Design of a 12-bit 5 MSPS Pipelined Analog to Digital Converter

A project report

submited by

Vinothkumar S
EE16M042

in partial fulfillment of the requirements
for the award of degree of

MASTER OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, MADRAS
JUNE 2018

THESIS CERTIFICATE

This is to certify that the thesis entitled **Design of 12-bit 5 MSPS pipelined Analog to Digital Converter**, submitted by **Vinothkumar S**, to the Indian Institute of Technology, Madras, for the award of the degree of **Master of Technology**, is a bonafide record of the research work carried out by him under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Nagendra Krishnapura
Project advisor
Associate Professor
Department of Electrical Engineering
IIT-Madras, 600036

Place: Chennai
Date:

ABSTRACT

KEYWORDS: Pipelining, MDAC, residue, non-overlapping clocks, latency, adder

This project involves the design of a 12-bit pipelined ADC with sampling rate of 5 MSPS for 2 V peak to peak differential input in SCL 0.18 µm CMOS technology with a 1.8 V supply. It is designed in 6 stages with each stage contributing 2 bits (effectively). Each stage has flash ADC and an MDAC, except the last stage which contains only flash ADC. The first stage converts the input to 3 bit binary data, generates amplified residue and pass it to next stage and the process goes on before the last stage, where the last stage just generates 2 bit binary code only. The stages are grouped into odd and even stages by using two non-overlapping clocks. Finally bits from each stage need are delayed and then added in overlapping fashion to get 12 bits. The latency of this ADC is about 3 clock cycles.

ACKNOWLEDGEMENTS

My sincere gratitude to my advisor Dr. Nagendra Krishnapura for his guidance and for showing patience towards me in completing this project. I want to thank him for giving this golden opportunity to work under him and for inspiring me towards research and systematic way of analyzing anything. This project would not have been completed without the help of many individuals who helped me at times when I got stuck at some problem. I would like to extend my sincere thanks to all of them. I want to thank Prof. Dr. Shanti Pavan for his wonderful VLSI Data Converters lectures. I want to thank my labmate Ravi Teja who spent lot of time with me in lab and motivated me to do better. Finally I want to thank my projectmate Shreesha who helped me a lot right from the starting of the project.

Contents

List of Figures

List of Tables

ABBREVIATIONS

ADC ..... Analog to Digital converter

CMFB ..... Common mode feedback

DAC ..... Digital to Analog converter

DEC ..... Digital error correction

DEM ..... Dynamic element matching

GBA ..... Gain boosting amplifier

LSB ..... Least bignificant bit

MSB ..... Most significant bit

MSPS ..... Mega samples per second

NGBA ..... NMOS gain boosting amplifier

NMOS ..... n-channel MOSFET

PGBA ..... PMOS gain boosting amplifier

PMOS ..... p-channel MOSFET

PSD ..... Power spectral density

S/H ..... Sample and hold

SC ..... Switched sapacitor

SCA ..... Switched sapacitor Amplifier

SAR ..... Successive Approximation ADC

UGB ..... Unity gain bandwidth

Chapter 1

Introduction

In the past few years, due to the rapid improvement in the speed of computation, area effectiveness and noise immunity, Digital Signal Processing(DSP) has garnered the attention of the industry. Hence most of the computations and signal processing which where done previously in the analog domain are now done in digital domain. Since the real world data is analog in nature, to process them using DSP or to store them, we need to first convert them to digital signals using Analog to Digital Converters(ADC). As humans can't interpret digital signals, we then have to convert back the processed or stored signals to analog using Digital to Analog Converter(DAC). This is illustrated in the Figure 1.1

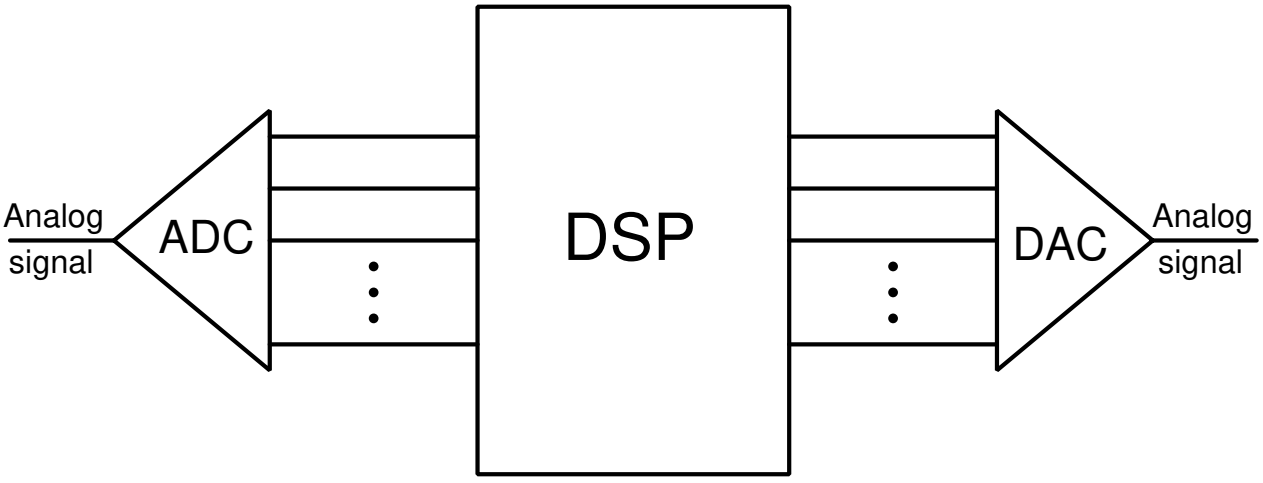


Figure 1.1: Functional view of interface devices

1.1 Basics of ADC

Analog to Digital Converter converts continuous time and continuous amplitude signal to discrete time and discrete amplitude signal. This conversion happens in three steps namely sampling, quantization and encoding. There are two key aspects which determines the accuracy with which the digital sequence of values captures the original continuous signal. They are the rate at which we sample (sampling rate, f_s) and the number of bits (resolution, N) used to define each sample.

1.1.1 Sampling

Sampling is the first step in analog to digital conversion. It is the process of converting continuous time signal to discrete time signal. It involves capturing the value of the signal at regular intervals of time $T_S(=\frac{1}{f_s})$. According to Nyquist sampling theorem, a signal bandlimited to 'B' Hz can be reconstructed perfectly if it is sampled with a sampling rate(f_s) of greater than or equal to '2B' Hz i.e. $f_s \geq 2B$. Figure 1.2 represents CT signal and the sampled version. Here the input signal frequency is 1 Hz and the sampling frequency is 7 Hz.

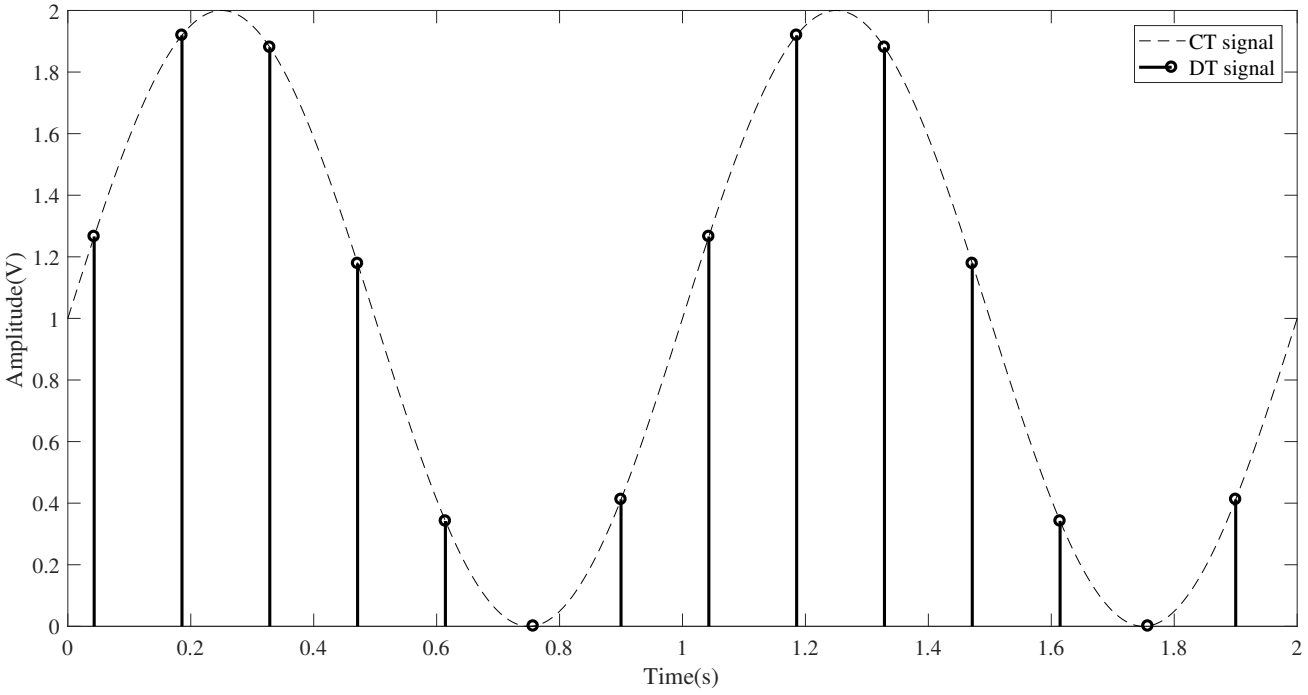


Figure 1.2: Continous Time signal and Discrete Time signal

1.1.2 Quantization

Quantization is the process of converting continuous amplitude signal to discrete amplitude signal. The block which performs quantization is called a quantizer. Quantization involves rounding the amplitude of the signal to the nearest predefined levels. Due to this, there is a deviation between actual signal and quantized signal which results in irrecoverable loss of information. Such deviation is called quantization error. There are two kinds of quantizer namely Uniform quantizer and Non-uniform quantizer. In Uniform quantizer, the step size is fixed whereas in Non-uniform quantizer, the step size varies. If the step size in uniform quantizer is denoted as Δ(which depends on the resolution of quantizer), then the maximum quantization error is Δ.

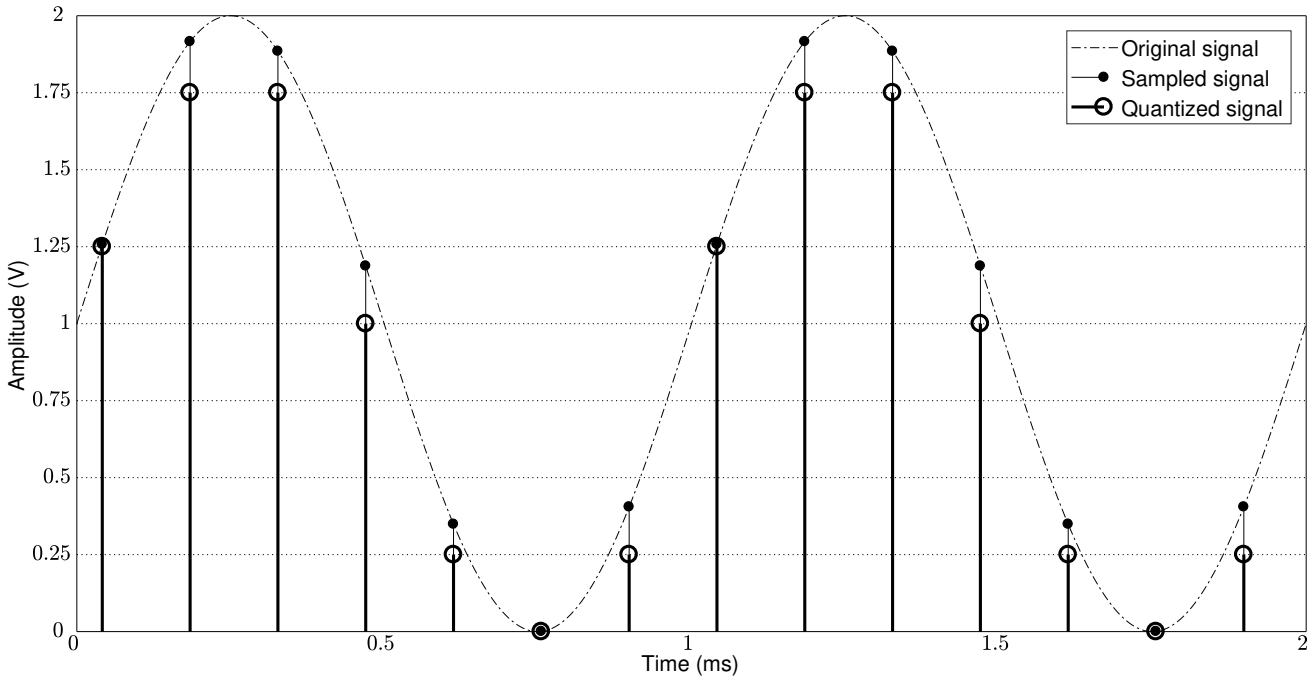


Figure 1.3: Continous Time signal, Discrete Time signal and Quantized signal

Since the quantization error is input dependent, it is very difficult to model it. Hence it is modelled as Additive White Gaussian Noise(AWGN). This assumption agrees with the statistical data.

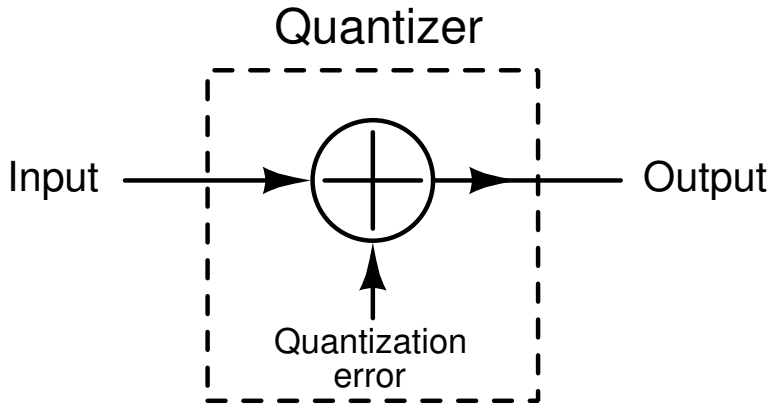


Figure 1.4: Model of quantizer

Let us assume the input as sinusoidal signal with the amplitude 'A' and the resolution of the quantizer as N. The signal power is $\frac{A^2}{2}$. The noise power is calculated as (assuming uniform probability of distribution of quantization error).

$$e_q = \frac{\Delta^2}{12} = \left(\frac{1}{12}\right)\left(\frac{A}{2^N}\right)^2 \tag{1.1}$$

Hence the maximum Signal to Noise Ratio (SNR) (which occurs for full scale input) can be calculated as

$$SNR = (6.02 * N + 1.76)dB \tag{1.2}$$

1.1.3 Encoding

It is the process of converting quantized signals to digital signal by assigning a unique label to each level. For example, in a 3 bit ADC, there are 8 levels which can be encoded as 000, 001 ... 111.

1.2 Types of ADC

Based on the speed of conversion, circuit complexity, power consumption etc., there are different varities of ADC architectures namely Flash or parallel ADC, Successive approximation ADC, Counter type ADC, Dual slope or integrating ADC, Sigma-Delta ADC, Pipelined ADC etc.

1.2.1 Flash or parallel ADC

The flash ADC comprises of series of resistors which provides unique reference voltages from the given reference voltage. The input is then compared with these reference voltages which results in strings of 0's followed by strings of 1's. These strings of 0's and 1's are fed to the priority encoder which produces corresponding binary output for the analog input. This is the fastest ADC whose conversion time is independent of its resolution. But it has a disadvantage that the component requirement increases exponentially with the increase in resolution. An N bit ADC requires $2^N - 1$ comparators and 2^N resistors. Flash ADC consumes lot of static power as there is a constant current flowing through the resistors. Resistor mismatch is also a concern as as it leads to shift in reference voltages.

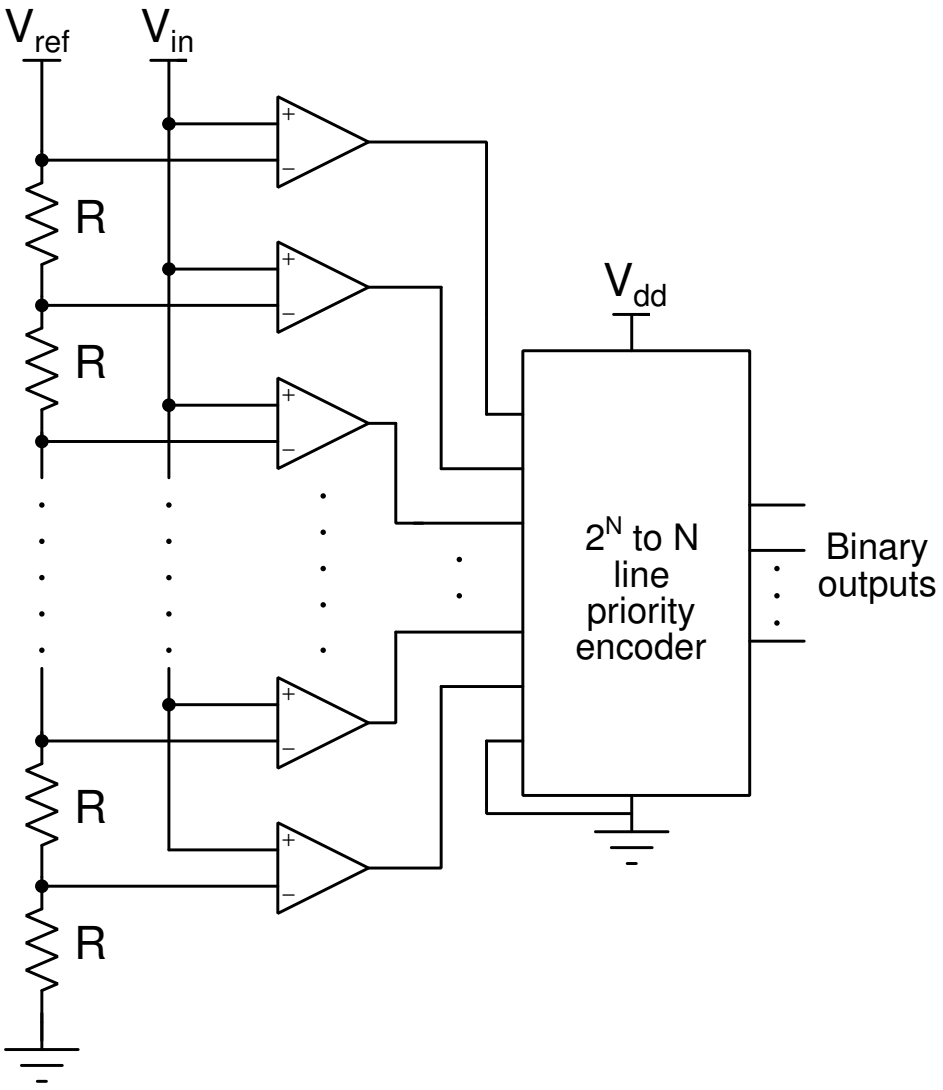


Figure 1.5: Block diagram of flash ADC

1.2.2 Successive Approximation ADC

The Successive approximation ADC uses binary search algorithm. Initially, the MSB of the binary output is set to 1. The DAC finds the corresponding analog output(V_D) which is then compared with the actual analog input(V_A). If $V_D > V_A$, it sets MSB to 0 and then its right next bit to 1 else it just sets its right next bit to 1. This process goes on until it sets the LSB. The maximum conversion time of this ADC is $N * T_{CLK}$.

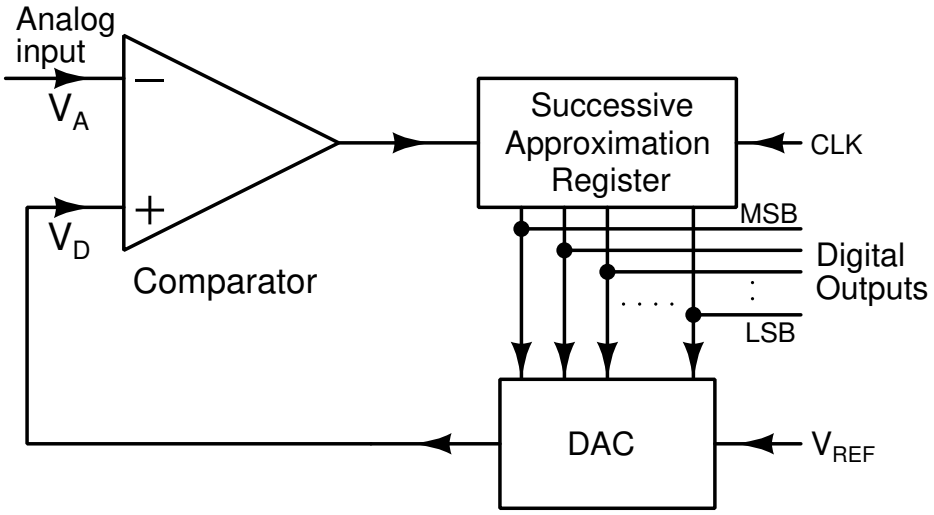


Figure 1.6: Block diagram of SAR ADC

1.2.3 Counter type ADC

In the counter type ADC, initially RESET signal is applied which sets the binary outputs of the counter to all 0's. Hence DAC output(V_D) is zero and the output of comparator is high. This allows the clock signal to propagate to the binary counter. The binary counter starts counting which results in the corresponding analog output (V_D) from DAC. This operation goes on until $V_D > V_A$. The counter stops at this instant and the conversion is said to be done. This is the simplest ADC in terms of circuitry but the problem with this is that the maximum conversion time is $(2^N - 1) * T_{CLK}$.

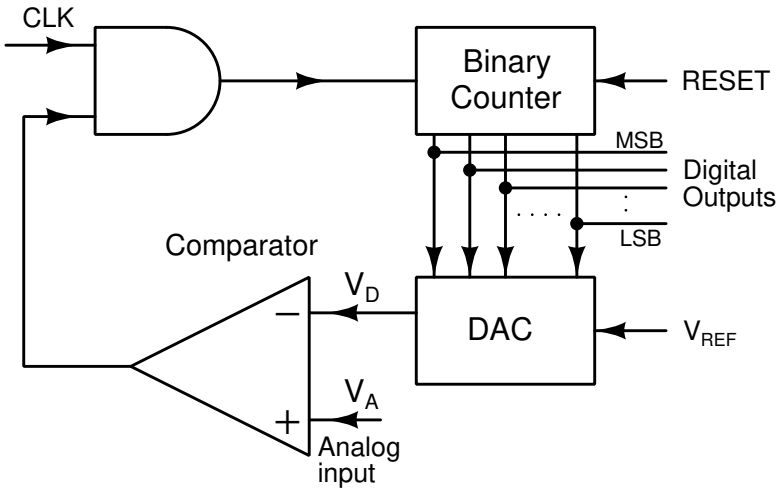


Figure 1.7: Block diagram of counter type ADC

1.2.4 Dual slope or integrating ADC

In this ADC, initially, the input (V_{IN}) is integrated for a fixed period of time. As the input to the RC integrator is positive, its output is negative which is fed as the input to comparator. As the input to the comparator is negative, its output goes to V_{DD} which allows the propagation of clock to the binary counter. The integration continues till the counter overflows. Once it overflows, the count value is reset and the switch position changes and hence RC starts integrating the reference($-V_{REF}$) with a positive slope. The integration stops when the output of RC goes slightly above 0. The binary counter value at that instance gives the exact digital output. The maximum conversion time of this ADC is $(2^N + 1) + 2^N$ which is $2^{N+1} + 1$. The operation of the ADC is independent of the component values (R,C) and is less sensitive to supply hum but it is the slowest ADC among all the architecture.

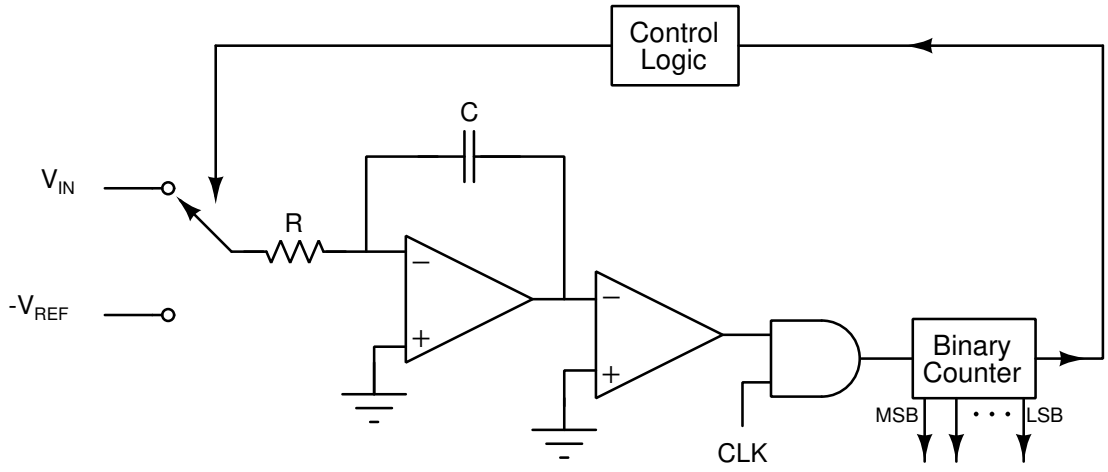


Figure 1.8: Block diagram of dual slope ADC

1.2.5 Delta-Sigma ADC

The Delta-Sigma ADC works based on the principle of over-sampling and noise shaping. It has three main blocks namely Delta-Sigma modulator, digital low-pass filter and decimator. As we know that the quantization introduces quantization noise which is due to the fact that the input analog signal has infinite possible states while the quantized signal has only fixed number of levels which is determined by the resolution (N) of the ADC. To reduce the quantization noise, one way is to increase the resolution of the ADC. But the other way is to increase the sampling rate to $K * f_S$ which allows the spreading of quantization noise. Even though the rms noise remains same, due to filtering the rms noise outside the band of interest goes away. Each factor-of-4 oversampling increases the SNR by 6dB, and each 6dB increase is equivalent to gaining one bit. The Delta-Sigma modulator produces 1's or 0's at the sampling rate. The density of 1's at the output of modulator is proportional to the input signal. It is evident from the figure that the input to output has a low pass response while the quantization noise to output has a high pass response. This pushes most of the quantization noise to higher frequencies (noise shaping). Oversampling has changed not the total noise power, but its distribution. If we apply a digital filter to the noise-shaped delta-sigma modulator, it removes more noise than does simple oversampling. The improvement in noise due to oversampling is given by the equation

$$SNR = (6.02 * N + 1.76 + 10log((k * f_s)/(2 * BW))\, dB \tag{1.3}$$

where k is the over sampling ratio(OSR), f_s is the sampling frequency and BW is the bandwidth of the input signal. The digital filtering filters out the high-frequency noise while the decimator reduce the data rate to usable amount. This type of modulator (first-order) provides a 9dB improvement in SNR for every doubling of the sampling rate. For higher orders of quantization, we can achieve noise shaping by including more than one stage of integration and summing in the sigma-delta modulator. Delta-Sigma ADC's are mostly employed in high resolution and low frequency applications. As the sampling rate is much higher than other ADC's, the complexity of the circuit is very high.

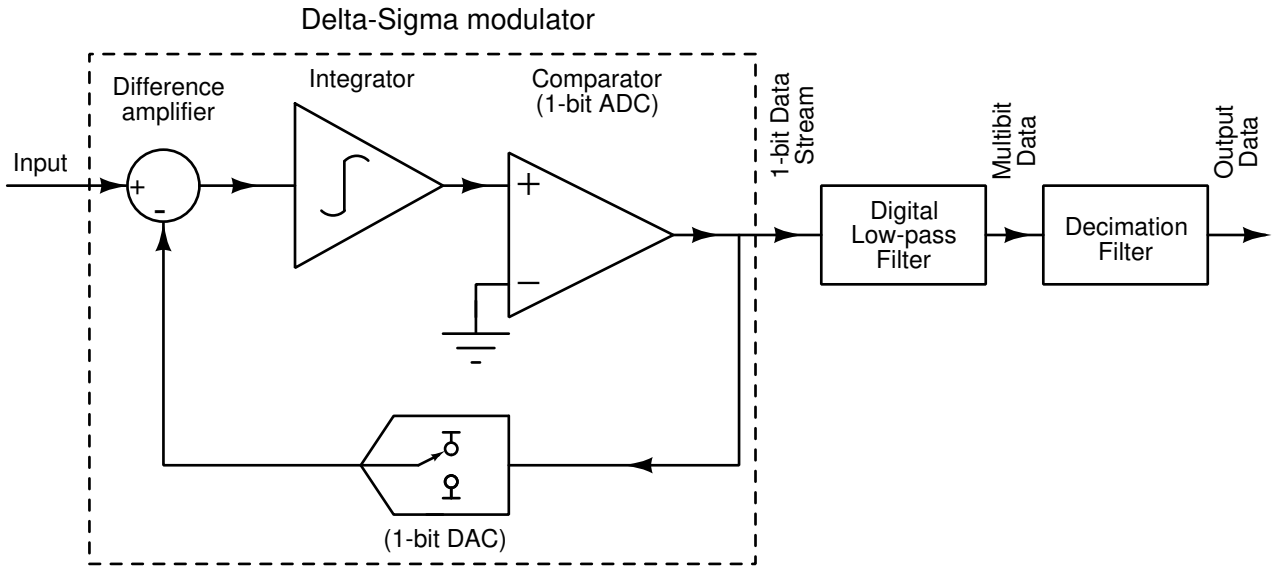


Figure 1.9: Block diagram of Delta-Sigma ADC

1.2.6 Pipelined ADC

The pipelined ADC architecture is a compromise between flash ADC and Successive Approximation ADC. SAR ADC resolves single bit at a time from MSB down to LSB and flash ADC resolves all the bits at a time. But the pipelined ADC resolves multiple bits at a time It achieves the same throughput as that of a flash ADC with reduced hardware complexity. The number of comparators needed is more than that of a SAR ADC but much lower than that of a flash ADC.

Original pipelined architecture

For simplicity let us consider N-bit two stage pipelined ADC as shown in Figure 1.10. The first stage stage resolves N_1 bits while the second stage which is just a flash ADC resolves N_2 bits.

The input is sampled by the ADC_1 and it gives N_1 number of bits. Depending on the output bits of ADC_1, DAC generates the output voltage which is then subtracted from the sampled input to generate the error. The error is amplified to full scale using an amplifier of gain 2^{N_1} and is given as input for next stage. The ADC_2 extracts the remaining N_2 bits. The outputs of two stages are added in non-overlapping fashion using adder. When everything works as expected (ideal case), quantization error V_q (residue) and the amplified residue (V_o) will have the following characteristics as shown in Figure 1.11.

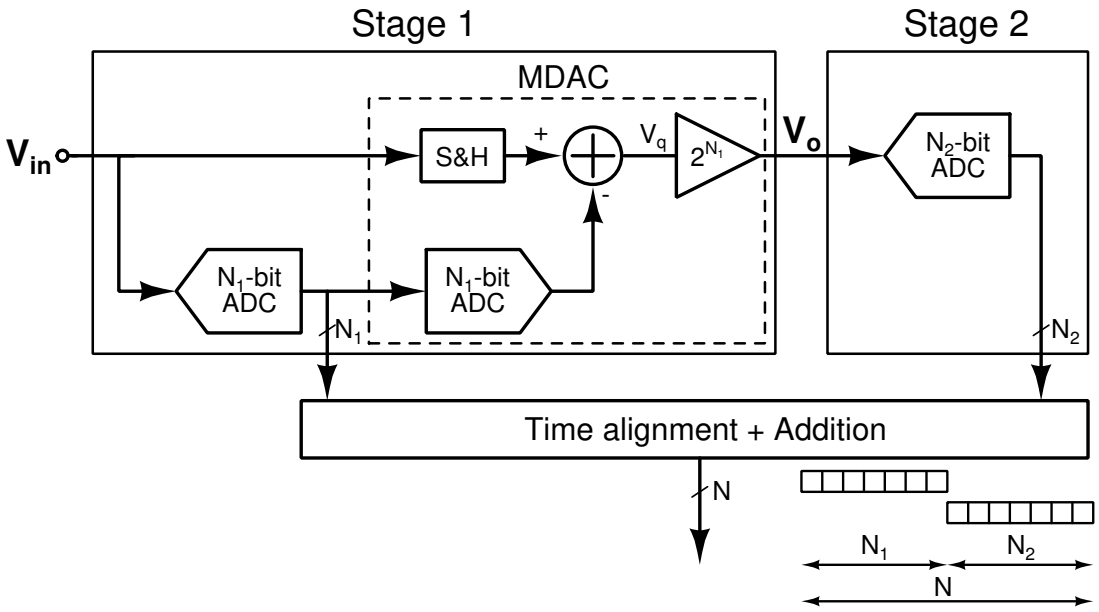


Figure 1.10: Block diagram of a N-bit pipelined ADC

- Quantization error (V_q) lie within 0 to V_{LSB}
- Maximum quantization error (V_q) occurs at an integer multiple of V_{LSB}
- The amplified residue exactly fit into the range (0 to V_{REF}) of next stage.

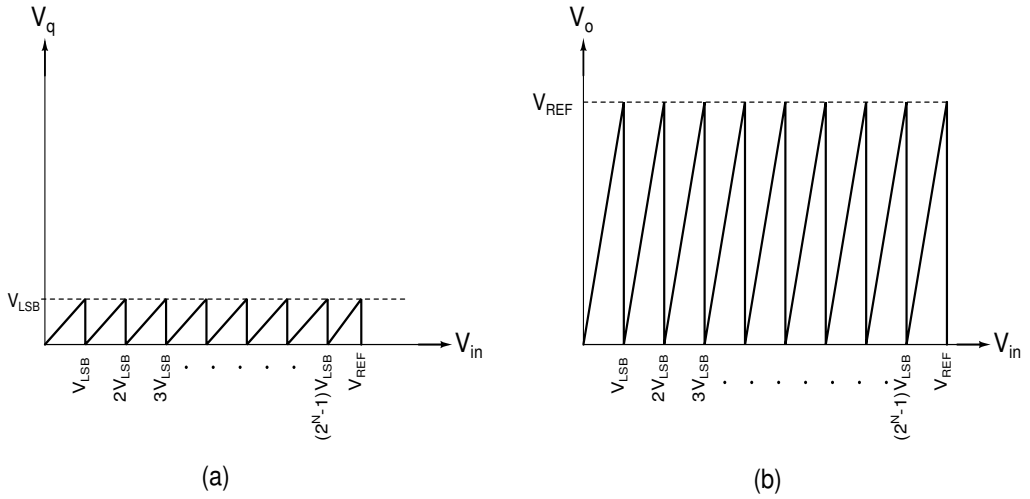


Figure 1.11: Two stage pipelined ADC (a) V_{in} vs V_q (b) V_{in} vs V_o

To get the final resolution of pipelined ADC, the bits from stage 1 needs to be delayed appropriately and then it is added with the bits from stage 2 in a non-overlapping fashion.

Modified pipelined architecture

The original pipelined ADC suffers from the problem that whenever is an error in the transition point of error voltage V_q due to comparator offset or if there is a gain error, then these leads to overloading of ADC_2 which is not acceptable. To get rid of this problem the following changes are made

- Reduce the interstage gain to 2^{N1-1}.
- Shift all the transition points of ADC_1 to the right by 0.5 LSB.
- Reduce the number of comparators of ADC_1 to 2^{N_1-1} by removing the last comparator.

The modified pipelined architecture is shown in Figure 1.12 and the modified residue is shown in shown in Figure 1.13.

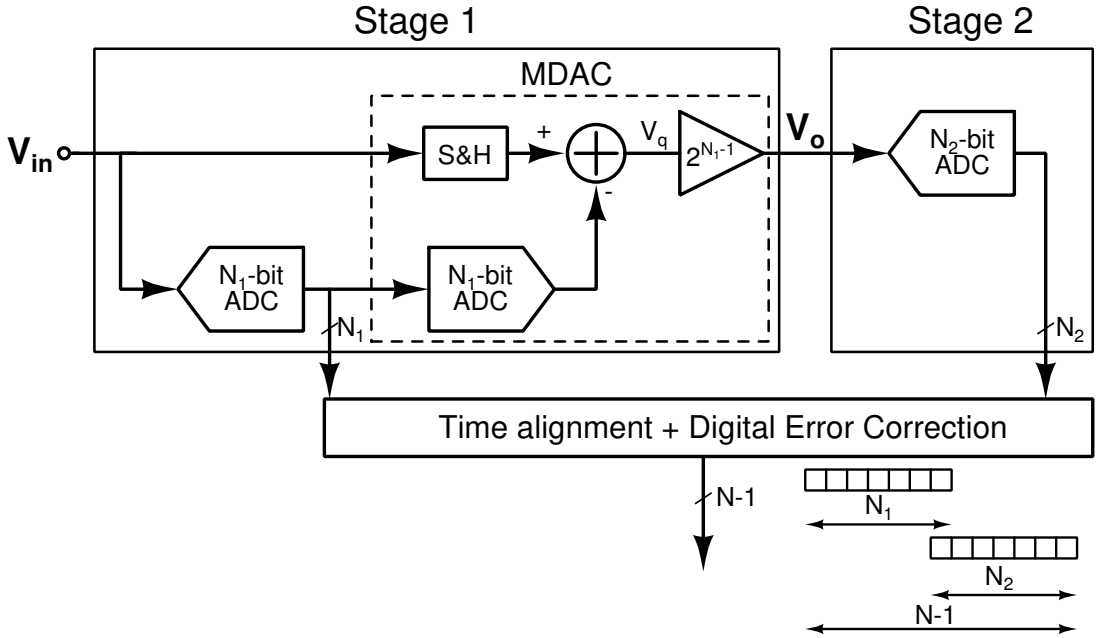


Figure 1.12: Modified pipelined ADC architecture

As the interstage gain is reduced to 2^{N1-1}, the resulting residue occupies less than the full scale of next stage. Hence the next stage output bits should be added with 1 bit overlap.

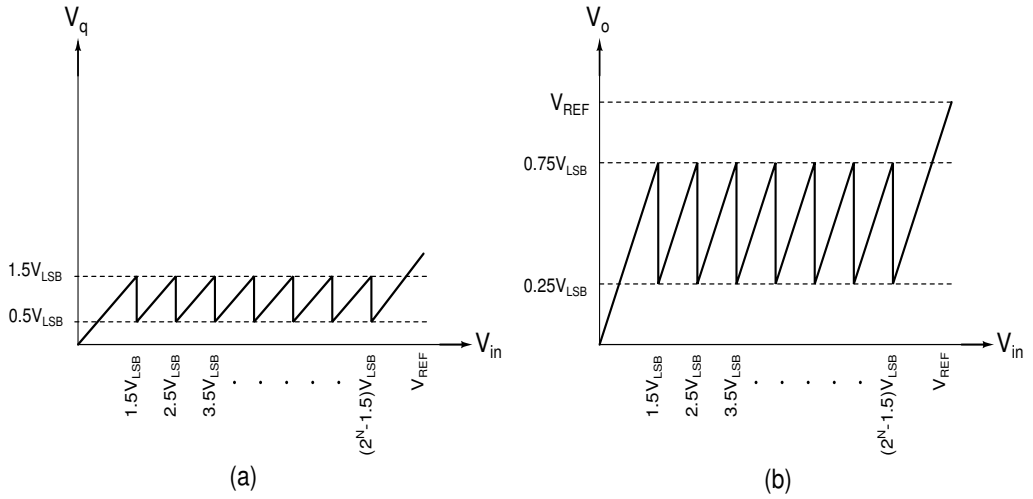


Figure 1.13: Modified pipelined ADC architecture (a) V_{in} vs V_q (b) V_{in} vs V_o

The block which performs the combined function of DAC, subtraction and amplification is called Multiplying DAC(MDAC). The pipelined ADC has been designed with the specification given in Table 1.1. Figure 1.14 shows the architecture of 6 stage 12 bit pipelined ADC. Each stage extracts 3 bits which are added in overlapping fashion to produce 12 bits.

Stages	6 (effectively 2 bits from each stage)
Sampling rate (f_S)	5 Mega samples/second
Supply voltage	1.8 V
Common mode voltage	0.9 V
Reference voltage	0.4 V & 1.4 V
Input	$2\,V_{PP}$ differential $V_{inp} = 0.9 + 0.5 * sin(2\pi * f_{in} * t)\,V$ $V_{inm} = 0.9 - 0.5 * sin(2\pi * f_{in} * t)\,V$

Table 1.1: Specifications of 12 bit pipelined ADC

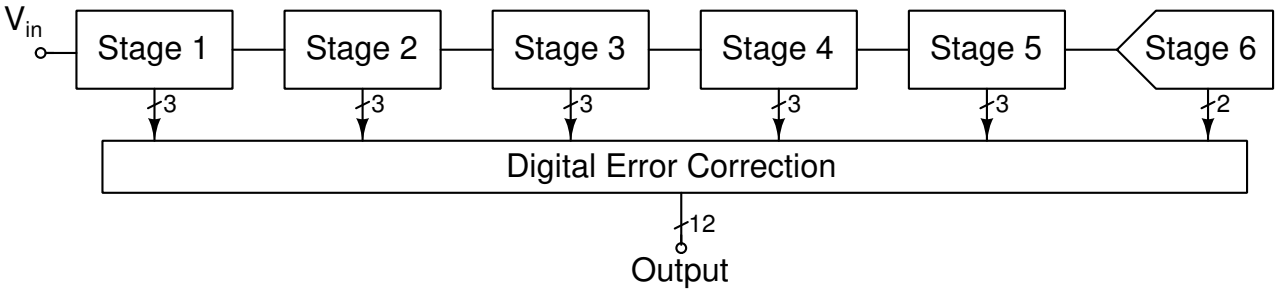


Figure 1.14: Block diagram of 12 bit pipelined ADC

Chapter 2

MDAC architecture

2.1 Switched capacitor amplifier

The circuit operates in two phases with the help of sampling (ϕ_s) and hold (ϕ_h) signals which are non-overlapping signals. During sampling phase, the input is sampled on to the sampling capacitor (C1). During hold phase, this charge gets transferred to the hold capacitor (C2). Hence the voltage across capacitor C2 is given by

$$V_{out} = \frac{C1}{C2} * V_{in} \tag{2.1}$$

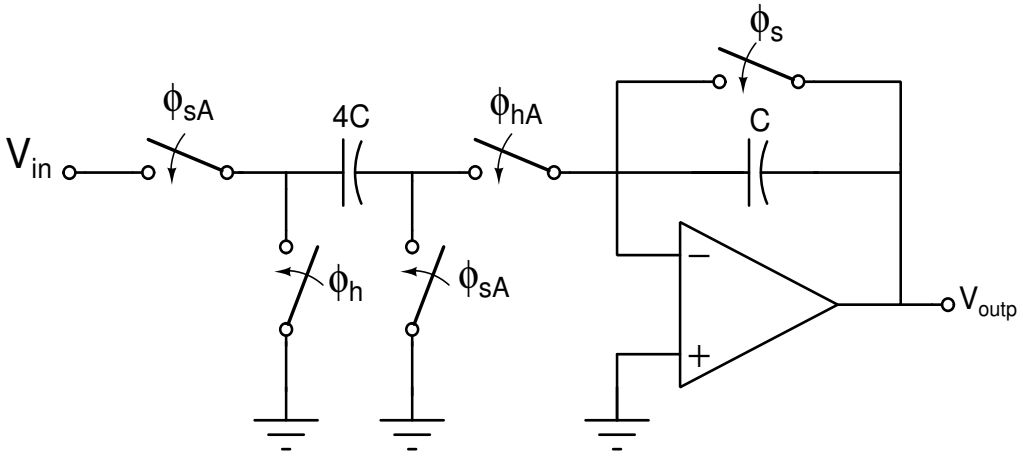


Figure 2.1: Circuit diagram of the switched capacitor amplifier

In Figure 2.1, ϕ_{sA} and ϕ_{hA} are used instead of ϕ_s and ϕ_h to get rid of charge injection problem.

2.2 MDAC

The switched capacitor amplifier discussed above can be modified to MDAC with few changes. During hold phase, the bottom plate of the sampling capacitor is connected to output of ADC in MDAC as opposed to ground in SCA. The circuit diagram of fully differential MDAC is shown in Figure 2.2.

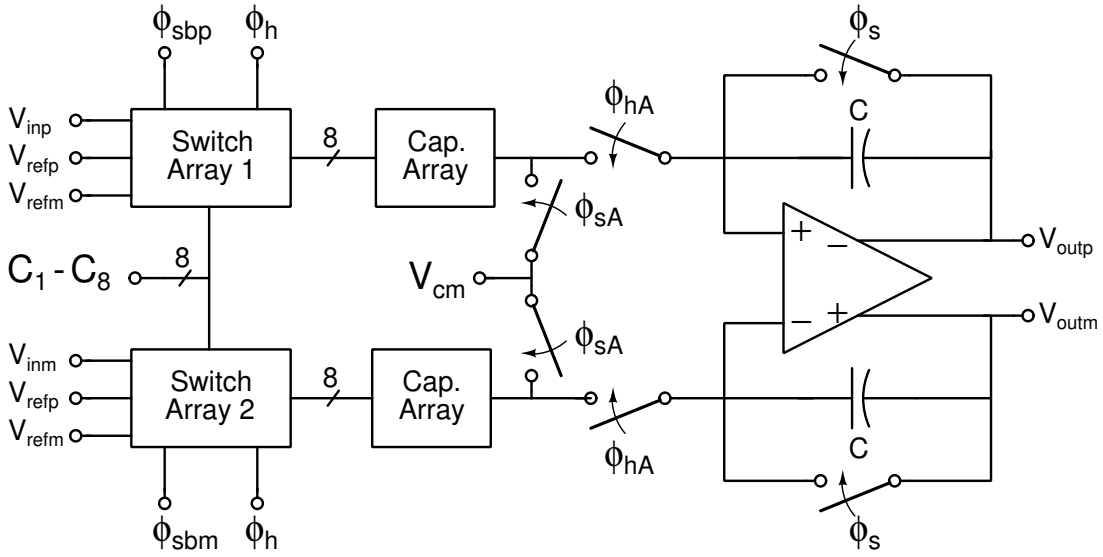


Figure 2.2: Circuit diagram of MDAC

The sampling capacitor (4C) is divided into an array of 8 capacitors of $\frac{C}{2}$ each to facilitate the connection of ADC output. The feedback capacitor (C) is also divided into 2 capacitors of $\frac{C}{2}$ each for matching purpose. The capacitor array is shown in Figure 2.3.

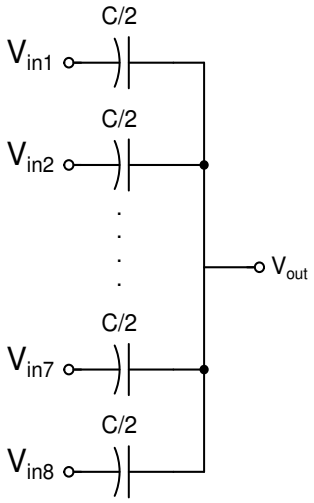


Figure 2.3: Circuit diagram of capacitor array used in MDAC

V_{in1} to V_{in8} are generated from the switch array 1 and switch array 2. Each switch array consists of 8 switch boxes. This is shown in Figure 2.4 and Figure 2.5.

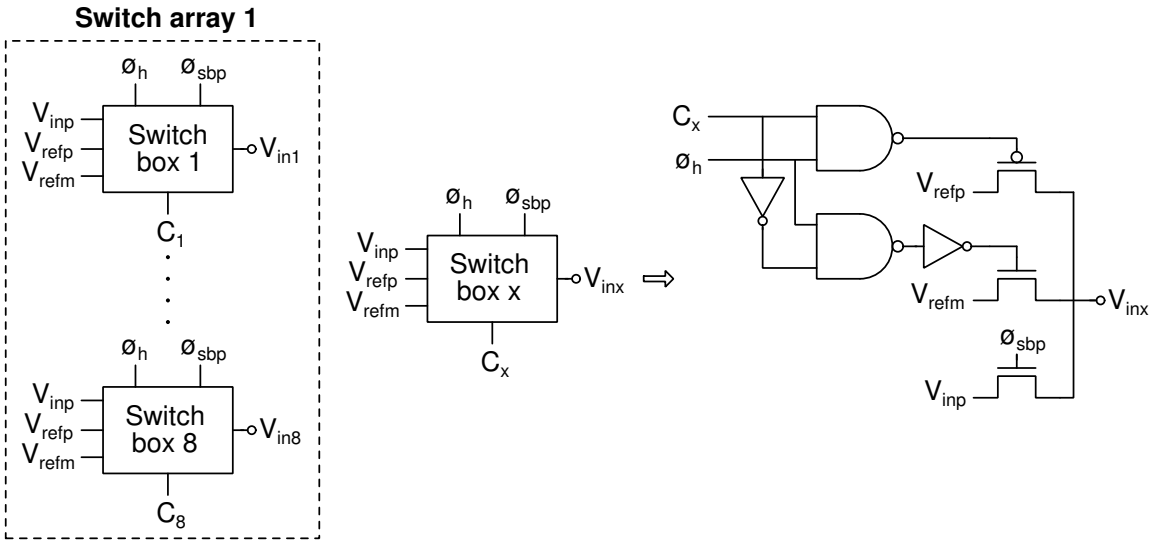


Figure 2.4: Circuit diagram of switch array 1 used in MDAC.

Phase	$\phi_{\mathbf{h}}$	$\mathbf{C_x}$	$\mathbf{V_{inx}}$
Sampling	0	x	V_{inp}
Hold	1	0	V_{refm}
Hold	1	1	V_{refp}

Table 2.1: Output of switch box used in switch array 1

During sampling phase irrespective of control signal from ADC, V_{inx} is V_{inp} for swith array 1 and V_{inm} for swith array 2 respectively. During hold phase depending on the control signal from ADC, V_{inx} is either V_{refp} or V_{refm} which is shown in Table 2.1 and Table 2.2.

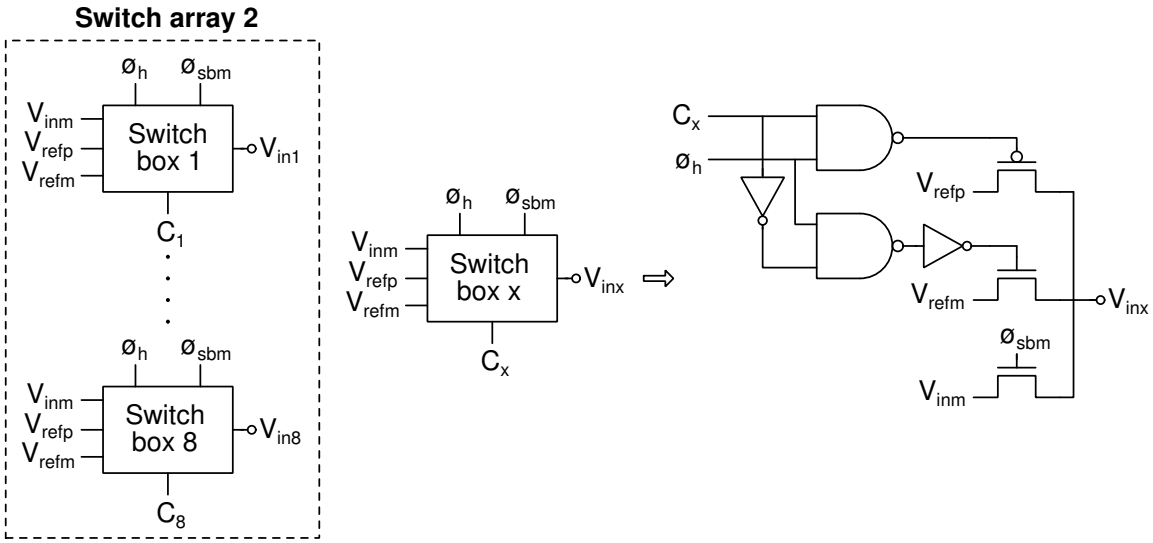


Figure 2.5: Circuit diagram of switch array 2 used in MDAC.

Phase	$\phi_{\mathbf{h}}$	$\mathbf{C_x}$	$\mathbf{V_{inx}}$
Sampling	0	x	V_{inm}
Hold	1	0	V_{refp}
Hold	1	1	V_{refm}

Table 2.2: Output of switch box used in switch array 2

During ϕ_s, V_{in} is given to the bottom plate of the capacitor. During ϕ_h, either V_{refp} or V_{refm} is given to the bottom plate of the capacitor depending on the output of ADC. Out of 8 control signals $(C_1 - C_8)$, $(C_1 - C_6)$ comes from flash ADC and C_7 and C_8 are tied to V_{DD} and GND respectively. Table 2.3 shows the relationship between DAC output and DAC control signals.

$\mathbf{C_1}$	$\mathbf{C_2}$	$\mathbf{C_3}$	$\mathbf{C_4}$	$\mathbf{C_5}$	$\mathbf{C_6}$	$\mathbf{C_7}$	$\mathbf{C_8}$	**DAC output**
1	1	1	1	1	1	1	0	$0.75V_{ref}$
0	1	1	1	1	1	1	0	$0.5V_{ref}$
0	0	1	1	1	1	1	0	$0.25V_{ref}$
0	0	0	1	1	1	1	0	0
0	0	0	0	1	1	1	0	$-0.25V_{ref}$
0	0	0	0	0	1	1	0	$-0.5V_{ref}$
0	0	0	0	0	0	1	0	$-0.75V_{ref}$

Table 2.3: DAC output vs DAC control signals

2.3 Bootstrap signal generator

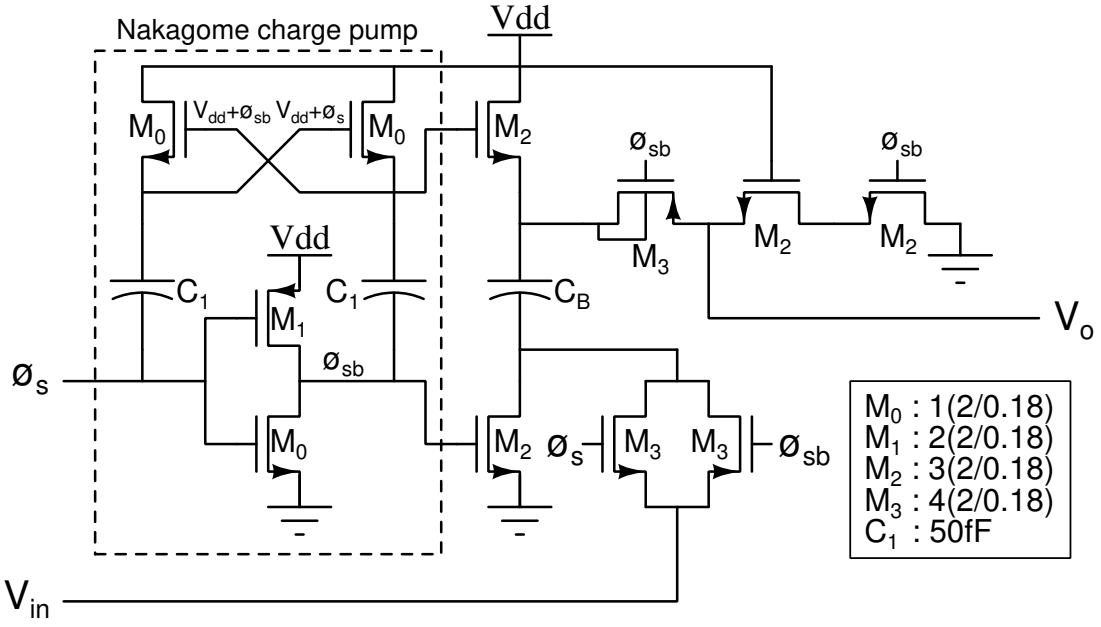


Figure 2.6: Circuit diagram of bootstrap signal generator

The ON resistance of the switch depends on V_{GS}. Hence, to make ON resistance independent of input signal, a dc voltage is applied between gate and source. This is done by connecting a capacitor and charging it when ϕ_s is low. Therefore the ON resistance is given by

$$R_{ON} = \frac{1}{\mu_n C_{ox} \frac{W}{L} V_{dc}} \tag{2.2}$$

Chapter 3

Determining C and opamp specification

3.1 Value of C

The value of C is chosen based on two criteria

- Thermal noise
- Capacitor matching

3.1.1 Thermal noise

For proper function of ADC, the thermal noise is restricted to $0.1V_{LSB}$

$$\sqrt{\frac{KT}{C_{sample}}} < 0.1V_{LSB} \tag{3.1}$$

As the sampling capacitor(C_{sample}) is divided into 8 capacitors of $\frac{C}{2}$ each, the minimum capacitance chosen to mitigate noise is given in Table 3.1.

Stage	$\boldsymbol{V_{LSB}}$**(mV)**	**C**	$\boldsymbol{C_{sample}}$
1	0.5	414 fF	1.66 pF
2	2	26 fF	0.10 pF
3	8	1.6 fF	6.4 fF
4	32	0.1 fF	0.4 fF
5	128	6 aF	24 aF

Table 3.1: Minimum capacitance to restrict the thermal noise to $0.1V_{LSB}$

3.1.2 Capacitor matching

The amplification in the MDAC depends on the ratio of capacitances. The values of capacitors however vary from the nominal value due to various reasons like imperfect edges of devices and chip gradients. This causes mismatch between capacitors. To get higher accuracy larger

value of capacitances are needed which increases the load of op-amp. To counter this problem Dynamic Element Matching(DEM) is introduced. The usage of DEM gives improved SFDR by converting the undesirable harmonic distortion (due to mismatch) into white noise.

Hence the value of capacitance (C) is chosen only taking thermal noise into consideration and it is assumed that mismatch can be countered by DEM. The value of C chosen for each stage is shown in Table 3.2.

Stage	**C (fF)**
1	1000
2	200
3	150
4	150
5	150

Table 3.2: Minimum capacitance to restrict the thermal noise to $0.1V_{LSB}$

3.2 Opamp specifications

High DC gain is required to achieve accurate closed loop gain of the interstage MDAC. Finite dc gain (Adc) and unity gain bandwidth(UGB) of the opamp result in static and dynamic errors respectively at the output of the MDAC.

3.2.1 Finite DC gain(A_{dc})

As DEC does not correct gain or linearity errors in the individual DAC and gain amplifiers, the front-end S&Hs and DACs need to be about 12-bit accurate. But the components in subsequent stages require less accuracy like 10-bit accuracy for stage 2, 8-bit accuracy for stage 3 and so on. The reduced accuracy is due to the fact that the later stage error terms are divided by the preceding interstage gain(s). Hence this fact is exploited to save additional power by making the subsequent pipelined stages progressively smaller. High For a inverting amplifier of gain = 4, finite DC gain of opamp results in a static error given by Eq.(3.3)

$$V_{out} = \frac{4}{1 + \frac{5}{A_{dc}}} * V_{FS} \tag{3.2}$$

$$Error = \frac{4}{1 + \frac{A_{dc}}{5}} * V_{FS} \tag{3.3}$$

3.2.2 Finite unity gain bandwidth(UGB)

The finite UGB results in dynamic error. The output voltage when the opamps UGB = ω_u rad/s and maximum output = V_{FS} is

$$V_{out} = (1 - e^{-t\beta\omega_u}) * V_{FS} \tag{3.4}$$

where β is the feedback factor and is equal to 0.2 in this case. By taking settling time $t = \frac{1}{2f_s}$ where f_s is the sampling frequency, the error can be calculated as in Eq.(3.5)

$$Error = e^{-\frac{\beta\omega_u}{2f_s}} * V_{FS} \tag{3.5}$$

The maximum tolerable error is $\frac{V_{LSB}}{2}$, where V_{LSB} is given by Eq.(2.8)

$$V_{LSB} = 2 * \frac{V_{FS}}{2^N} \tag{3.6}$$

where N is the number of bits to be extracted by the stages following this stage. The opamp specs obtained from the above analysis are summarised in Table 3.3.

Stage	$\boldsymbol{A_{dc}}$**(dB)**	$\boldsymbol{\omega_u}$**(MHz)**
1	74	487
2	62	400
3	50	310
4	38	222
5	26	133

Table 3.3: Opamp specification

Chapter 4

Opamp Topology

A gain boosted folded cascode opamp is used in the MDAC stage. The gain boosted opamp consists of one main opamp (NMOS folded cascode) and four auxiliary opamps (two NMOS folded cascode and two PMOS folded cascode opamps). The main opamp along with auxiliary opamps gives very high dc gain. Hence the name gain boosted opamp.

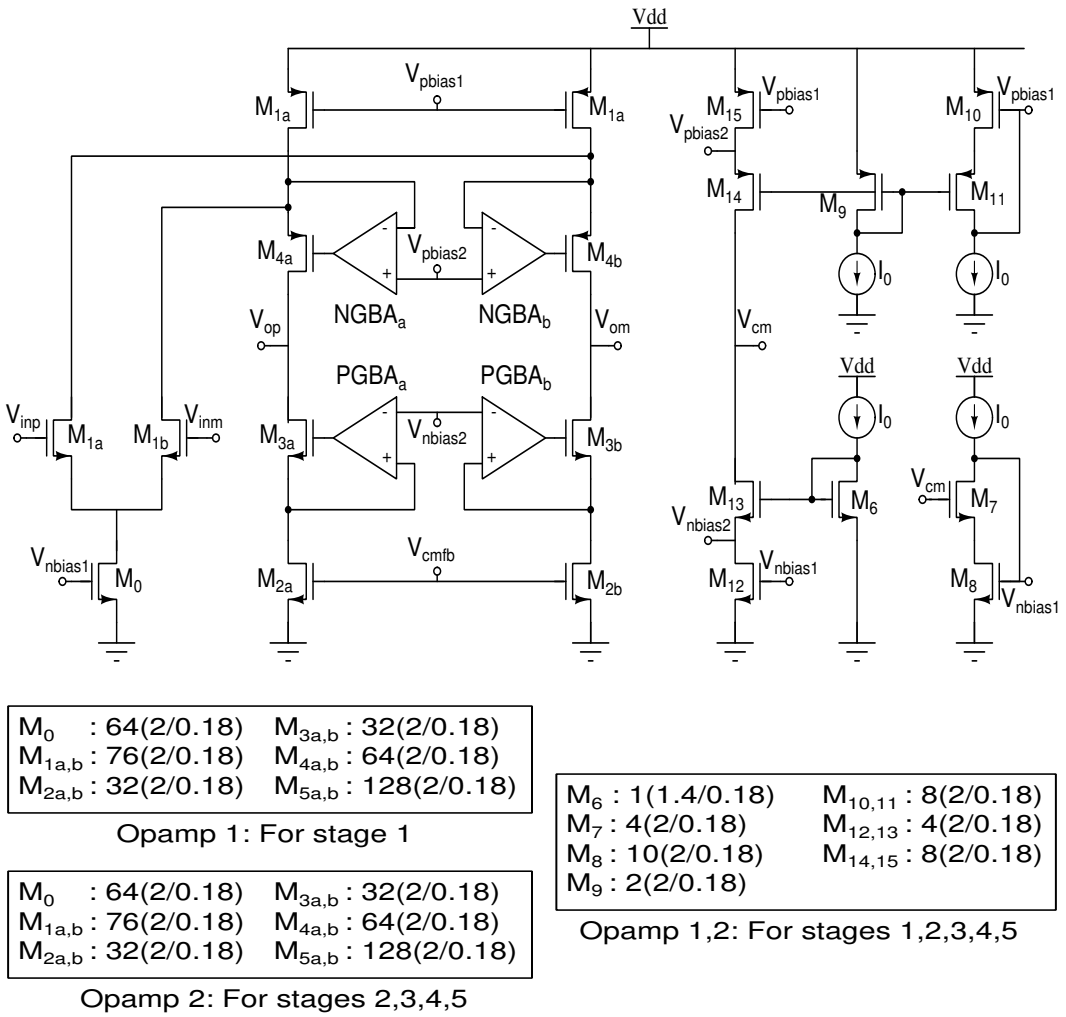
4.1 Main opamp

For the main opamp, a folded cascode architecture has been used to increase the output swing. The load seen by each opamp is different as different values of capacitors are used for sampling in different stages. The load seen by the opamp in stage N is given by Eq.(4.1).

$$C_L = 0.8 * C_N + 4 * C_{N+1} \tag{4.1}$$

where C_N is the value of the feedback capacitor of Stage N.

As the load and accuracy requirements decrease for later stages, two different opamps have been designed. The first one, Opamp1, is for stage 1 and the second one, Opamp2, is for stages 2,3,4 and 5 . The opamps in later stages have been biased at lower currents to reduce power consumption. The circuit diagram for the main opamp is shown in Fig. 4.1. The biasing current details for different stages are provided in Table 4.1.



M_0 : 64(2/0.18)	$M_{3a,b}$: 32(2/0.18)
$M_{1a,b}$: 76(2/0.18)	$M_{4a,b}$: 64(2/0.18)
$M_{2a,b}$: 32(2/0.18)	$M_{5a,b}$: 128(2/0.18)

Opamp 1: For stage 1

M_0 : 64(2/0.18)	$M_{3a,b}$: 32(2/0.18)
$M_{1a,b}$: 76(2/0.18)	$M_{4a,b}$: 64(2/0.18)
$M_{2a,b}$: 32(2/0.18)	$M_{5a,b}$: 128(2/0.18)

Opamp 2: For stages 2,3,4,5

M_6 : 1(1.4/0.18)	$M_{10,11}$: 8(2/0.18)
M_7 : 4(2/0.18)	$M_{12,13}$: 4(2/0.18)
M_8 : 10(2/0.18)	$M_{14,15}$: 8(2/0.18)
M_9 : 2(2/0.18)	

Opamp 1,2: For stages 1,2,3,4,5

Figure 4.1: Circuit diagram of main opamp with transistor sizes

Stage	$I_0(\mu A)$
1	160
2	160
3	160
4	160
5	50

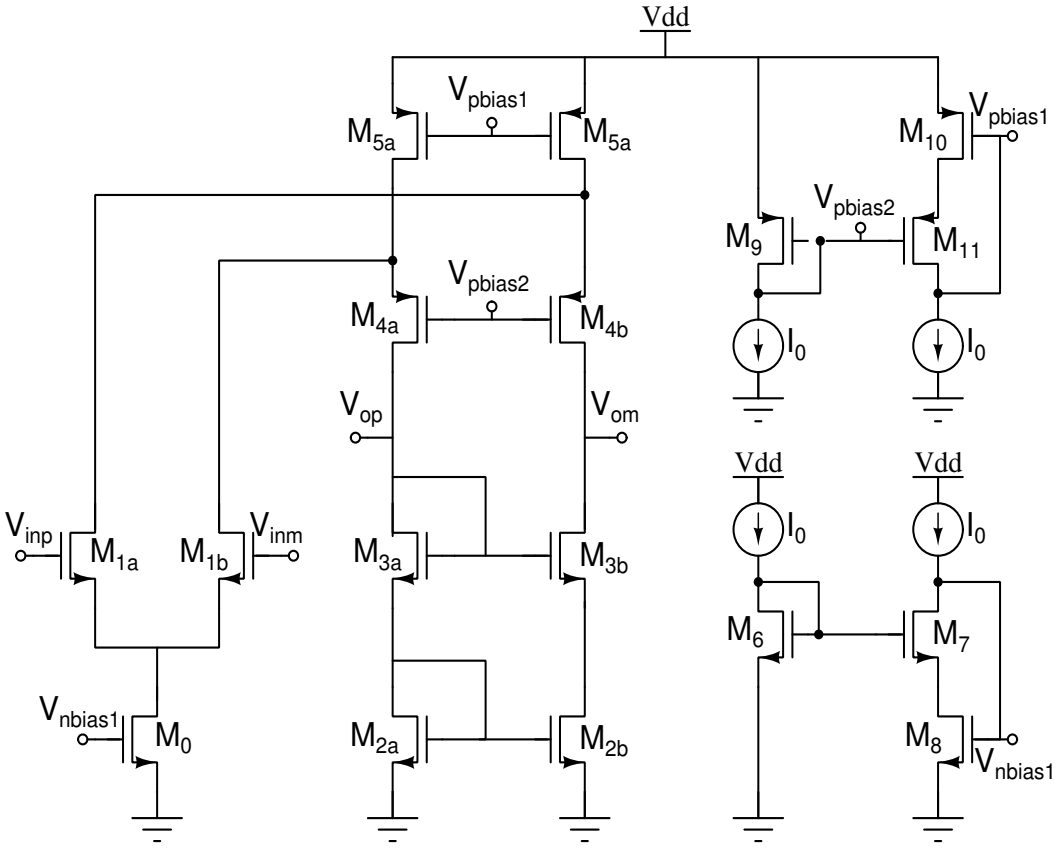
Table 4.1: Biasing current details of the main opamp

4.2 Gain boosting amplifiers

It is based on boosting the performance of the cascode used in the main opamp by using an amplifier (GBA). The effective looking in resistance (R_{cas}) of cascode is increased by a factor A_0, where A_0 is the gain of the GBA.

4.2.1 NGBA

This opamp is a NMOS input folded cascode. This architecture is chosen because the input common mode for this opamp is close to V_{dd} i.e.V_{dd} 2Δ V. Similar to main opamp, two NGBAs (NGBA1 and NGBA2) have been designed. NGBA1 is used in stage 1 and NGBA2 is used in stages 2,3,4 and 5. The circuit diagram for the NGBA is shown in Fig. 4.2.



M_0 : 16(2/1.08)	$M_{4a,b}$: 16(2/0.54)	M_9 : 1(1.5/0.54)
$M_{1a,b}$: 12(2/0.36)	$M_{5a,b}$: 12(2/0.54)	$M_{10,11}$: 4(2/0.54)
$M_{2a,b}$: 8(2/1.08)	M_6 : 2(1/1.08)	I_0 : 40uA
$M_{3a,b}$: 12(2/0.54)	$M_{7,8}$: 8(2/1.08)	

Details for $NGBA_1$

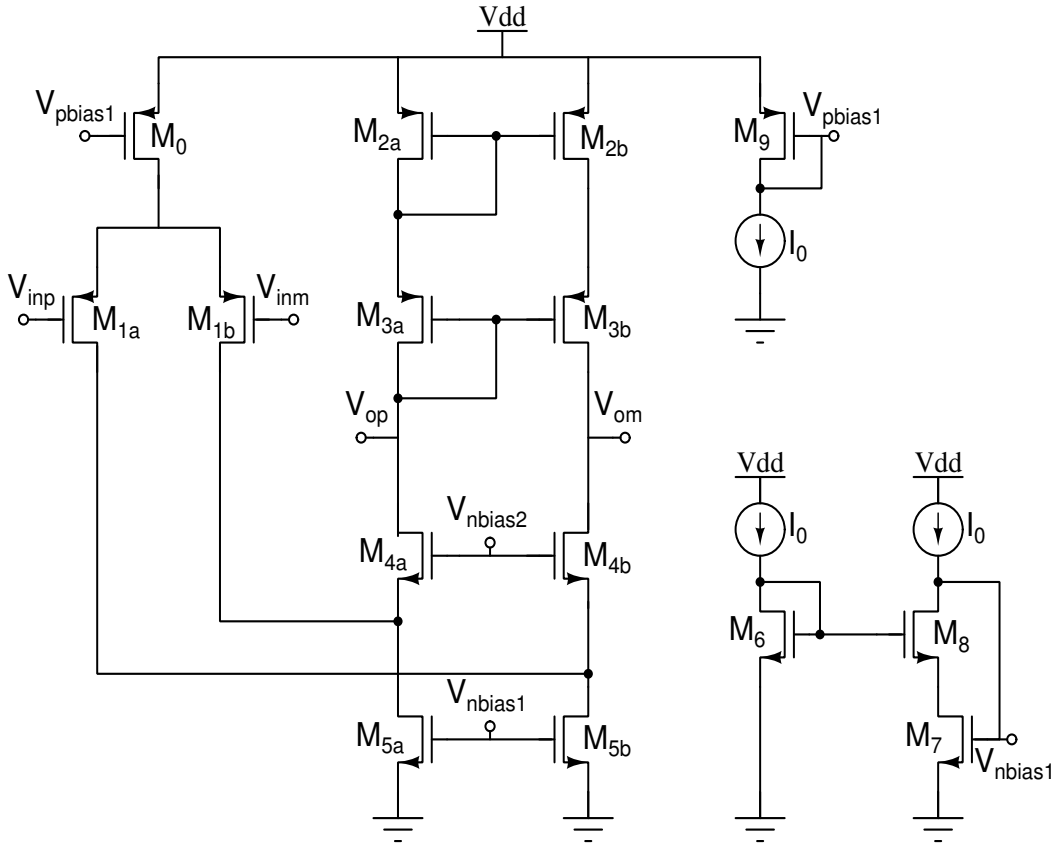
M_0 : 8(2/1.08)	$M_{4a,b}$: 8(2/0.54)	M_9 : 1(1.5/0.54)
$M_{1a,b}$: 8(2/0.36)	$M_{5a,b}$: 6(2/0.54)	$M_{10,11}$: 1(1.5/0.54)
$M_{2a,b}$: 4(2/1.08)	M_6 : 1(1/1.08)	I_0 : 40uA
$M_{3a,b}$: 6(2/0.54)	$M_{7,8}$: 2(2/1.08)	

Details for $NGBA_2$

Figure 4.2: Circuit diagram of NMOS Gain Boosting Amplifier (NGBA) with transistor sizes

4.2.2 PGBA

This opamp is a PMOS input folded cascode. This architecture is chosen because the input common mode for this opamp is close to ground i.e.2ΔV. Similar to NGBA, two PGBAs (PGBA1 and PGBA2) have been designed. PGBA1 is used in stage 1 and PGBA2 is used in stages 2,3,4 and 5. The circuit diagram for PGBA are shown in Fig. 4.3.



M_0 : 16(2/1.08)	$M_{4a,b}$: 8(2/0.72)	M_9 : 8(2/1.44)
$M_{1a,b}$: 12(2/0.36)	$M_{5a,b}$: 12(2/1.44)	I_0 : 40uA
$M_{2a,b}$: 12(2/0.25)	M_6 : 2(1.5/0.75)	
$M_{3a,b}$: 12(2/0.25)	$M_{7,8}$: 8(2/1.44)	

Details for $PGBA_1$

M_0 : 8(2/1.08)	$M_{4a,b}$: 4(2/0.72)	M_9 : 8(2/1.44)
$M_{1a,b}$: 6(2/0.36)	$M_{5a,b}$: 6(2/1.44)	I_0 : 40uA
$M_{2a,b}$: 6(2/0.25)	M_6 : 2(1.5/0.75)	
$M_{3a,b}$: 6(2/0.25)	$M_{7,8}$: 4(2/1.44)	

Details for $PGBA_2$

Figure 4.3: Circuit diagram of PMOS Gain Boosting Amplifier (PGBA) with transistor sizes

4.3 Common mode feedback (CMFB) Circuit

A switched capacitor CMFB is used for the main opamp. This architecture is chosen to preserve the high DC gain of the opamp (If a resistive CMFB is used, the value of resistance needs to be very high). The value of the capacitance should be chosen appropriately as this capacitance loads the differential output. The circuit diagram for the CMFB circuit is shown in Fig. 4.4.

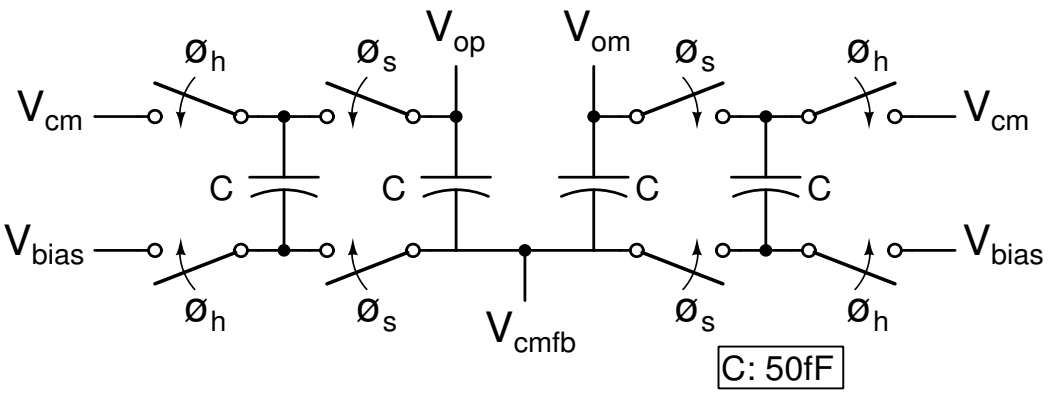


Figure 4.4: Circuit diagram of Common Mode Feedback

Chapter 5

MDAC Control Circuit

The MDAC control circuit generates the control signals to the MDAC depending on the input. It consists of a 2.5-bit flash ADC, bubble correction circuit, butterfly scrambler (for DEM) and thermometer to binary converter.

5.1 Flash ADC

The 2.5-bit flash ADC consists of two resistive ladders(7 resistors in each ladder) and 6 comparators as shown in Fig. 5.1.

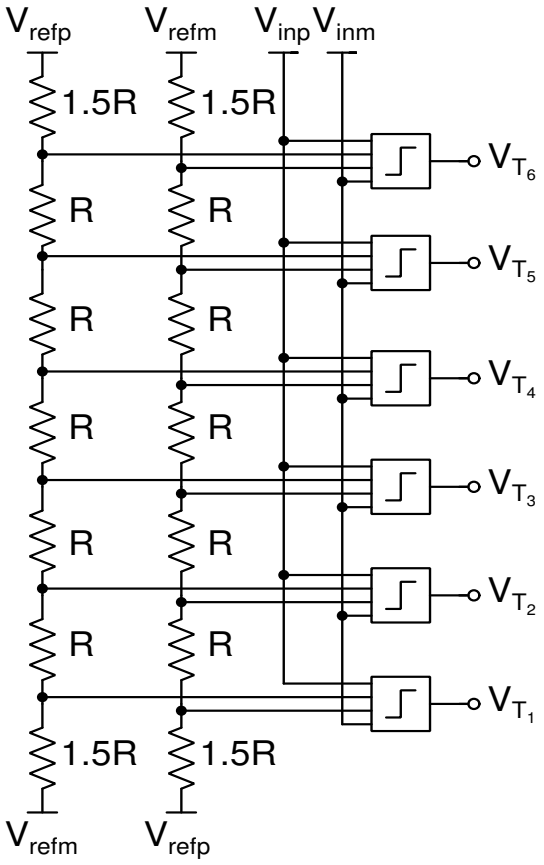


Figure 5.1: Circuit diagram of resistive divider used in 2.5 bit flash ADC

Comparator

The comparator contains transmission gates, capacitors, a latch and a D-FF. It uses dynamic latch to save the power. The dynamic latch is a fully differential latch which compares the analog signals presented at its inputs. Depending on the polarity of the differential input, the logic output is produced. If the voltage difference (V_{in+} - V_{in}) is positive the latch output will go high (1.8 V), otherwise its output will go low (0 V). The comparator operates in 5 phases. LATCH_A is the advanced version of LATCH which is to get rid of charge injection problem.

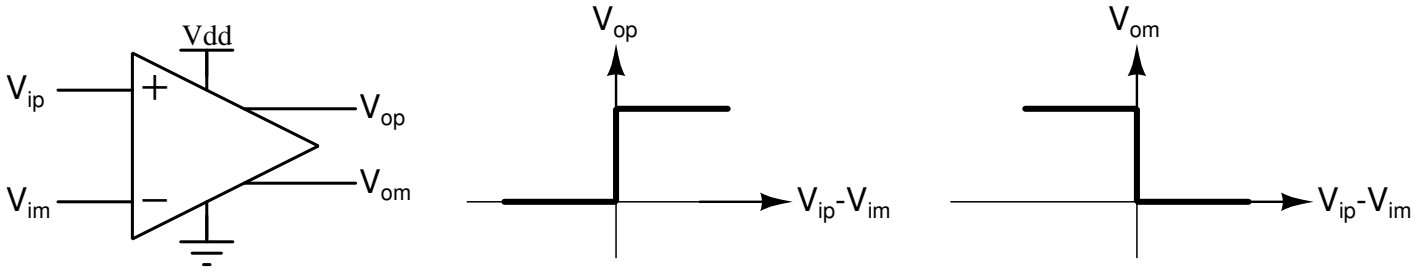


Figure 5.2: Ideal transfer function of latch

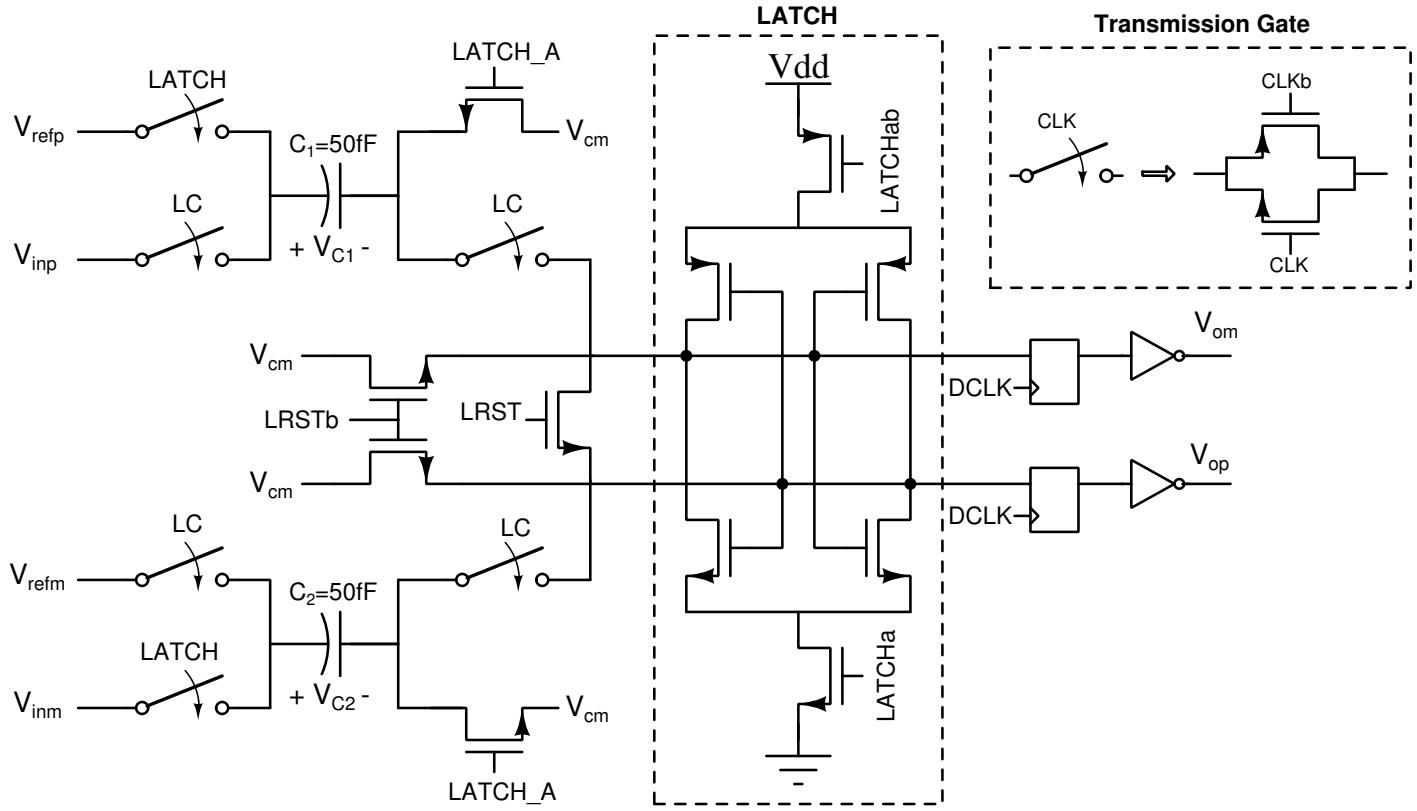


Figure 5.3: Circuit diagram of latched comparator used in 2.5 bit flash ADC

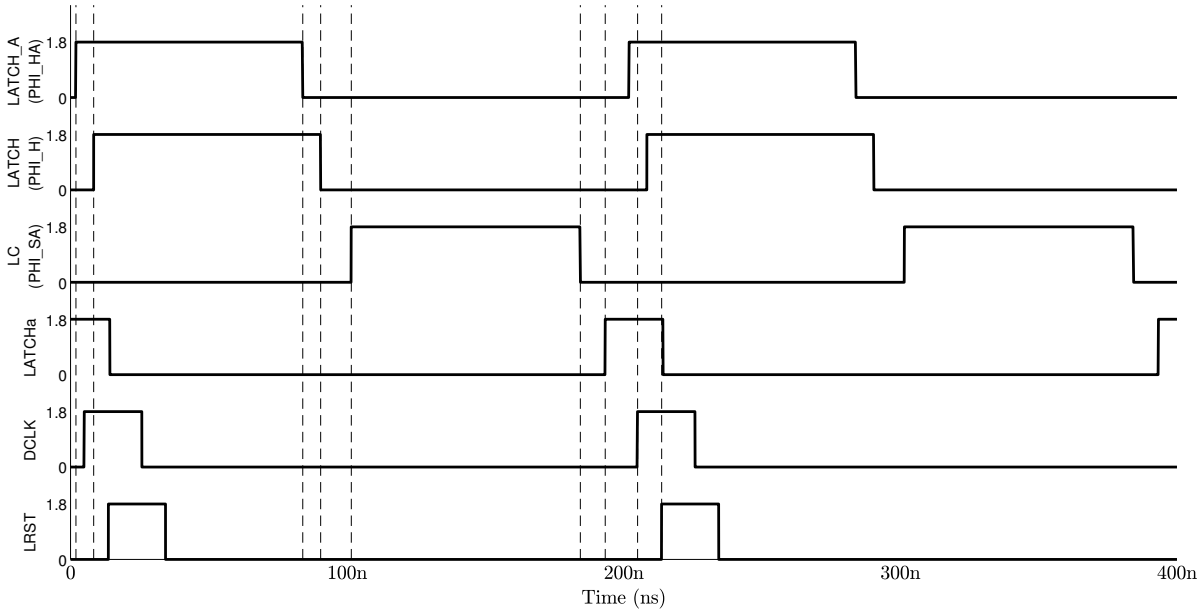


Figure 5.4: Timing diagram of the latched comparator

LATCH_A and LATCH phase During LATCH_A phase V_{cm} is applied to the top plate of capacitors C_1 and C_2. During LATCH phase V_{refp} and V_{refm} are applied to the bottom plates of capacitors C_1 and C_2 respectively. Hence the voltages across the capacitors (V_{C1} and V_{C2}) are $V_{refp} - Vcm$ and $V_{refm} - V_{cm}$ respectively.

LC phase

During this phase, input is applied to the capacitors on one side and the other side of capacitor is connected to the gate of the inverter which has some parasitic capacitance which is smaller than the physical capacitance. Hence the inputs to the latch ip and im are $V_{ip} - V_{refp} + V_{cm}$ and $V_{im} - V_{refm} + V_{cm}$ respectively and the differential input$(ip - im)$ is $(V_{ip} - V_{im}) - (V_{refp} - V_{refm})$.

LATCHa phase

During this phase the latch is connected to the supply and the latch starts regeneration of the output. If the differential input to the latch $(ip - im)$ is positive,the output is logic 1 otherwise the output is logic 0.

DCLK phase

DCLK comes after giving sufficient time for the latch to regenerate the output. During the rising edge of the DCLK, the D-FF latches the output. It is then buffered appropriately.

LRST phase

During this phase, latch is reset by shorting their outputs and connecting them to V_{cm}. This improves regeneration time of the latch.

5.2 Bubble Correction circuit

This is employed to correct the bubble which is an incorrect 0 in strings of 1 or incorrect 1 in strings of 0. This circuit uses majority of 3 logic for bubble correction.

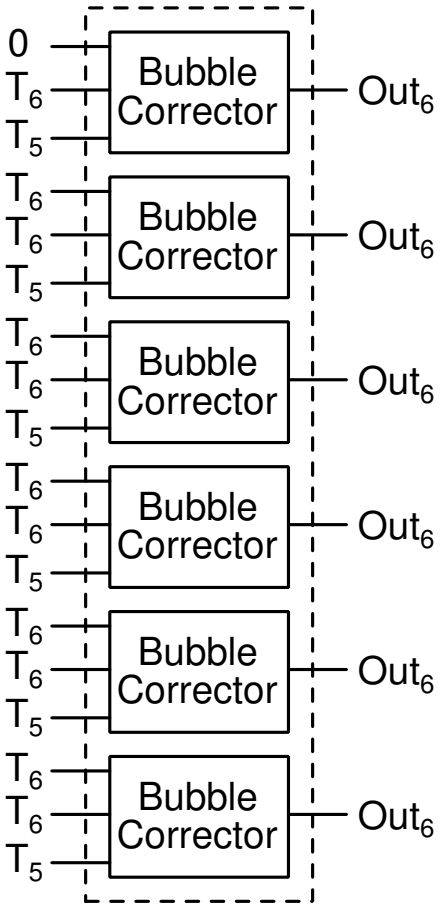


Figure 5.5: Block diagram of bubble correction logic

The bubble corrector takes 3 inputs and produces an output of logic 1 if there are more than 1 logic 1's at the input, otherwise the output is logic 0. The boolean expression for this logic is

$$OUT = A.B + B.C + C.A = \overline{(\overline{A}.\overline{B}).(\overline{B}.\overline{C}).(\overline{C}.\overline{A})} \tag{5.1}$$

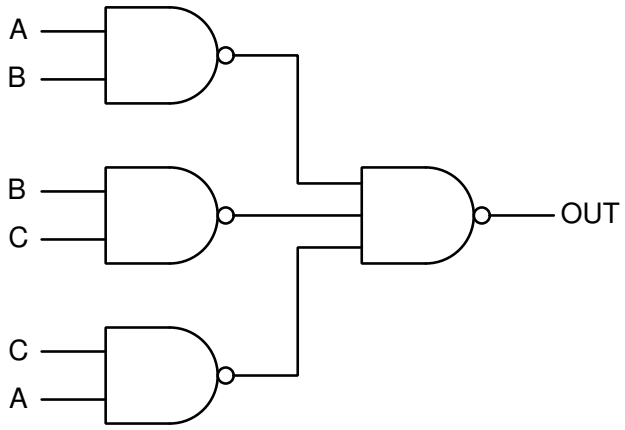


Figure 5.6: Circuit diagram of bubble corrector

A	B	C	OUT
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Table 5.1: Truth table of bubble corrector

5.3 Thermometer to binary encoder

The thermal codes are converted to binary codes by simply adding all the six thermal bits. Since full adder can take 3 bits at a time, we need totally 4 full adder to add all the 6 bits. The block level implementation of this is shown in Figure 5.7.

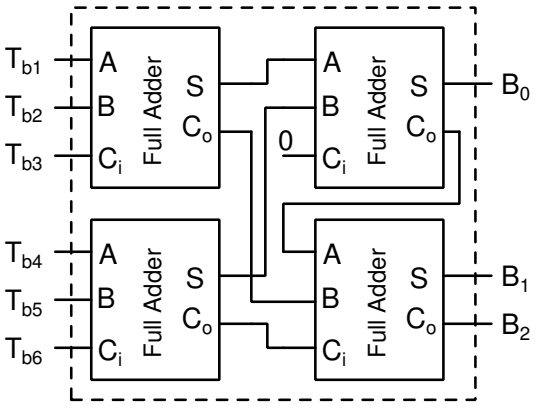


Figure 5.7: Block diagram of thermal to binary converter

The full adder is implemented using the following boolean expression

$$
\begin{aligned}
Carry(C_{i+1}) &= A.B + B.C_i + C_i.A \\
&= A.B + C_i.(A + B)
\end{aligned}
$$

$$
\begin{aligned}
Sum(S) &= A \oplus B \oplus C_i \\
&= A.\overline{B}.\overline{C_i} + \overline{A}.B.\overline{C_i} + \overline{A}.\overline{B}.C_i + A.B.C_i \\
&= \overline{C_{i+1}}.(A + B + C) + A.B.C
\end{aligned}
$$

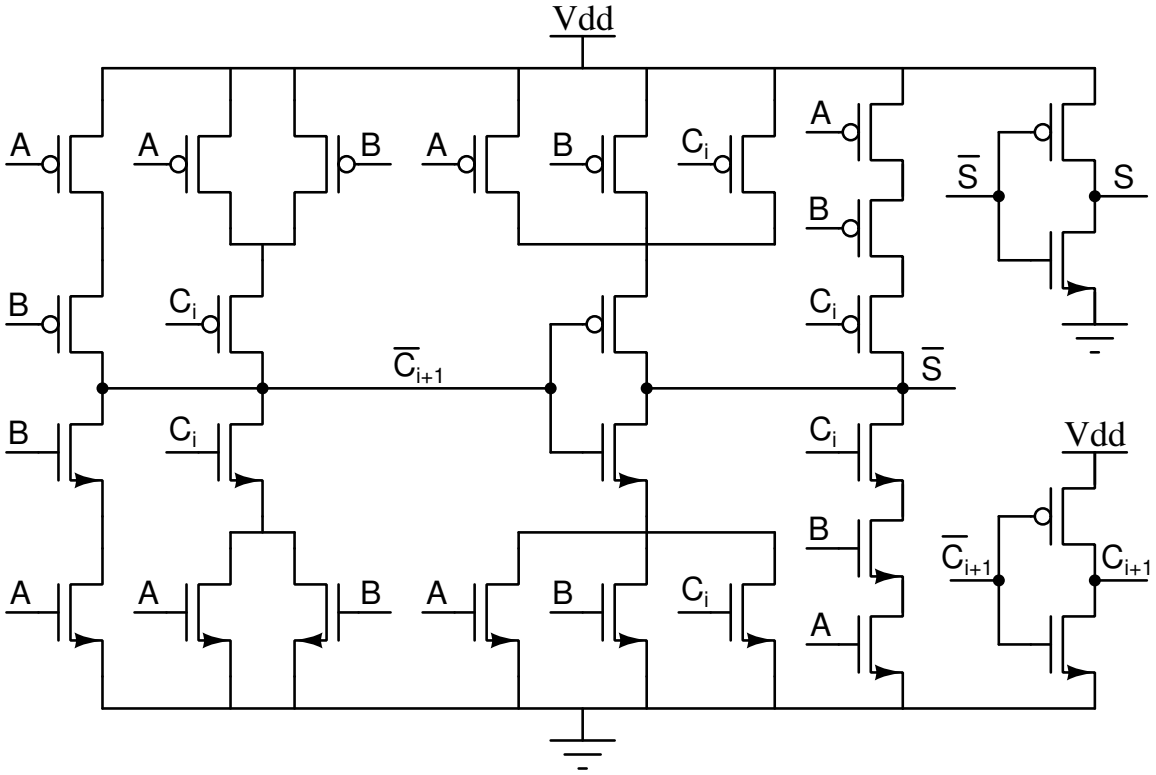


Figure 5.8: Circuit diagram of full adder

5.4 Dynamic Element Matching (DEM)

Dynamic linearity is one of the most important parameters in applications where dynamic range is high. It is commonly formulated as SFDR which is the difference in dB between full scale fundamental tone and the largest spur(mostly 3rd harmonic) in the ADC. Capacitor mismatch in stage 1 is a major problem in pipelined ADC which leads to reduced SFDR. In order to improve SFDR, the harmonic distortions are converted to white noise by randomizing the DAC connections. In each cycle randomly choose a set of connections. Performing DEM in the first stage of pipeline (m effective bits) improves the SFDR by $10.log(2^m - 1)$[3]. DEM takes input

from bubble corrector (6 inputs) and mixes with V_{DD} and *gnd* to generate 8 outputs which is given to MDAC.

5.4.1 Butterfly scrambler

To randomize the input to DAC, actually 8*8 switch matrix is needed. This results in 64 switches and 40320 (8!) switching combinations which is very cumbersome to design. Hence an 8-line butterfly scrambler is used for simplicity. This uses 7 switches and generates 128 combinations ($2^1 * 2^2 * 2^4$).

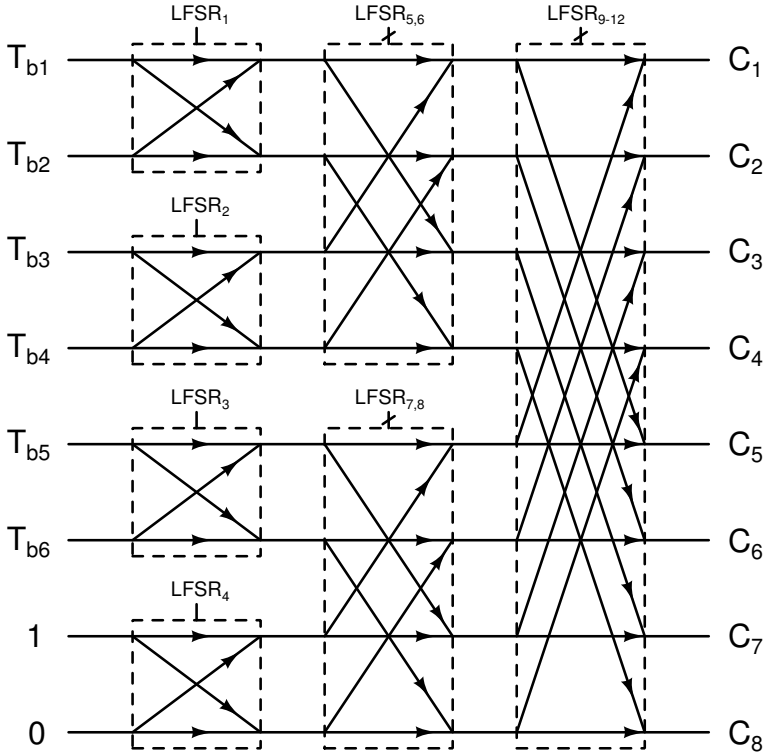


Figure 5.9: Logic diagram of butterfly scrambler

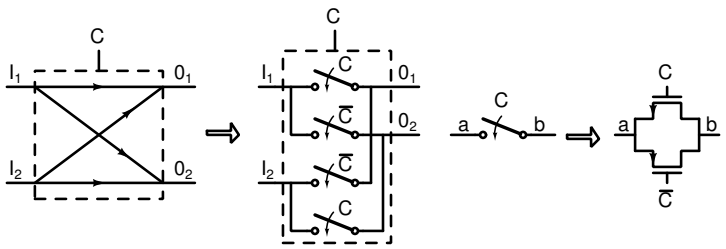


Figure 5.10: Circuit diagram of switch used in butterfly scrambler

5.4.2 LFSR

LFSR is a shift register whose input bit is a linear function of its previous state(s). It contains D-FF, and few EX-OR gates. An LFSR is defined by a polynomial. A maximum length polynomial of degree n will have $2^n - 1$ different states. Such polynomials are called primitive polynomials because they will give us maximum length periods when shifting. To generate maximum possible period of $2^{15} - 1$ with 15 D-FF, one of the polynomials which requires minimum number of taps (EX-OR gates) is $X^{15} + X^{14} + 1$. To avoid LFSR going to locked state (when all states are 0), an all zero detector is used to load a logic 1 if it goes to locked state. The all 0 detector is a nor logic whose output is 1 when all states are 0.

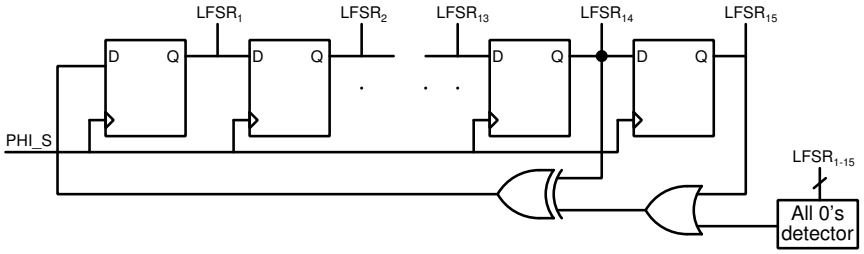


Figure 5.11: Circuit diagram of LFSR

Chapter 6

Delay and Digital Error Correction

6.1 Delay

Since the bits corresponding to an input are processed at different instance of time by different stages, the bits must be delayed appropriately before addition. To do this, D-FFs are used as delay elements. The up arrow in the timing diagram of Figure 6.1 shows the availability of output bits from respective stage for an input. The final output comes from stage 6.

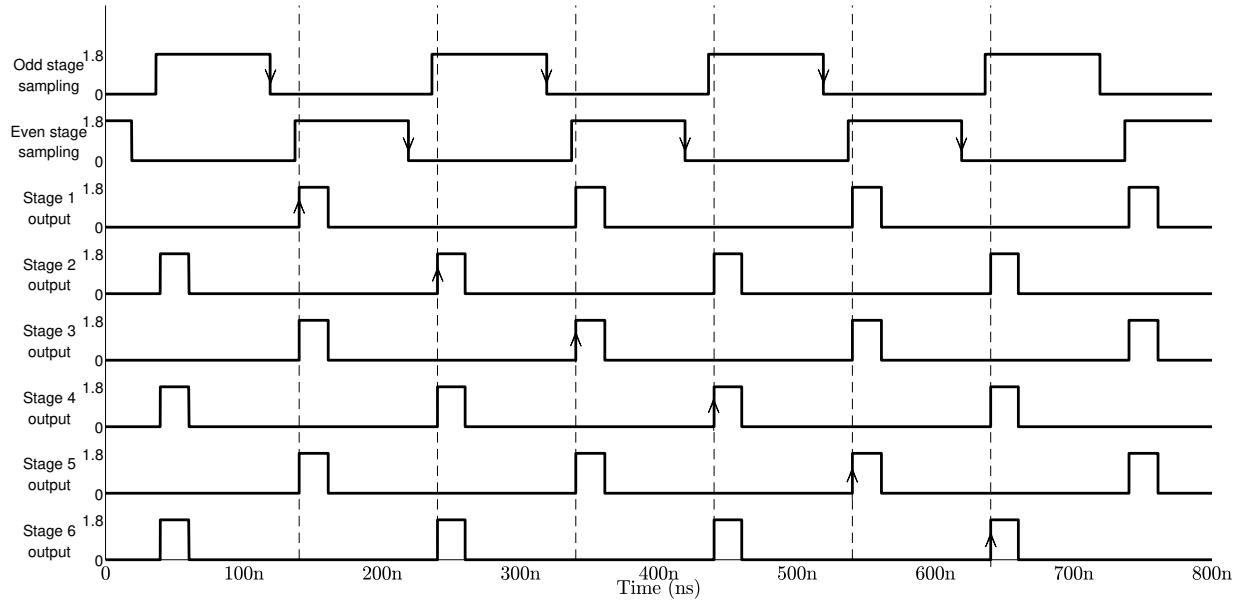


Figure 6.1: Timing diagram of bits availability from each stage

Output bits from odd stages comes at $DCLK_1$ while the output bits from even stages comes at $DCLK_2$. The output bits from each stage should be delayed appropriately to make it in sync with the output from final stage (stage 6) which comes at $DCLK_2$. Figure 6.2 shows the delay logic used.

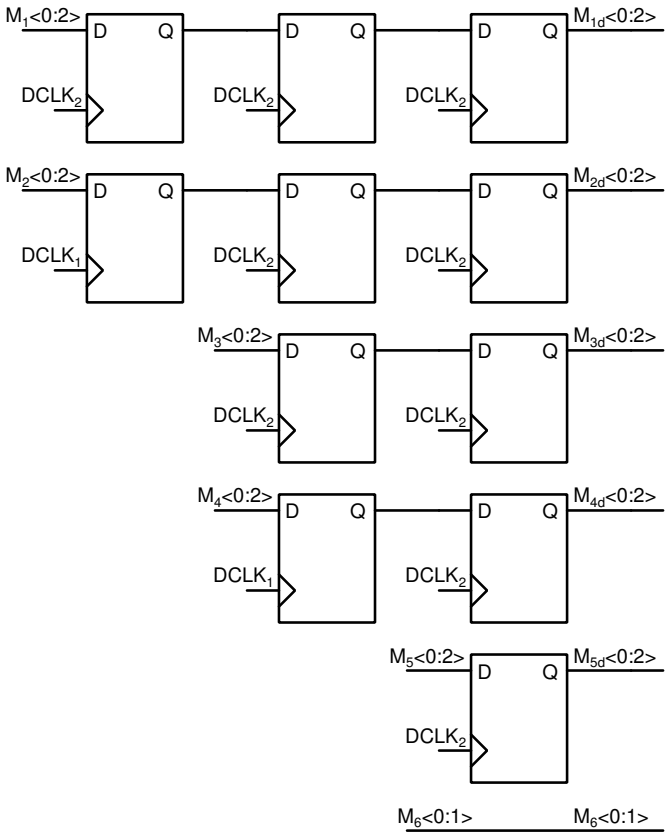


Figure 6.2: Block diagram of delay logic

6.2 Digital Error Correction

Digital Error Correction is employed to greatly reduce the accuracy requirement of the flash ADC (and thus individual comparators). It is to be noted that the 3-bit residue at the summation node has a dynamic range of one-eigth of the reference (except at the extremes), but the gain is only 4. Hence the input to stage 2 occupies only half the dynamic range (when there is no error in the first stage).

If one of the comparators in the first stage has a significant offset and if an analog input close to the trip point is applied, it leads to incorrect 3-bit code and hence incorrect DAC output which results in different residue. As long as the amplified residue doesn't over-range the subsequent stage, then the code generated by the subsequent stage when added with the incorrect 3-bit code, it will give the correct ADC output code. DEC will not correct errors for errors made in the final 2-bit flash stage. Any error made at the conversion is suppressed by the large cumulative gain (4^5) preceding the 2-bit flash. Thus the final stage only needs to be more than 2-bit accurate.

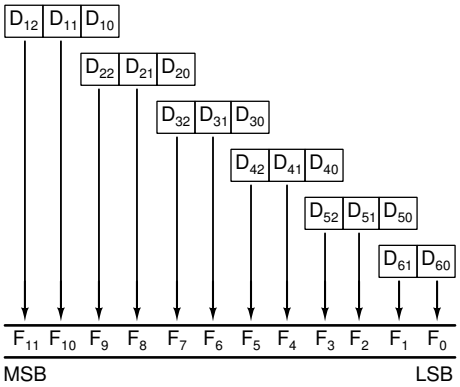


Figure 6.3: DEC algorithm, D_{xy} is the output of stage x where y=0 is the LSB

Since the interstage gain is reduced by half, and the bottom reference point is shifted by 0.5 LSB, the bits are added with 1 bit overlap. Henc the overall accuracy of the pipelined ADC is $(N_1 + N_2 + N_3 + N_4 + N_5 + N_6 - 5)$ bits which is 12 bits, where N_i is the bits from i^{th} stage. This is illustrated in Figure 6.3.

Chapter 7

Clock generation

7.1 MDAC clocks

This block generates the non-overlapping sampling and amplification clocks required for proper operation of MDAC. The non-overlapping period is adjusted by varying the even number of inverters used in forward path.

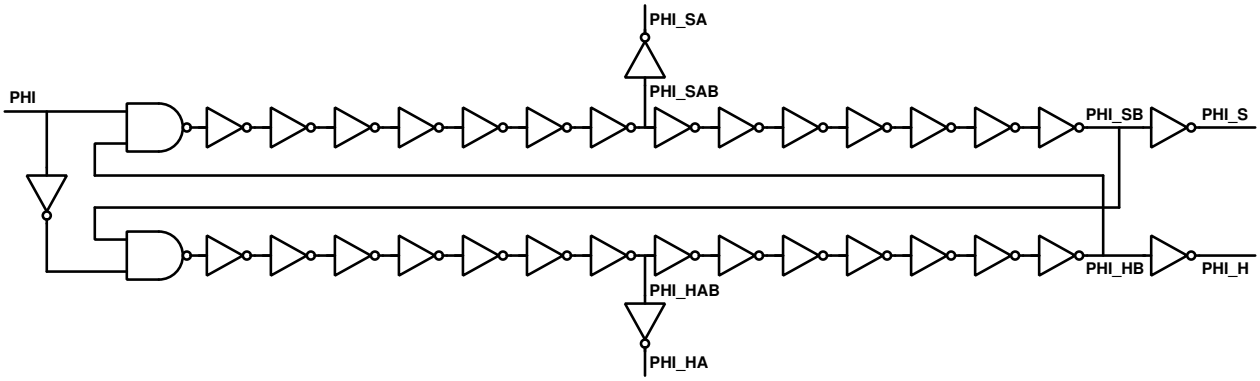


Figure 7.1: Circuit diagram of non-overlapping clock generator for MDAC

7.2 Latch clocks

The latch samples the input in same time as the MDAC samples the input. The latch should start regenerating once the input is sampled onto latch (at the falling edge of PHI_S). Since the D-FF is rising edge triggered, complementary signal(PHI_SB) should be given as clock. For proper operation of latch, sufficient time should be given for the latch to regenerate, after it sample the input. After giving some time for regeneration, the latch output should be sampled in D-FF using DCLK. Once the D-FF sample the latch, the latch should be reset using LRST to improve regeneration time of latch. Two such latch clock generators are needed for odd and even stages.

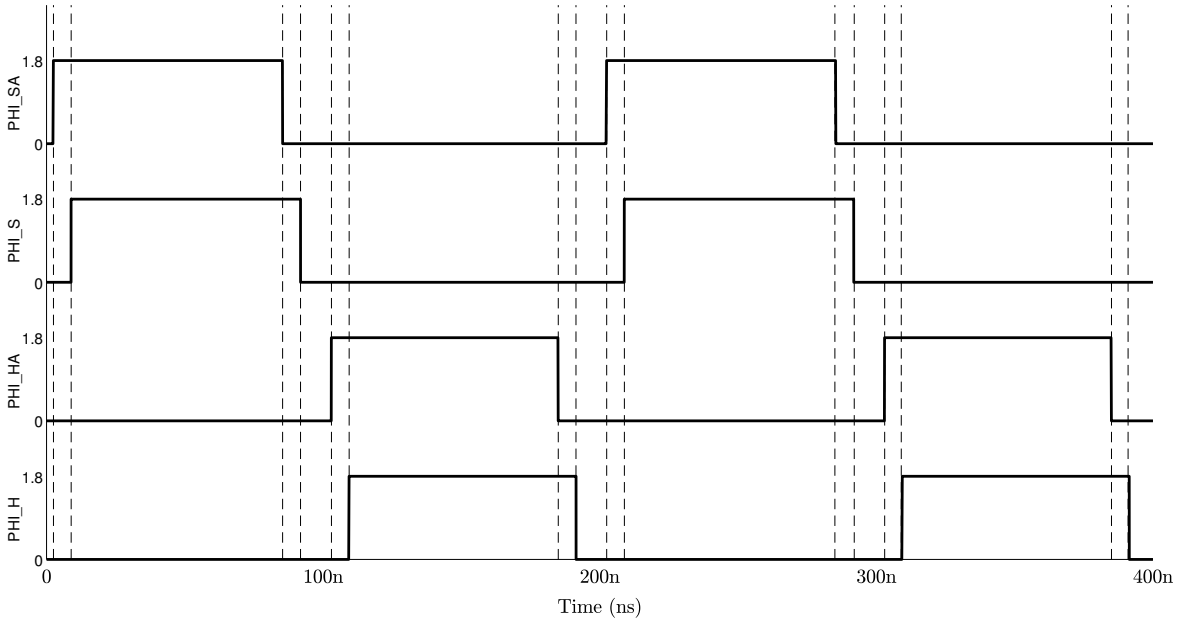


Figure 7.2: Sampling and hold clocks used in MDAC

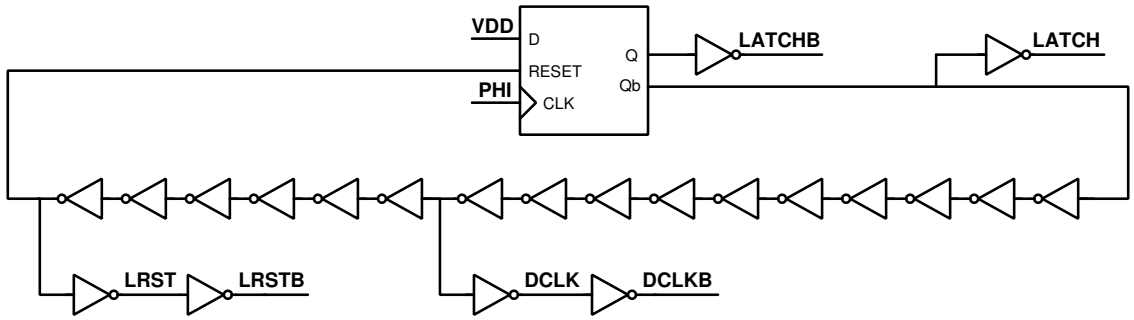


Figure 7.3: Circuit diagram of clock generator for latch

7.3 Clock buffers

The clocks generated above have to drive huge load (6 stages). Hence to provide the driving capability, they are buffered using the clock buffer shown in Figure 7.4.

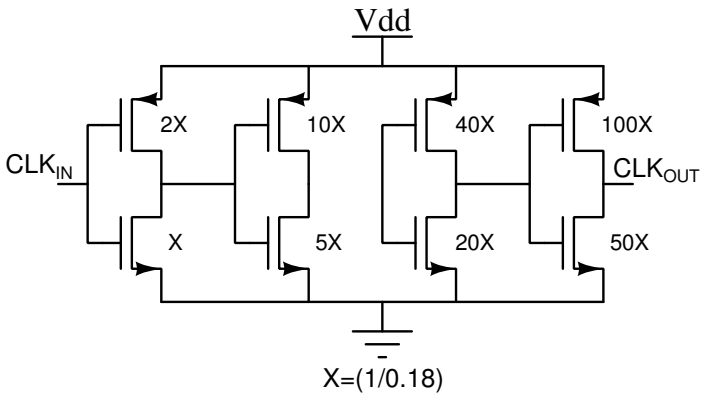


Figure 7.4: Circuit diagram of clock buffer

Chapter 8

Layout and Simulation results

8.1 Layout

The pipelined ADC is laid out in SCL 180nm technology using CADENCE virtuoso tool. The dimension of top level layout is 1mm x 1.7mm.

The layout of capacitor array is very crucial as it affects the gain of the amplifier. The capacitors are laid out in a common centroid fashion by adding dummies which nullifies the mismatch due to chip gradient.

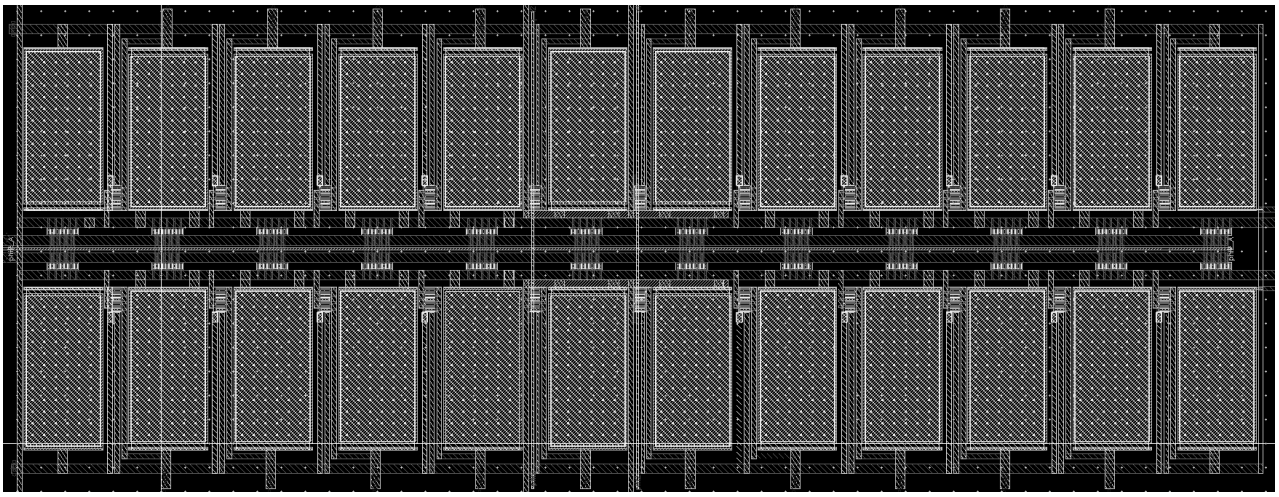


Figure 8.1: Layout of stage 1 capacitor array (0.26mm X 0.1mm)

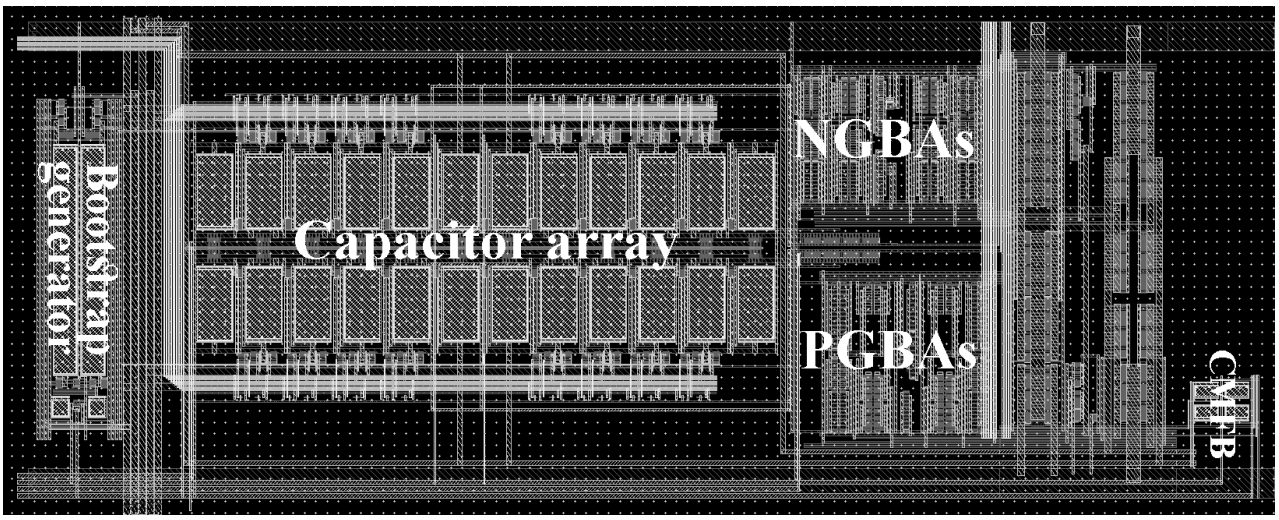


Figure 8.2: Layout of stage 1 MDAC (0.55mm X 0.21mm)

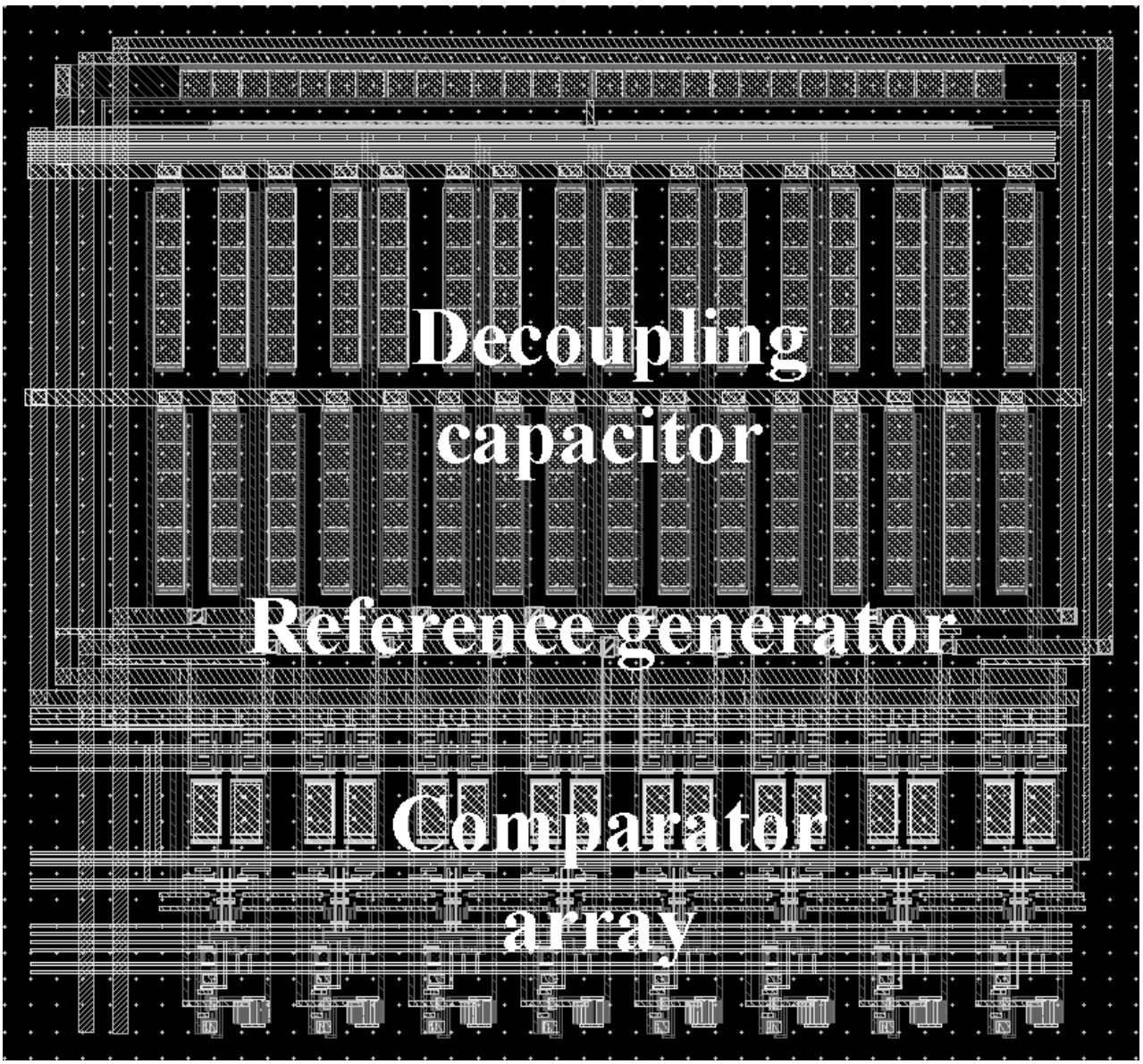


Figure 8.3: Layout of flash ADC (0.21mm X 0.19mm)

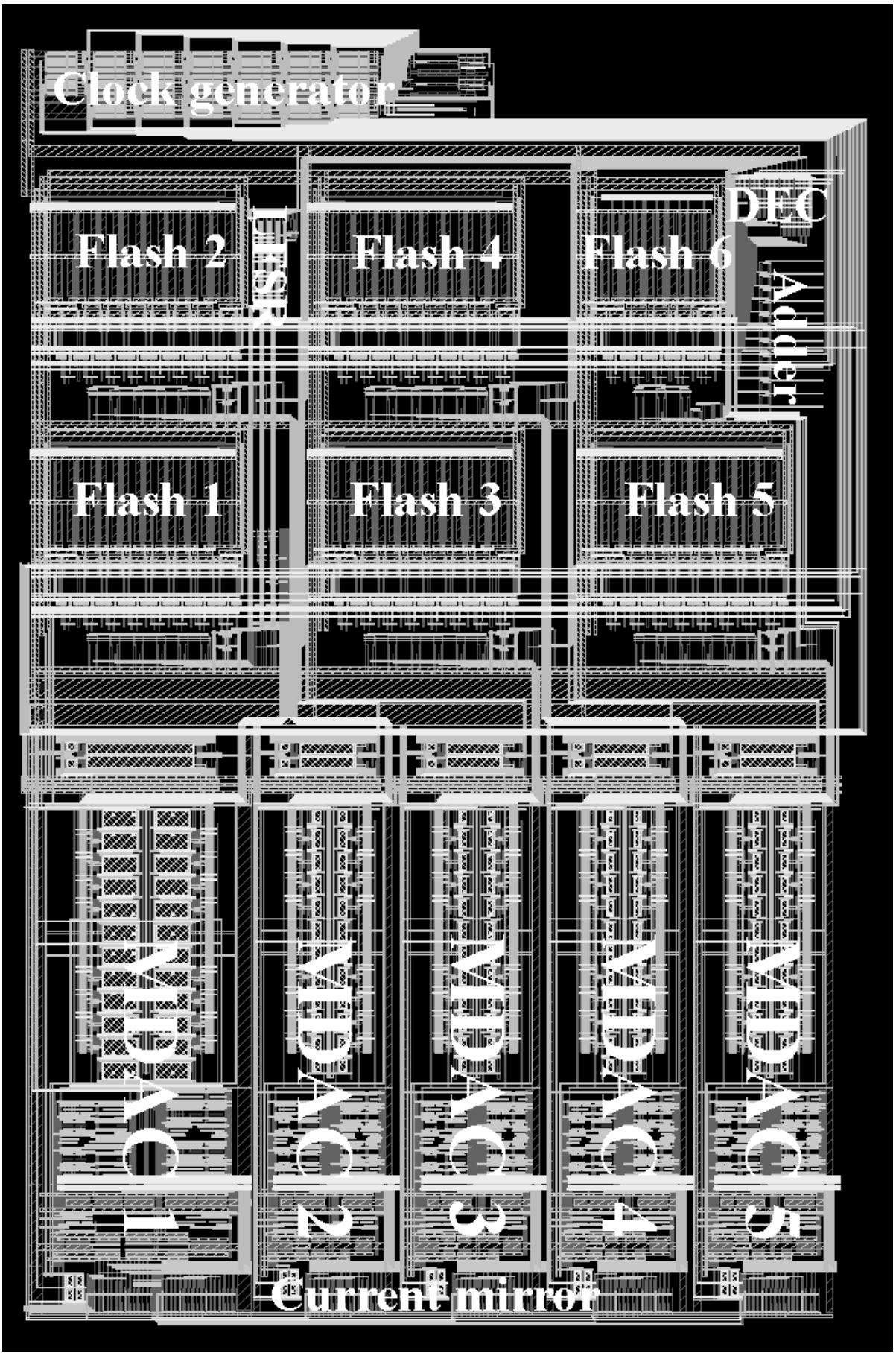


Figure 8.4: Top level layout of pipelined ADC (1mm X 1.5mm)

8.2 Simulation results in SCL 180nm

The simulation was carried out with the extracted view (C+CC) after layout. The simulation results of extracted view closely matches with that of the simulation results of schematic.

8.2.1 Simulation results of flash ADC

The flash ADC is excited with a differential 2 VPP. The reference voltages of flash are 0.4 V and 1.4 V . Table 8.1 compares the simulation result of extracted view after layout.

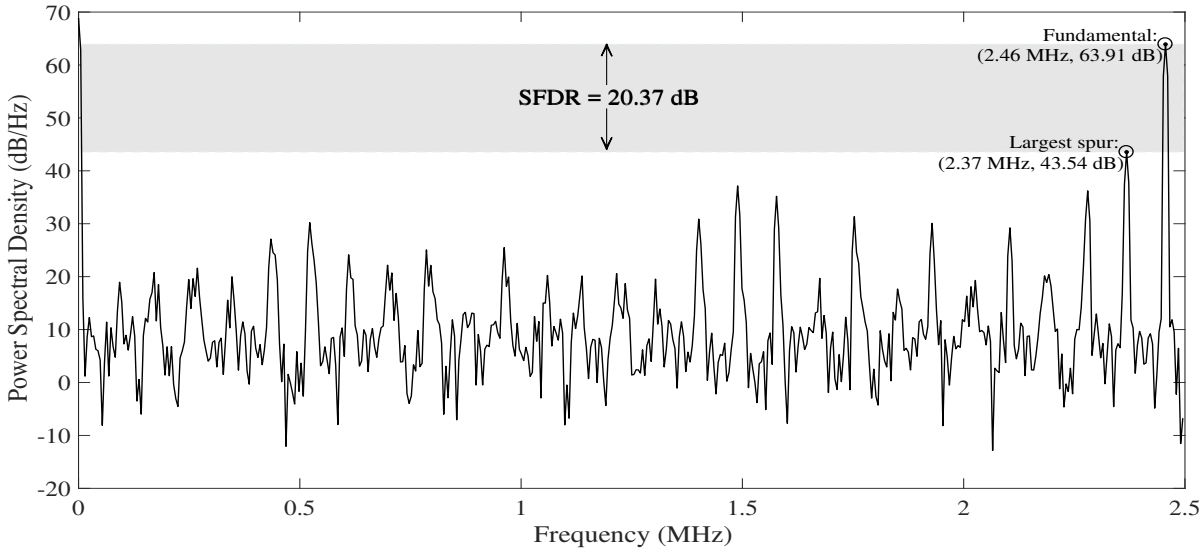


Figure 8.5: Output power spectral density plot of flash ADC for 2 V_{PP} input at 2.46 MHz with nominal process corner(TT) at 27 °C.

Parameter	SS 0°C	SS 70°C	TT 27°C	FF 0°C	FF 70°C
SNR	17.30 dB	17.32 dB	17.25 dB	17.18 dB	17.27 dB
ENOB	2.58 bits	2.58 bits	2.57 bits	2.56 bits	2.58 bits
SFDR	20.89 dB	20.85 dB	20.37 dB	20.54 dB	20.73 dB
Power	2.33 mW	2.21 mW	2.15 mW	2.11 mW	2.30 mW

Table 8.1: Comparison of various parameters of flash ADC for 2 V_{PP} input at 2.46 MHz under different process corners

8.2.2 Simulation results of entire pipelined ADC

The ADC is excited with an input of full scale amplitude (2 VPP differential input) at 2.46 MHz. The output power spectral density plot is shown in Figure 8.6. The simulation results of extracted view after layout for various process corners are shown in Table 8.2.

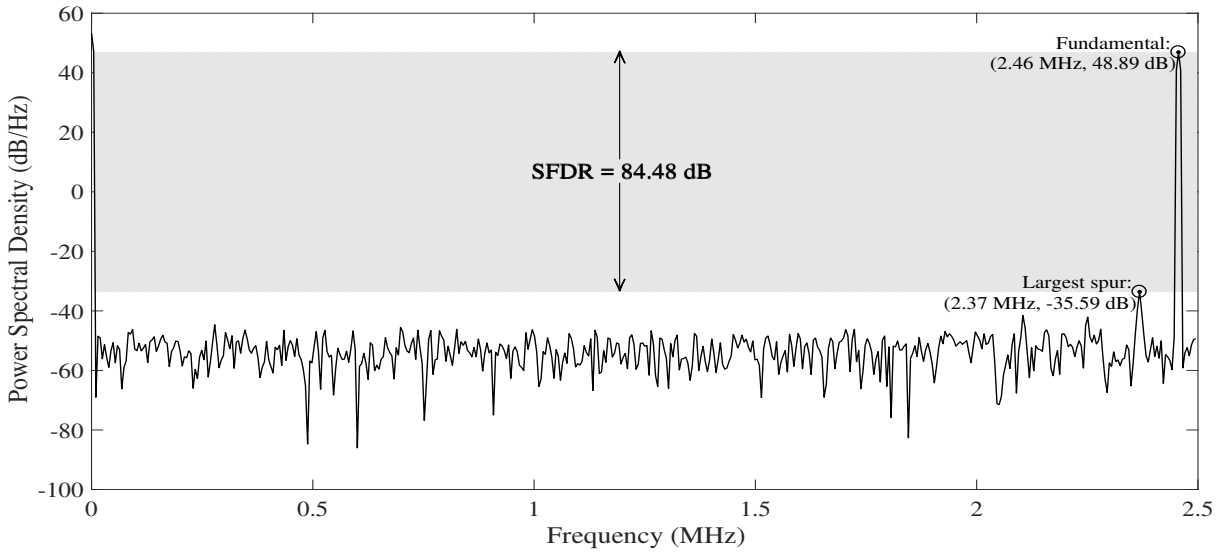


Figure 8.6: Output power spectral density plot of pipelined ADC for 2 V_{PP} input at 2.46 MHz with slow process corner (SS) at 0 °C.

Parameter	**SS 0°C**	**SS 70°C**	**TT 27°C**	**FF 0°C**	**FF 70°C**
SNR	71.75 dB	73.73 dB	73.68 dB	72.90 dB	72.43 dB
ENOB	11.62 bits	11.95 bits	11.83 bits	11.82 bits	11.74 bits
SFDR	84.48 dB	86.13 dB	85.83 dB	84.56 dB	84.73 dB
Power	51.12 mW	51.05 mW	52.17 mW	51.3 mW	57.42 mW

Table 8.2: Comparison of various parameters of pipelined ADC for 2 V_{PP} input at 2.46 MHz under different process corners

Parameter		**Given in thesis**		**Simulated**	
Corner	**Temp.**	**SNDR(dB)**	**ENOB(bits)**	**SNDR(dB)**	**ENOB(bits)**
SS	0 °C	73.60	11.91	71.75	11.62
SS	70 °C	72.90	11.80	73.73	11.95
TT	27 °C	73.20	11.85	73.68	11.83
FF	0 °C	74.00	11.98	72.90	11.82
FF	70 °C	73.40	11.88	72.43	11.74

Table 8.3: Comparision of performance paramters of ADC with result from thesis

8.3 Conclusion

A 6 stage 12-bit pipelined ADC operating at 5 MSPS sampling rate for 2VPP differential input has been designed and laid out in SCL 180 nm technology. The thermal noise problem is resolved by choosing sufficiently large capacitance, the capacitor mismatch problem is resolved by using Dynamic Element Matching (DEM) and common centroid layout, comparator offset problem is resolved by using Digital Error Correction (DEC) technique. The latency of the ADC is about 3 clock cycles. Table 8.2 show that the ADC has a resolution close to 12-bit and the power consumption is close to 52 mW.

8.4 Future work

The opamp in MDAC is useful only during amplification phase. Hence opamp sharing can be employed to save power and space.

References

[1] Shravan Siddartha N, "Design of 12-bit 40 MSPS Pipelined Analog to Digital Converter", Dual Degree thesis, IIT Madras, May 2011

[2] Saisundar. S, Shan Jiang, Kevin T. C. Chai, David Nuttman and Minkyu Je, "14-bit 1MS/s Cyclic-Pipelined ADC" *International Journal of Electronics and Communication Engineering*, Vol.7, No.9, 2013

[3] Mohammad Taherzadeh-Sani and Anas A.Hamoui, "Analysis of Dynamic Element Matching(DEM) in Pipelined ADCs", *IEEE International Symposium on Circuits and Systems*, May, 2006

[4] Pseudo Random Number Generation Using Linear Feedback Shift Registers, `https://www.maximintegrated.com/en/app-notes/index.mvp/id/4400`