the set limit. Testing will also allow to get a better insight on the minimum current consumption drawn by the analogue section, thus probably reducing also this contribution.

Perspectives for the medium term include a renewed layout of the ADC suitable for a stacked chip topology and the adoption of more scaled technologies to further improve the overall speed.

Appendix A.

Integrator boundaries in a first order Sigma-Delta

We shall proof that in a stable[3] first-order Sigma-Delta with DC input u, starting from some cycle n the output of the integrator v' is at any subsequent cycle $n + k$ bounded in the range:

$$u - 1 \leq v' \leq u \qquad \text{(A.1)}$$

In other terms, the set $[u - 1, u]$ is a positively invariant set for v' [11]. An immediate consequence is that, if the initial condition $v'(0)$ is within this range, then v' will always be bounded as per Eq. (A.1).

In Eq. (A.1) it was assumed that the DAC output d was binary and its output could either be $d^H = 1$ or $d^L = 0$; $v'(n)$ is the output of an integrator with unitary gain, referred to the common mode (which in this case is $(d^H + d^L)/2 = 1/2$).
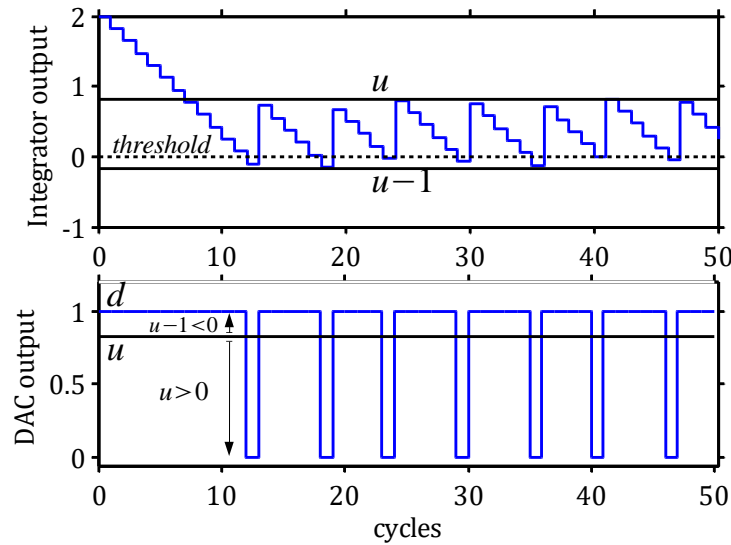

Figure A.1 Integrator's output gets locked within the range [*u-1* , *u*]. In the example, *u*=0.825

---

[3] i.e. such that $d^L < u < d^H$, as seen in Chapter 3.1.4

Let's consider the inequality $v' \leq u$: we first observe that, if $v'(0) > u$, then $v'(0) > 0$, hence according to Eq. (3.16) it will be $d(1) = d^H = 1$, and the input of the integrator at cycle 1 will be $u - d^H = u - 1 < 0$. Therefore, v' will start decreasing until it will finally be, for some cycle n, $v'(n) \leq u$ (specifically, v' will keep decreasing until $v'(n) < 0 < u$). This situation is illustrated in Figure A.1.

We now have to show that, given $v'(n) \leq u$ for some cycle n, it is impossible for v' to increase beyond u for any future cycle. In order for this to happen, in fact, it would need to be $v' < 0$ for some cycle k (so that $d(k + 1) = d^L$ and the input of the integrator at the corresponding cycle $k + 1$ will be positive) and, at the following cycle, $v'(k + 1) > u$ (otherwise, if $v'(k + 1)$ is larger than 0 but lower than u, it will certainly decrease again at cycle $k + 2$, thus we would be sure that condition (A.1) was not broken). This is however impossible, since the maximum positive excursion of the integrator is $\Delta v' = u - d^L = u$.

A similar argument can prove that $u - 1 \leq v'(n)$.

Appendix B.

Input noise of the telescopic cascode OTA

The MOSFETs in an amplifier contribute to noise: their contribution can be represented with either a series voltage generator or a parallel current generator, as shown in Figure B.1.
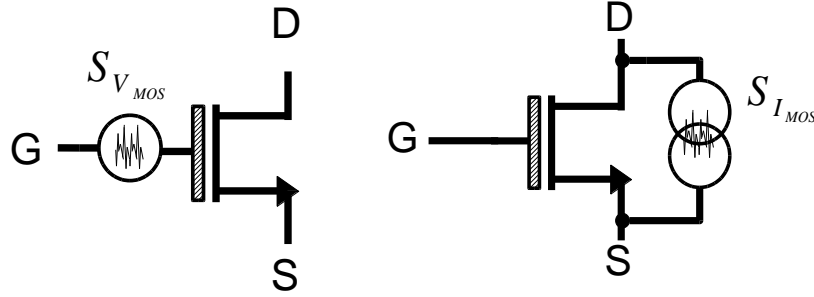

Figure B.1 Series (left) and parallel (right) equivalent noise sources of a MOSFET

The white component of noise is the channel resistance, and it can therefore be expressed using the Johnson-Nyquist theorem:

$$S_{V_{MOS}} = 4kTR_{channel} = \frac{4kT\gamma}{g_m} \tag{B.1}$$

$$S_{I_{MOS}} = \frac{4kT}{R_{channel}} = 4kT\gamma g_m \tag{B.2}$$

The factor γ takes into account that the depth of the conductive layer is not constant along the transistor's length; for a transistor in saturation, $\gamma = 2/3$ [35].

In order to perform the noise assessment, we recall the well known Norton theorem: every linear bipole which does not act as an ideal voltage source can be represented by the parallel of the resistance seen at its two terminals and a current generator of value i_{SC}, where i_{SC} is the current flowing through its terminals when short circuited. Applying the theorem at the output pin of the OTA, its output voltage can be expressed as $V_{OTA} = i_{SC}R_{out}$, and the short circuit current due to a differential voltage will be $i_{SC}^{diff} = g_{m_{IN}}v_{diff}$. The input equivalent noise voltage source of the OTA can hence be calculated as the differential input voltage that gives an output short circuit current equal to that of all noise sources.
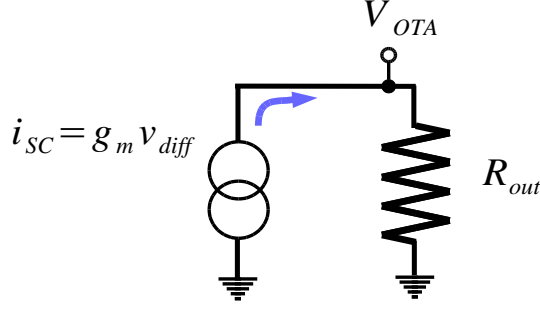
$$i_{SC} = g_m v_{diff}$$

$$V_{OTA}$$

$$R_{out}$$

Figure B.2 Norton theorem applied to the OTA output

Noise sources in the OTA schematic are represented in Figure B.4. The common mode current from the bias transistor M9 will be rejected by the OTA's large common mode rejection ratio (CMRR), and it is hence neglected. Without performing calculations, we furthermore note that only a fraction of the noise current from cascode transistors M3-6 is transferred to the output: applying the shift theorem to each current source and noticing that the source shows low impedance, we can see that the current injected at one node will be recollected by the other (as shown in Figure A.1Figure B.3), thus the residual will be negligible.

Current from transistors M7-8 will instead almost entirely flow towards the output, specifically a fraction dependent on the drain-to-source resistances r_0 of transistors M8 and M6:

$$\frac{r_0^{(8)}}{r_0^{(6)}//(1/g_m^{(6)}) + r_0^{(8)}} \sim 1 \tag{B.3}$$

Thus, we have:

$$i_{SC}^{noise} \cong i_{M1} + i_{M2} + i_{M7} + i_{M8} \tag{B.4}$$

$$S_{V_{in}}^2 = \frac{S_{i_{SC}}^2}{g_{m_{IN}}^2} = 2 \cdot 4kT\gamma \cdot \frac{g_{m_{IN}} + g_{m_{MIRR}}}{\left(g_{m_{IN}}\right)^2} = 2 \cdot 4kT\gamma \cdot \frac{1}{g_{m_{IN}}}\left(1 + \frac{g_{m_{MIRR}}}{g_{m_{IN}}}\right) \tag{B.5}$$
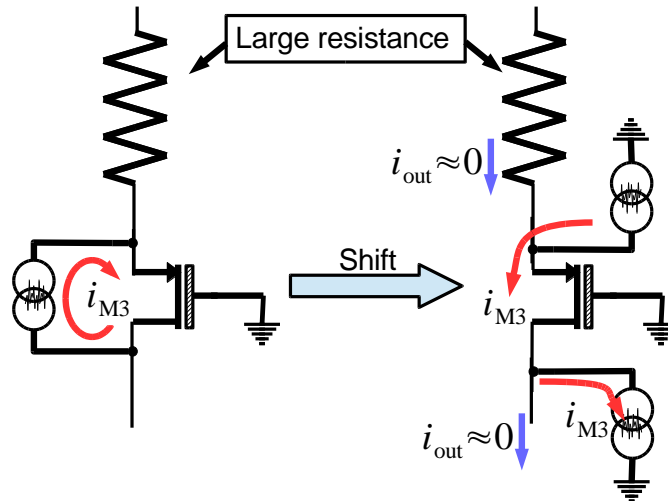
Figure B.3 Negligible noise of the cascode transistors (M3 in the example)
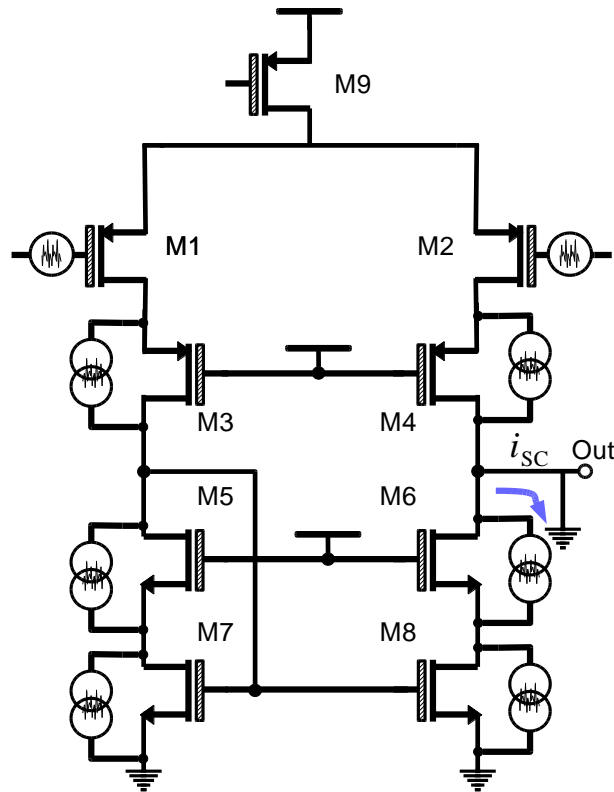

Figure B.4 OTA noise sources

Bibliography

[1] J. Nakamura, B. Pain, T. Nomoto, T. Nakamura and E. Fossum, "Design of an Image Sensor with on-Chip Oversampling Analog-to-Digital Conversion," in *IEEE Workshop on CCDs and Adv. Image Sensors*, Bruges, Belgium, 1997.

[2] J. R. Janesick, Photon Transfer, SPIE Press, 2007.

[3] M. J. Pelgrom, Analog-to-Digital Conversion, Springer, 2013.

[4] J. A. Leñero-Bardallo and A. Rodriguez-Vazquez, "ADCs for Image Sensors: Review and Performance Analysis," in *Analog Electronics for Radiation Detection*, CRC Press, 2016, p. 290.

[5] R. Fontaine, "The State-of-the-Art of Mainstream CMOS Image Sensors," in *International Image Sensors Workshop*, Vaals, The Netherlands, 2015.

[6] B. Fowler, D. Yang and A. El Gamal, "SERIAL ANALOG-TO-DIGITAL CONVERTER USING SUCCESSIVE COMPARISONS". US Patent 5801657, 1 September 1998.

[7] BOLIDE Technology Group, [Online]. Available: http://bolideco.com/support/technology-highlights/pixim-digital-pixel-system-technology.html. [Accessed 21 June 2016].

[8] Z. Ignjatovic, D. Maricic and M. F. Bocko, "Low Power, High Dynamic Range CMOS Image Sensor Employing Pixel-Level Oversampling Sigma-Delta Analog-to-Digital Conversion," *IEEE Sensors Journal,* vol. 12, no. 4, pp. 737-747, 2012.

[9] Omnivision, "Applications - Medical imaging," [Online]. Available: http://www.ovt.com/applications/application.php?id=5. [Accessed 14th May 2016].

[10] A. J. Jerri, "The Shannon Sampling Theorem-Its Various Extensions and Applications: A Tutorial Review," *Proceedings of the IEEE,* vol. 65, no. 11, pp. 1565-1596, 1977.

[11] S. R. Norsworthy, R. Schreier and G. C. Temes, Delta–Sigma Data Converters: Theory, Design, and Simulation, Wiley-IEEE Press, 1996.

[12] R. Schreier and G. C. Temes, Understanding Delta-Sigma Data Converters, Wiley-IEEE Press, 2004.

[13] A. Xhakoni and G. Gielen, "Low-Noise Detectors through Incremental Sigma-Delta ADCs," in *Analog Electronics forRadiation Detection*, Abingdon, CRC Press, 2016, pp. 71-90.

[14] A. Mahmoodi and J. Dileepan, "Optimization of Delta-Sigma ADC for Column-Level Data Conversion in CMOS Image Sensors," in *IMTC 2007 - IEEE Instrumentation and Measurement*, Warsaw, Poland, 2007.

[15] Y. Chae and J. Cheon, "A 2.1 M Pixels, 120 Frames/s CMOS Image Sensor With Column-Parallel Delta-Sigma ADC Architecture," *IEEE Journal of Solid-State Circuits,* vol. 46, pp. 236-247, January 2011.

[16] J. Márkus, J. Silva and G. C. Temes, "Theory and Applications of Incremental Delta Sigma Converters," *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS,* vol. 51, no. 4, pp. 678-690, 2004.

[17] J. Màrkus, P. Deval, V. Quiquemp, J. Silva and G. C. Temes, "Incremental Delta-Sigma Structures for DC Measurement: an Overview," in *IEEE 2006 Custom Intergrated Circuits Conference (CICC)*, San Jose, CA, USA, 2006.

[18] S. Kavusi, H. Kakavand and A. El Gamal, "On Incremental Sigma–Delta Modulation With Optimal Filtering," *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS,* vol. 53, no. 5, pp. 1004-1015, 2006.

[19] J. D. Reiss, "Understanding Sigma-Delta Modulation: The Solved and Unsolved Issues," *Journal of Audio Engineering Society,* vol. 56, no. 1/2, pp. 49-64, 2008.

[20] S. Lindfors, "A Two-Step Quantization Delta-Sigma ModulatorArchitecture With Cascaded Digital Noise Cancellation," in *The 7th IEEE International Conference on Electronics*, Jounieh, Lebanon, 2000.

[21] Y. Mengyun, D. Wu and Z. Wang, "A 15-bit Two-Step Sigma-Delta ADC With Embedded Compression For Image Sensor Array," in *2013 IEEE International Symposium on Circuits and Systems (ISCAS2013)*, Beijing, 2013.

[22] C. Jansson, "A high resolution, compact, low-power ADC suitable for array implementation in standard CMOS," *IEEE Transaction on circuits and systems,* vol. 42, no. 11, 1995.

[23] P. Rombouts, "A 13.5-b 1.2-V Micropower Extended Counting A/D Converter," *IEEE*

JOURNAL OF SOLID-STATE CIRCUITS, vol. 36, no. 2, 2001.

[24] A. L. Lacaita and M. Sampietro, Circuiti elettronici, CittàStudi, 1997.

[25] A. S. Sedra and K. C. Smith, Microelectronic circuits, Oxford University Press.

[26] S. H. Ardalan and J. J. Paulos, "An Analysis of Non-Linear Behavior in Sigma-Delta Modulators," *IEEE Transactions on circuits and Systems,* Vols. CAS-34, no. 6, pp. 593-603, 1987.

[27] A. Xhakoni and G. Gielen, "Designing Incremental Sigma-Delta ADCs for low Thermal Noise in Image Sensors," in *International Image Sensors Workshop*, Snowbird, Utah, 2013.

[28] J. Silva, U. Moon and J. Steensgaard, "Wideband low-distortion delta-sigma ADC topology," *Electronics Letters,* pp. 737-738, 2001.

[29] B. Sheu and C. Hu, "Modeling the switched-induced error voltage on a switched-capacitor," *IEEE Transactions on Circuits and Systems,* vol. 30, no. 12, pp. 911-913, 1983.

[30] J. H. Shieh, M. Patil and B. J. Sheu, "Measurement and Analysis of Charge Injection in MOS Analog Switches," *IEEE JOURNAL OF SOLID-STATECIRCUtTS,* vol. 22, no. 2, pp. 277-281, 1987.

[31] C. Eichenberger and W. Guggenbuhl, "On charge injection in analog MOS switches and dummy switch compensation techniques," *IEEE Transactions on Circuits and Systems,* vol. 37, no. 2, pp. 256-264, 1990.

[32] C. C. Enz and G. C. Temes, "Circuit Techniques for Reducing the Effects of Op-Amp Imperfections: Autozeroing, Correlated Double Sampling, and Chopper Stabilization," *Proceedings of the IEEE,* vol. 84, no. 11, pp. 1584-1614, 1996.

[33] Y. Chae, I. H. Lee, J. Cheon, G. Han and S. Heon Ham, "Decimation filters, analog-to-digital converters including the same, and image sensors including the converters". USA Patent US 8,233,068 B2, 31 July 2012.

[34] A. P. Chandrakasan and R. W. Brodersen, "Minimizing Power Consumption in Digital CMOS Circuits," *PROCEEDINGS OF THE IEEE,* vol. 83, no. 4, pp. 498-523, 1995.

[35] P. R. Gray and R. G. Meyer, Analysis and Design of Analog Integrated Circuits, 3rd ed., New York: Wiley, 1993, p. 727.

[36] M. Yue, D. Wu and Z. Wang, "A 15-bit two-step sigma-delta ADC with embedded compression for image sensor array," in *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, Beijing, 2013.

[37] Z. Gao, J. C. Liobe and M. F. Bocko, "Indirect-Feedback Sigma-Delta Image Sensors: Theory, Modeling and Design," *IEEE Transactions on Circuits and Systems,* vol. 61, no. 1, pp. 48-61, 2014.

[38] J. M. de la Rosa and R. del Rio, CMOS Sigma-Delta Converters, John Wiley & Sons Ltd, 2013.

[39] H.-L. Chen, P.-S. Chen and J.-S. Chiang, "A Low-Offset Low-Noise Sigma-Delta Modulator With Pseudorandom Chopper-Stabilization Technique," *IEEE Transactions on circuits and systems,* vol. 56, no. 12, pp. 2533-2544, 2009.

[40] R. W. Brodersen, P. R. Gray and D. A. Hodges, "MOS Switched-Capacitor Filters," *Proceeding of the IEEE,* vol. 67, no. 1, pp. 61-75, 1979.

[41] A. Xhakoni, H. Le-Thai and T. Geurts, "PTC-Based Sigma-Delta ADCs for High-Speed, Low-Noise Imagers," *IEEE Sensors Journal,* vol. 14, no. 9, pp. 2932-2933, 2014.