Design of Components for High Speed, Wide Bandwidth CTDSM

A Project Report

submitted by

Narasimhan Rajagopal

in partial fulfilment of the requirements

for the award of the degree of

Master Of Technology

Under the guidance of

Dr. Y. Shanthi Pavan

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

May 2015

THESIS CERTIFICATE

This is to certify that the thesis titled **Design of Components for High Speed, Wide Bandwidth CTDSM**, submitted by **Narasimhan Rajagopal**, to the Indian Institute of Technology, Madras, for the award of the degree of **Master of Technology**, is a bona fide record of the research work done by him under our 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. Y. Shanthi Pavan
Project Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date:

ACKNOWLEDGEMENTS

First and foremost I would like to dedicate this thesis to my project guide Prof. Shanthi Pavan who was graceful enough to accept me as his student. It was through his wonderful courses that I developed an interest towards analog circuits. He was instrumental in driving the project to its respectable state as it currently stands.

I would like to thank Prof. Aniruddhan for his interesting course on Analog Integrated Circuits. I express my thanks to Prof. Nagendra Krishnapura for his video lectures and for being our M.Tech faculty advisor. I would like to thank Prof. Nitin for his intriguing course on Digital IC design and the corresponding lab. I convey my thanks to Chaitra, Rahul and Sridhar from IBM and Prof. Ravishankar for their thought provoking course on Advanced topics in VLSI. I would like to thank Prof. C.S.Ramalingam for the interesting DSP course. I would also like to thank Prof. Karmalkar, Prof. Anjan and Prof. Amitava Das Gupta for their courses on VLSI Technology, Semiconductor Device Modelling and MOS Device Modelling. I wish to thank Prof. Kamakoti for his lively courses on DSTTD(Digital System Testing and Testable Design) and Professor Vishwanathan for his logically intriguing course on CAD for VLSI. I express my gratitude to my mentors Ankesh and Amrith for their suggestions on various technical aspects of the project. I would like to thank my project team-mate Sreejish without whom this project would not be complete. I would also like to thank Ashwin and Prasanth for the numerous technical discussions.

Thanks to my lab mates Vaishnavi, Saravana Kumar, Praveen Kumar Varma, Naga, Sandeep, Pradeep Kola, Kamlesh, Rakshit and Aparna for the numerous fun we had in lab. Special thanks to Praveen for proof-reading the thesis and suggesting several changes. I would like to thank my 2013 M.Tech batch-mates without whom these 2 years of my life would not be memorable. I would also like to thank Janaki mam for her help in solving PC related issues.

This acknowledgement is incomplete without thanking my Parents and Grandparents for their enormous love and affection. But for their motivation I would not have cracked GATE and ended up writing this thesis.

And finally I wish to thank the reader(your Highness) for taking time to read this thesis. Hope its contents are helpful. Kindly pardon me for any mistakes in this thesis.

ABSTRACT

KEYWORDS: CTDSM ; High Speed; DWA; Wide Bandwidth; Clockless RZ DAC.

Continuous Time Delta Sigma Modulator(CTDSM) is a rapidly developing field in the analog world, especially in the Analog to Digital Converter(ADC) landscape. CTDSMs provide a power efficient way to convert analog signal to digital. Due to oversampling constraints these are usually restricted to low and medium bandwidths(< 20 MHz) and medium speeds(< 1 GHz). But of-late CTDSMs are being designed for high speed, wide bandwidth applications. In order to achieve high bandwidths it is necessary to design at low Over Sampling Ratios(OSR).

This work deals with the design of specific components in 65 nm technology that are used in implementing a 2.6 GHz CTDSM with a bandwidth of 100 MHz. A 17 level Flash ADC is designed with low latency. In order to reduce the effect of the mismatch of DAC elements a new high speed Data Weighted Averaging(DWA) block is implemented. The clock generation network and the output driver network are designed. A novel clockless Return-to-Zero(RZ) Digital to Analog Converter(DAC) to be used in the fast feedback path is also discussed.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ADC	Analog to Digital Converter
CA	Continuous Amplitude
DA	Discrete Amplitude
CT	Continuous Time
DT	Discrete Time
AC	Alternating Current
DC	Direct Current
DSM	Delta Sigma Modulator
DEM	Dynamic Element Matching
CTDSM	Continuous Time Delta Sigma Modulator
DTDSM	Discrete Time Delta Sigma Modulator
DAC	Digital to Analog Converter
DWA	Data Weighted Averaging
DNL	Differential Non-Linearity
INL	Integral Non-Linearity
FFT	Fast Fourier Transform
MSA	Maximum Stable Amplitude
NRZ	Return to Zero
RZ	Non Return to Zero
NTF	Noise Transfer Function
$\mathbf{Q_{Noise}}$	Quantization Noise
STF	Signal Transfer Function
SNR	Signal To Noise Ratio
SQNR	Signal To Quantization Noise Ratio
OBG	Out of Band Gain
PSD	Power Spectral Density

OSR	Over Sampling Ratio
IBQN	In-Band Quantization Noise
R+C+CC	Extraction of parasitic Resistance, Capacitance and Coupling Capacitance
C+CC	Extraction of parasitic Capacitance and Coupling Capacitance
OP-AMP	Operational Amplifier

CHAPTER 1

Introduction

1.1 Motivation

Due to the advancement in computers, most of the processing done today are digital in nature. However most of the real life data that requires processing like temperature, heartbeat, sound etc., are in the analog domain and they need to be first converted to digital domain for processing. Therefore, there is an increasing demand for high speed and high precision Analog to Digital Converters[1, 2].

There are many ADC architectures like Flash ADC, SAR ADC, pipeline ADC and Delta Sigma modulator. Flash ADCs are used at high speeds. However due to the hardware complexity they are restricted to lower resolutions(<6-bit). For higher resolutions, SAR and pipeline ADCs are used. SAR ADC is highly hardware efficient but takes more time due to the fact that it uses a single comparator. Pipeline ADCs are faster than SAR. However as resolution increases both SAR and pipeline ADCs require tighter constraints(offset, noise), making it difficult to design, especially at high speeds. Therefore for higher resolution Delta Sigma modulator is attractive due to the fact that the comparator accuracy(offset) constraints are relaxed by the loop filter gain. CTDSM is especially attractive due to its inherent anti-aliasing capability and resistive input load. However in high speed CTDSM, clock jitter is a major concern. Using a multi-bit ADC in the CTDSM loop, relaxes the jitter constraints but introduces the problem of DAC non-linearity due to the mismatch of DAC elements. The DAC mismatch can be mitigated by using Dynamic Element Matching(DEM). Among the various DEM methods Data Weighted Averaging(DWA) is widely used due to its hardware efficiency. But DEM methods(including DWA) are difficult to incorporate in high speed CTDSMs due to their huge delays(generally more than 500 ps[1]).

1.2 Thesis Organization

The thesis os organized as follows:

Chapter 2 deals with basics of ADCs and CTDSMs. It also gives an overview of the CTDSM architecture used in this work.

Chapter 3 discusses the design of 17 level ADC and issues related to its implementation.

Chapter 4 highlights the design of new high speed DWA.

Chapter 5 discusses the various blocks of the clock generation and output driver network.

Chapter 6 presents a novel clockless Current Steering(CS) Return to Zero(RZ) Digital to Analog Converter(DAC).

Chapter 7 concludes the thesis.

CHAPTER 2

Basics concepts of Delta Sigma Modulator

2.1 ADC Basics

2.1.1 Sampling and Quantization

ADCs basically convert Continuous Time(CT), Continuous Amplitude(CA) signals to Discrete Time(DT), Discrete Amplitude(DA). Generally, CT, CA is converted to DT, CA first using a sampler(sample and hold circuit) and then converted to DT, DA using a quantizer. Some quantizers also have inbuilt sampler and do not require an explicit sampler. In theory sampling is an operation where there is no loss of information provided the baseband signal, that is being sampled, is band limited and satisfies Nyquist sampling Theorem given by

$$f_s > 2f_b \tag{2.1}$$

where f_s is the sampling rate and f_b is the bandwidth of the baseband signal.

The signal can be perfectly reconstructed from its samples by using a low pass filter of bandwidth $f_s/2$. If the signal has frequencies above f_b then there will be aliasing(spectral folding) and there will be loss of information(signal cannot be perfectly reconstructed). Even though the signal bandwidth is less than $f_s/2$ it will be accompanied by noise which is not band limited and leads to aliasing. Hence generally the sampler is always preceded by an anti-aliasing filter(low pass filter with bandwidth $f_s/2$) to filter out the noise and other interfering signals that cause aliasing. Quantization is a non-linear operation and leads to loss of information. Hence sampling is done first followed by quantization. If quantization is done first then the input to the sampler is not band limited(quantization being non-linear generates harmonics) and leads

to aliasing. The basic unit of a quantizer is a comparator. It compares the input with a reference(threshold) and outputs a 1 or 0 based on whether the input is greater or smaller than the reference.

A 2 level ADC(or single bit ADC) uses a single comparator(single reference) and the output has 2 levels. A N+1 level ADC requires N references and produces a output that has N+1 level.

In order to match the digital processing speeds of several GHz the ADCs should be able to provide data at comparable speeds. The fastest ADC is the flash ADC, which due to its parallel nature gives output almost instantaneously(in a flash and hence the name flash ADC). But it requires lot of hardware. A N-bit Flash ADC($2^N + 1$ levels) requires 2^N comparators. So there is an exponential increase in hardware complexity for every bit. So flash ADCs are not built beyond 6 bits due to the hardware complexity. Moreover since flash ADCs are open loop systems they have their own performance issues.

For higher precision(more than 6 bits), other types of ADCs like SAR or over-sampled ADCs are used. SAR ADCs have been used for a long time now but they require an anti-aliasing filter upfront and have Switched capacitor input which is difficult to drive. Moreover they demand accurate matching of analog components and often require trimming and calibration techniques that further increases the power consumption. CTDSM(Continuous time Delta Sigma Modulator), which is a type of oversampled converter, is gaining importance as it provides high precision(high dynamic range) over a wide bandwidth while being power efficient.

2.1.2 Quantizer Characteristics

The Characteristic of a quantizer is shown in Figure. 2.1.

Assuming that the input is within the quantizer range(between $-N\Delta$ to $+N\Delta$ for 2N+1 level mid-tread quantizer[3]), the quantization error is restricted to within $\pm\frac{\Delta}{2}$ where Δ is the step size(or LSB) of the quantizer. Assuming that the error is uniformly dis-

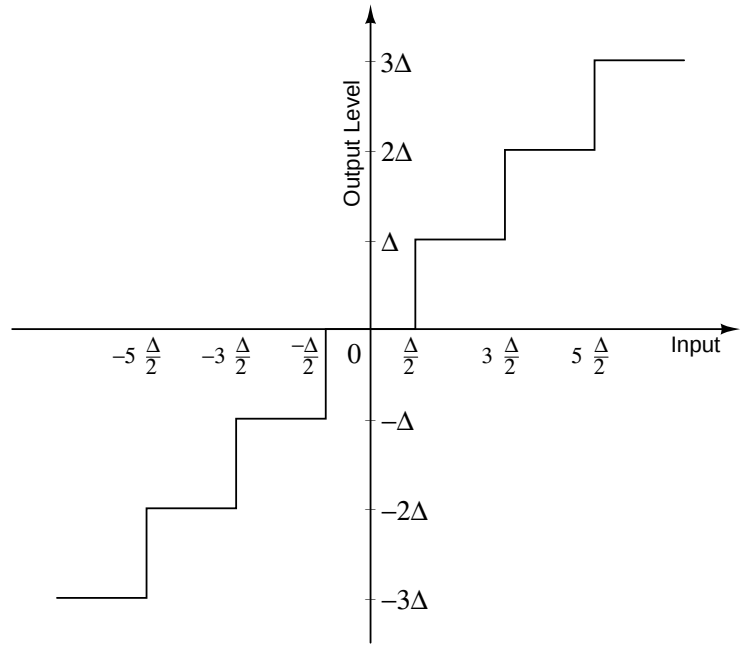


Figure 2.1: Transfer characteristics of a Quantizer

tributed the variance of the quantization noise is $\frac{\Delta^2}{12}$. The PSD is shown in Figure. 2.2.

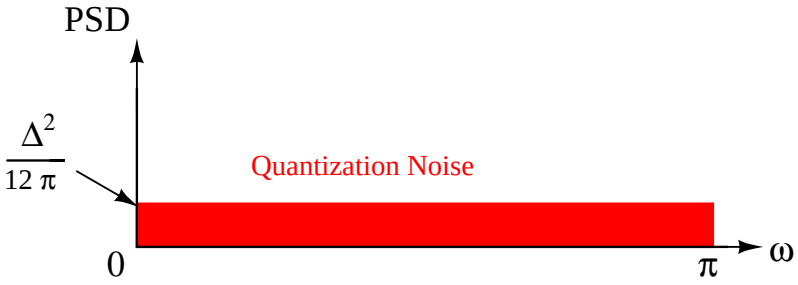


Figure 2.2: PSD of Quantization Noise

The following assumptions are made to obtain the given PSD.

- The quantization error is a noise that is independent of signal.
- Quantization error is uniformly distributed between $-\frac{\Delta}{2}$ and $+\frac{\Delta}{2}$.
- The quantization error is independent from sample to sample.

In essence the quantization error is assumed to be a white noise with mean squared value $\frac{\Delta^2}{12}$. Though none of these 3 assumptions are correct, they serve as useful approximations to make the analysis simpler and the PSD so obtained is close to the actual PSD.

In Nyquist rate ADCs(where sampling rate f_s is only slightly greater than twice the bandwidth of the signal), the output signal occupies the entire digital spectrum(from 0 to π). Hence the entire quantization noise(Q_{Noise}) is within the band of the signal(in-band). The Signal To Quantization Noise Ratio(SQNR) of a L level ADC for a sine wave input(with frequency<f_s/2) is given by

$$SQNR = 6log_2(L) + 1.78 \text{ dB} \tag{2.2}$$

If $L = 2^N$ then the SQNR for such a N bit ADC is given by

$$SQNR = 6N + 1.78 \text{ dB} \tag{2.3}$$

So if the number of bits increases by 1 then the SQNR increases by 6 dB.

2.2 DSM Basics

2.2.1 Oversampling

Instead of quantizing the signal at Nyquist rate, a faster sampling rate can be used. The oversampled signal does not occupy the entire digital spectrum[3]. As the OSR(Over Sampling Ratio which is defined as $\frac{f_s}{2f_b}$) increases the spectral coverage of the signal decreases. The Q_{Noise}(assumed to be white) still covers the entire spectrum and hence its In-band content reduces as OSR increases as shown in Figure. 2.3.

Hence this out of band noise can be removed by using a digital low pass filter with a cut-off of $\frac{\pi}{OSR}$(assuming ideal filter with sharp cut off). The low pass filter does not affect the signal since the signal content is within $\frac{\pi}{OSR}$. The Q_{Noise} after filtering has only the in-band portion which is $\frac{\Delta^2}{12OSR}$. As OSR doubles the Q_{Noise} power reduces by 2 and hence the SQNR improves by 3 dB or essentially a 0.5 bit improvement. An N bit ADC with OSR of 4 after filtering looks like a N+1 bit ADC in terms of SQNR.

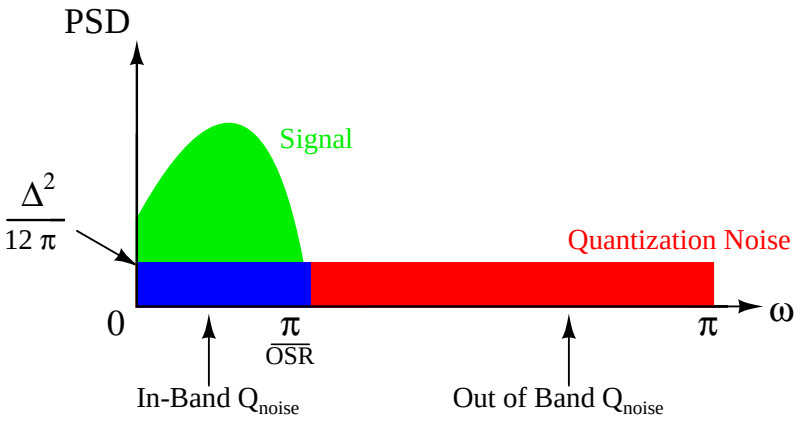


Figure 2.3: PSD of Oversampled ADC

2.2.2 Oversampling with noise shaping using negative feedback

By oversampling the In-Band Quantization Noise(IBQN) was reduced. In addition noise shaping can also be used to further reduce the IBQN. Noise shaping is a process by which Q_{Noise} is shaped in such a way that most of its contents lie outside the signal band. It basically makes use of negative feedback to shape the Q_{Noise}.

Consider the general negative feedback circuit shown in Figure. 2.4.

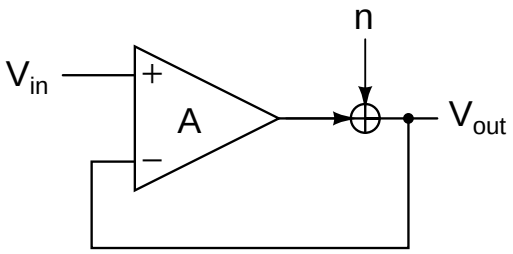


Figure 2.4: Negative Feedback Example

The output V_{out} is given by

$$V_{out} = \frac{A}{1+A} V_{in} + \frac{n}{1+A} \tag{2.4}$$

where A is the gain of the amplifier.

It is assumed that n is some additive noise source. This noise is getting attenuated by the gain of the amplifier due to negative feedback.

As A $\rightarrow \infty$, $V_{out} = V_{in}$ and there is no noise contribution at the output. This serves as motivation for Delta Sigma modulators in which the quantizer is embedded in a similar loop as shown in Figure. 2.5. The additive noise model along with unit delay is used to model the quantizer.

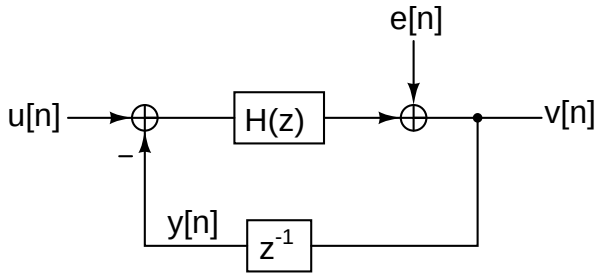


Figure 2.5: DSM Block Diagram

Since the delay z^{-1} is an integral part of the quantizer, only $y[n]$ can be accessed. $y[n]$ is just a delayed version of $v[n]$.

$$V(z) = \frac{H(z)}{1 + H(z)z^{-1}}U(z) + \frac{E(z)}{1 + H(z)z^{-1}} \tag{2.5}$$

$$V(z) = STF(z)U(z) + NTF(z)E(z) \tag{2.6}$$

H(z) is an integrator(low pass filter) with high gain in the signal band. Hence $STF \approx 1$ in the signal band and the quantization noise(E(z)) in the signal band is attenuated by the large gain of H(z).

2.2.3 First order DSM

For a first order DSM, $H(z) = \frac{1}{1-z^{-1}}$ which gives

$$STF(z) = 1 \tag{2.7}$$

$$NTF(z) = 1 - z^{-1} \tag{2.8}$$

The Q_{Noise} PSD is given by

$$PSD_Q(\mathrm{e}^{j\omega}) = \frac{\Delta^2}{12\pi} \tag{2.9}$$

The Q_{Noise} PSD at the output

$$PSD_Q(\mathrm{e}^{j\omega}) = \frac{\Delta^2}{12\pi}|NTF(\mathrm{e}^{j\omega})|^2 \tag{2.10}$$

Since at lower frequencies $|NTF(\mathrm{e}^{j\omega})| \approx \omega$, IBQN(assuming large OSR) is given by

$$IBQN = \int_0^{\frac{\pi}{OSR}} \frac{\Delta^2}{12\pi}|NTF(\mathrm{e}^{j\omega})|^2 d\omega \tag{2.11}$$

$$= \frac{\Delta^2\pi^2}{36\,OSR^3} \tag{2.12}$$

As seen from the IBQN expression, doubling the OSR reduces the IBQN by 8 times thereby improving SQNR by 9 dB(1.5 bits improvement). But the total noise variance has actually increased as seen from Equation. (2.14).

$$\text{Total } Q_{Noise} = \int_0^{\pi} \frac{\Delta^2}{12\pi}|NTF(\mathrm{e}^{j\omega})|^2 d\omega \tag{2.13}$$

$$= \frac{\Delta^2}{6} \tag{2.14}$$

Though the total noise has increased its in-band content has reduced thereby improving SQNR after digital low pass filtering.

2.2.4 Higher order DSM

Since with 1^{st} order DSM, an improvement of 1.5 bits for every doubling of OSR is obtained, one can expect to get better performance with a higher order loop filter. As order increases the loop gain at lower frequencies increases thereby reducing IBQN. It

can be shown for an N^{th} order filter that at low frequencies $|NTF(e^{j\omega})| \propto \omega^N$. Hence $IBQN \propto \frac{1}{OSR^{2N+1}}$. So there is an improvement of $N + 0.5$ bits for every doubling of OSR. But as order increases the total noise variance also increases since OBG increases. This also increases the swing of the quantizer which can cause quantizer to saturate thereby breaking the feedback and leading to instability. Therefore order cannot be increased beyond a limit.

2.3 CTDSM Overview

2.3.1 Need for CTDSM

The loop filter of the DSM can be implemented in either CT or DT domain. Switched capacitors are the building blocks of DTDSM. These switched capacitor loads are difficult to drive especially at high speeds. Moreover DTDSM requires an anti-aliasing filter upfront. These issues are not there with CTDSMs. CTDSM has implicit anti-aliasing property which attenuates signals around multiples of the sampling frequency. Moreover CTDSM has resistive input load which are easier to drive. Hence CTDSMs are widely used.

2.3.2 Loop Filter

Figure. 2.6 shows the block Diagram of CTDSM.

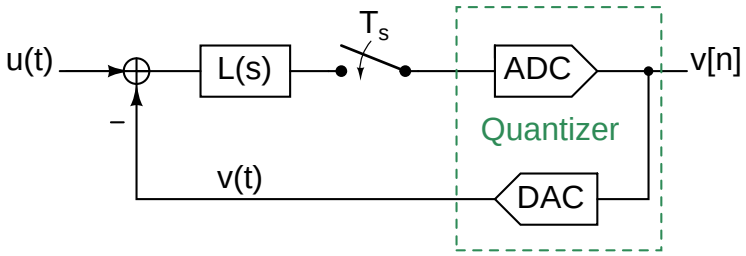


Figure 2.6: CTDSM Block Diagram

The quantizer output is converted to discrete-time by using a Digital to Analog Con-

verter(DAC) in the feedback path. The loop filter L(s) is chosen such that its sampled output resembles that of the Discrete time counterpart H(z). This relationship is shown in Figure. 2.7.

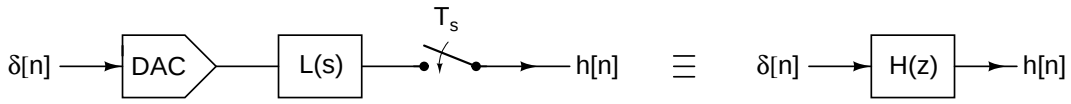


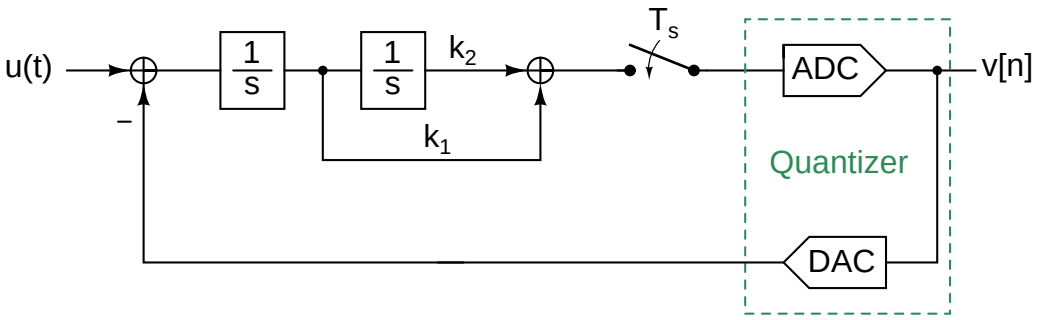
Figure 2.7: Relationship between CT and DT Loop Filter

The loop filter consists of a cascade of integrators. The integrators are implemented either using active RC filters or Gm-C filters. Active RC are generally preferred for their linearity. But they are prone to RC time constant variations due to process. Hence RC tuning is required to keep the time constants close to their nominal values.

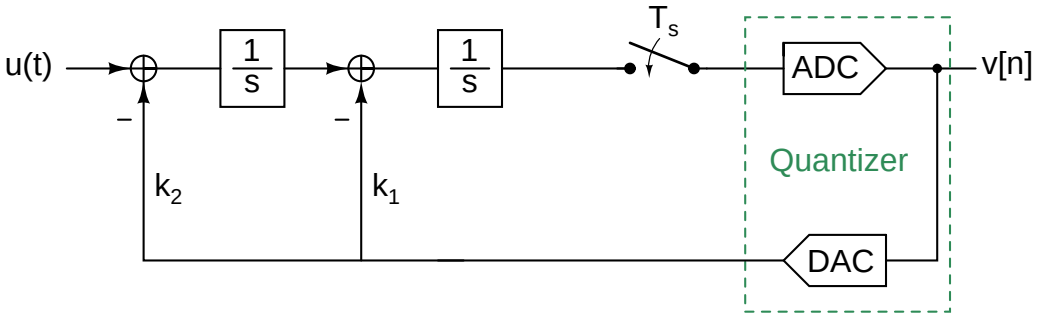
There are several topologies for the loop filter. The commonly used ones are Cascade of Integrators with Feed Forward(CIFF) and Cascade of Integrator with Feed Back(CIFB).They are shown in Figure. 2.8 for a second order CT loop filter. Hybrids of CIFF and CIFB are also used.

2.3.3 Quantizer

The quantizer within the CTDSM loop consists of the ADC along with the feedback DAC. The quantizer can be either single bit(2 Level) or multi-bit(multi-Level). Single bit quantizers have been extensively used in the past due to the fact that the quantizer is inherently linear(both ADC and DAC). They are also easy to design and consumes lesser area and power. But for high bandwidth applications they require high sampling rates due to the low Out of Band Gain(OBG) constraint of 1.5[4]. Moreover they are more sensitive to clock jitter which aggravates at higher speeds when the rise fall times are comparable to the time period of the clock. Hence multi-bit quantizers are nowadays considered for high bandwidth applications due to their relaxed jitter constraints and aggressive NTF(due to increased OBG tolerance). But in multi-bit quantizers the issue of non-linearity crops up. The ADC non-linearity can be tolerated to an extent since it gets noise shaped similar to quantization noise. But the DAC non-linearity, due to



a) CIFF



b) CIFB

Figure 2.8: Loop Filter Topologies

mismatch of DAC elements, is a serious issue since it is similar to corrupting the input itself. Fortunately the mismatch of DAC elements can be mitigated by using Dynamic Element Matching(DEM). There are several DEM schemes of which Data Weighted Averaging(DWA) is popular one due to its simple and efficient implementation. But at high speeds DEM techniques are difficult to implement because of the fact that their delays generally exceed a clock period making them ineffective.

2.4 CTDSM Architecture and Specifications

A 4th order loop filter with CIFFB(Cascade of Integrators with Feed Forward and Feedback) topology is used in the CTDSM. Input feed forward paths are used to keep the swings of the OP-AMPs within limits. Assistant DACs are used to improve the unity gain frequency of the last OP-AMP. The NTF is designed with optimized zeroes. The CTDSM is shown in Figure. 2.9.

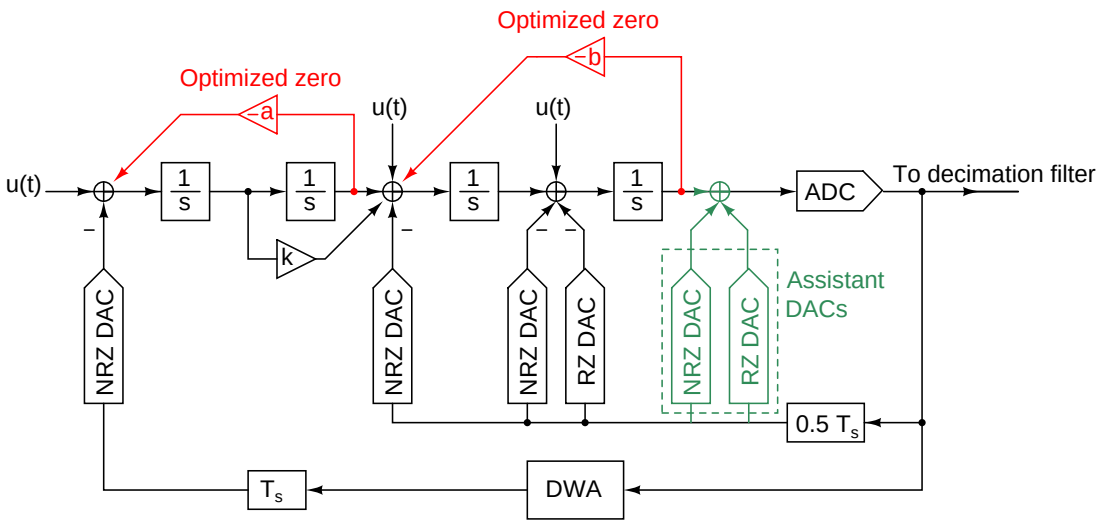


Figure 2.9: CTDSM Architecture

$T_s = \frac{1}{f_s}$ is the time period of the ADC sampling clock. The 0.5 T_s and T_s blocks denote that the outputs of these blocks are synchronized at 0.5 T_s and T_s(with respect to the ADC sampling clock) respectively.

The CTDSM is designed to operate at a sampling frequency(f_s) of 2.6 GHz over a

bandwidth of 100 MHz(ORS of 13). The loop filter is designed such that the shaped IBQN is lower than the thermal noise. The target SNR limited by thermal noise is 74 dB and the target SQNR is chosen to be higher than 83 dB. The design of loop filter and the DACs are not part of this thesis.

CHAPTER 3

Flash ADC Design

3.1 Flash ADC Components

Flash ADC is one of the fastest ADC available. It utilizes maximum parallelism at the expense of area and power dissipation. A typical flash ADC is shown in Figure. 3.1.

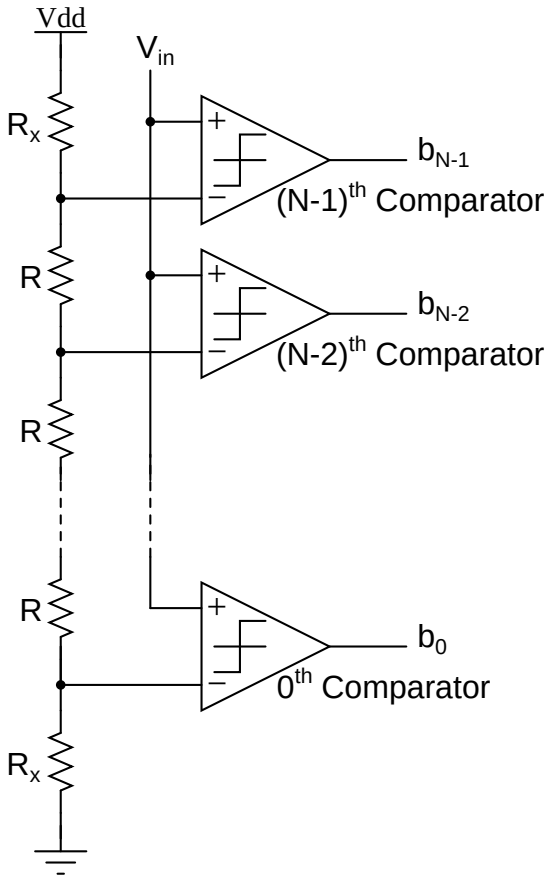


Figure 3.1: N level Flash ADC

A N level flash has N-1 comparators. There are N-1 output lines producing Thermometer code(a number of 1's followed by a number of 0's). This is shown in Figure. 3.1 for a 4 level Flash ADC(3 comparators) along with the corresponding decimal code.

Table 3.1: Output Thermometer codes for a 4 level Flash ADC

Comparator Outputs			Output Code (in Decimal)
MSB		LSB	
2^{nd}	1^{st}	0^{th}	
0	0	0	0
0	0	1	1
0	1	1	2
1	1	1	3

Each comparator has a different threshold voltage based on its position. Hence there will be N-1 threshold voltages equally spaced within the signal range. These threshold voltages are generated using a resistive ladder. In the differential picture there will be two resistive ladders.

Each comparator generally consists of a sample and hold circuit(sampler), subtraction circuit and regenerative latch as shown in Figure. 3.2. Alternatively there can be just one sampler common for all the comparators. The subtraction circuit subtracts the reference from the input. The output of the subtraction circuit is fed to the regenerative(regen) latch. If input of the regenerative latch is positive(>0) then the output regenerates to logic 1 else it regenerates to logic 0.

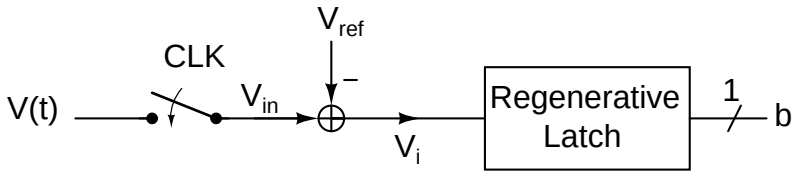


Figure 3.2: Block Diagram of a Comparator

Since both the sampling operation and subtraction operation are linear, they can be interchanged as shown in Figure. 3.3.

Instead of performing explicit sampling, it can be made part of the regenerative latch. The clocked regenerative latch transformation is shown in Figure. 3.4.

At high speeds it is not advisable to have switches in the analog signal path since it leads to non-linearities and additional delay. Hence clocked regenerative latch is an

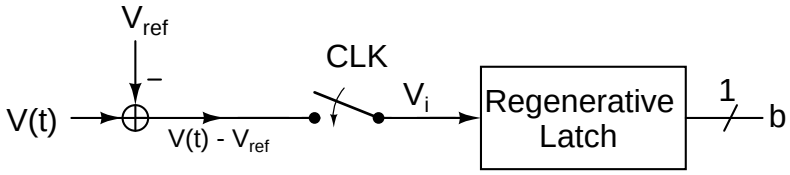


Figure 3.3: Alternate Block Diagram of Comparator

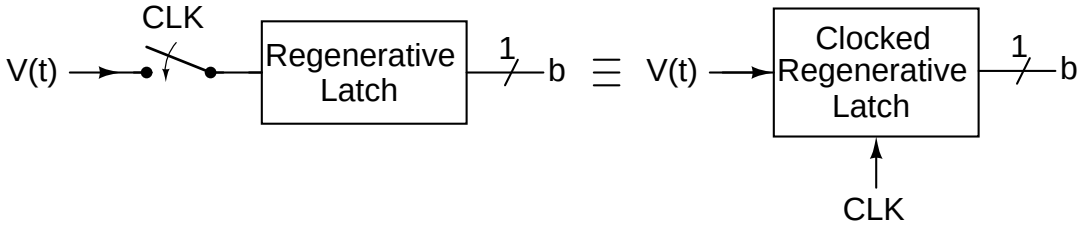


Figure 3.4: Clocked Regenerative Latch Transformation

advantage.

The block diagram of unit comparator with the clocked regenerative latch is shown in Figure. 3.5.

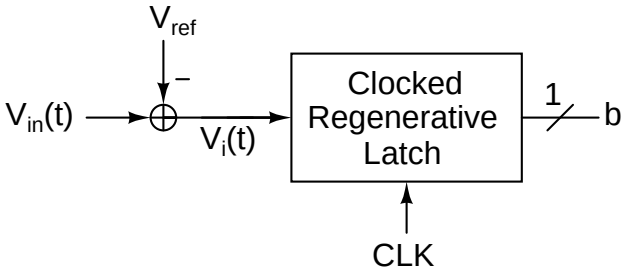


Figure 3.5: Unit Comparator Block with Clocked Regenerative Latch

The differential unit comparator block is shown in Figure. 3.6.

The differential references V_{refp} and V_{refm} are centred around the same common mode as the inputs V_{inp} and V_{inm}. This ensures that the common mode gets removed on subtraction. This enables the clocked regenerative latch to have a different common mode.

At every rising edge of clock(CLK) the clocked regenerative latch regenerates as given in Table. 3.2.

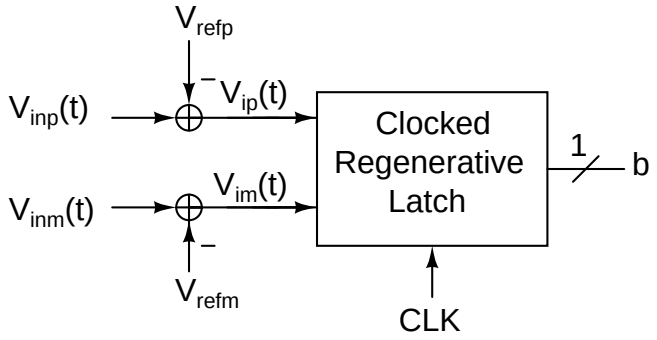


Figure 3.6: Differential Unit Comparator Block

Table 3.2: Clocked Regenerative Latch Operation on Rising edge of Clock

Input	Output
$V_{ip} > V_{im}$	$b = 1$, $\overline{b} = 0$
$V_{ip} < V_{im}$	$b = 0$, $\overline{b} = 1$

3.2 Subtraction Circuit

The subtraction circuit is one of the most critical component of high speed flash ADCs. It is the main culprit that causes the references of comparators to change from their nominal value leading to Differential and Integral Non-Linearity(DNL and INL). Subtraction is generally performed by passing the input signal through a capacitor charged to the reference voltage as shown in Figure. 3.7.

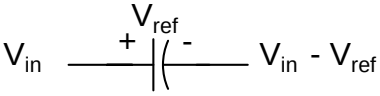


Figure 3.7: Subtraction using charged capacitor

The charged capacitor acts as a voltage source enabling subtraction. The capacitor needs to be recharged regularly due to the fact that the charge leaks with time. During this recharge phase the input must be disconnected from the capacitor. Since both the plates of the capacitor are available for charging, it can be ensured that the subtracted value rides over the common mode of the clocked regenerative latch as shown differentially in Figure. 3.8. Parasitic capacitance at the input nodes of the regenerative latch

leads to hysteresis. Therefore the bottom plates of the capacitors are chosen to face away from the regenerative latch inputs.

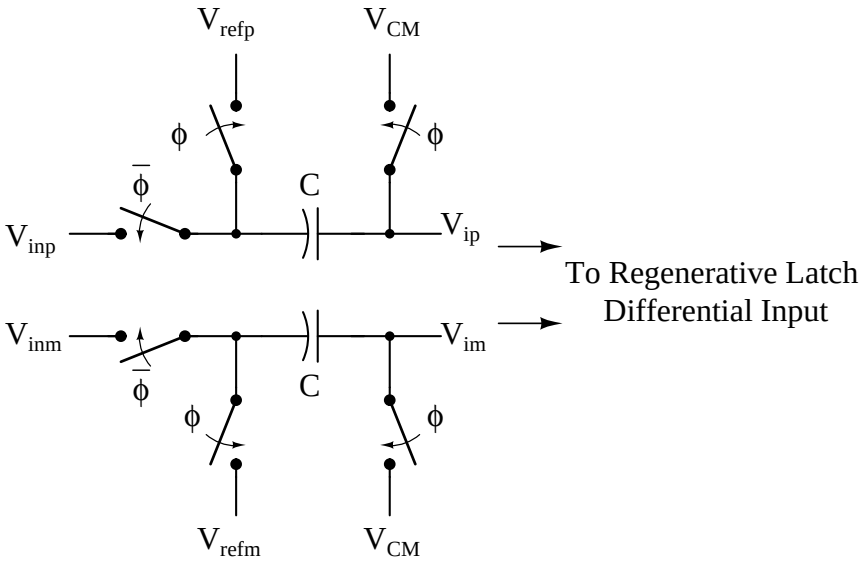


Figure 3.8: Differential Subtraction Circuit

$V_{inp}, V_{inm}, V_{refp}$ and V_{refm} are given by

$$V_{inp} = V_{CMin} + \frac{V_{in}}{2} \tag{3.1}$$

$$V_{inp} = V_{CMin} - \frac{V_{in}}{2} \tag{3.2}$$

$$V_{refp} = V_{CMin} + \frac{V_{ref}}{2} \tag{3.3}$$

$$V_{refm} = V_{CMin} - \frac{V_{ref}}{2} \tag{3.4}$$

The output of subtraction circuit V_{ip} and V_{im} are given by

$$V_{ip} = V_{inp} - V_{refp} + V_{CM} = V_{CM} + \frac{V_{in} - V_{ref}}{2} \tag{3.5}$$

$$V_{im} = V_{inm} - V_{refm} + V_{CM} = V_{CM} - \frac{V_{in} - V_{ref}}{2} \tag{3.6}$$

Since switches are used to disconnect the inputs during recharging phase (Φ), it leads to non-linearity. Moreover the ADC is not operational during this phase. These switches in the signal path can be avoided if the subtraction capacitors are not recharged directly.

This can be achieved by first charging another pair of capacitors and using these capacitors to charge the subtraction capacitors. This modification is shown in Figure. 3.9.

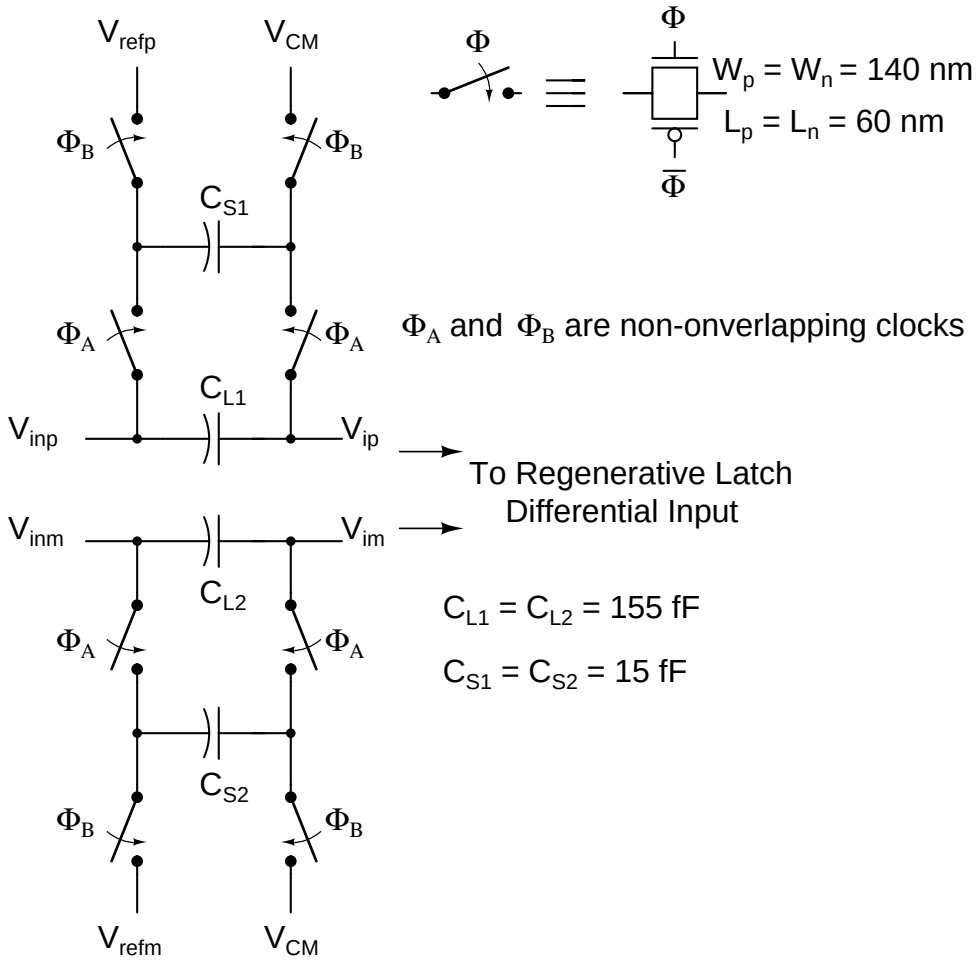


Figure 3.9: Subtraction Circuit without input switches

Now since the subtraction capacitors are not charged directly, there is no need for switches in the signal path. In order to prevent direct connection between subtraction capacitors and the reference and common mode voltages, non-overlapping clocks are used. Since the subtraction capacitors do not lose charge quickly these non-overlapping clocks can be much slower than than the ADC sampling frequency f_s. A frequency of $\frac{f_s}{16}$ is chosen for these non-overlapping clocks. The lower frequency of operation also helps in keeping the switch sizes small.

The capacitance value of the subtraction capacitor is chosen such that it is at-least 10 times larger than the capacitance of the input of the regenerative latch. This ensures

that the charge lost by the subtraction capacitor due to voltage division is minimized. The size of the capacitor used to charge the subtraction capacitor is chosen to be about 10 times smaller than the subtraction capacitor since it only needs to provide a small amount of charge to the subtraction capacitor to restore the lost charge. But on start-up it takes about a 800 clock cycles(about 300 ps for f_s = 2.6 GHz) for the subtraction capacitors to get fully charged.

3.3 Clocked Regenerative Latch

The regeneration operation utilizes the positive feedback of back to back connected inverters to regenerate a small voltage to full rail logic. The regeneration time depends on the strength of differential input. If differential input is large then the regeneration is faster. But if the differential input is small then it takes a long time to regenerate. At high speeds a small differential input may not regenerate to valid logic level due to time constraint. This issue is called metastability.

Pre-amplifiers are used to amplify the differential signal first before regenerating in order to reduce the offset of the latch[5, 6]. But at high speeds, pre amplifier adds to the delay and becomes difficult to accommodate.

Though there are several ways to achieve regeneration, the strong arm latch[7] shown in Figure. 3.10 is an exiting prospect due to the following advantages

- Has inbuilt pre-amplification due to the input differential pair that mitigates metastability.
- It is a clocked regenerative latch that eliminates switches to sample the input.
- Being fully differential it provides common mode rejection
- Uses only single phase of clock which is advantageous since it is difficult to generate and propagate differential clock through the long ADC chain. Moreover single phase clock saves routing and power consumption.

Before the start of every clock cycle the outputs of the regenerative latch needs to be reset to some value so that they are not influenced by the previous decision. Here they

are reset to logic 1(vdd) using the transistors M7 and M8. For this purpose also the same clock is used thereby eliminating the need for a reset clock.

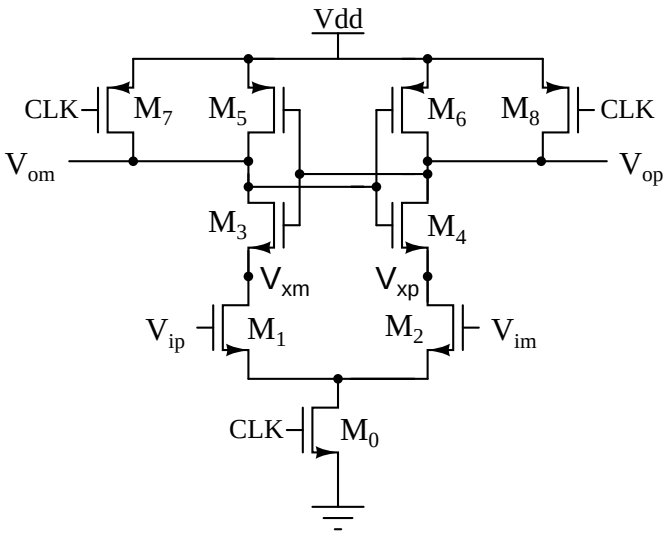


Figure 3.10: Clocked Regenerative Latch

The sizes of the transistors used are shown below. All transistors are of minimum length(60 nm) for high speed operation.

Transistor	Width(μm)
M_0	6
M_1, M_2	4
M_3 to M_8	1

The operation of the latch is explained using the timing diagram shown in Figure. 3.11.

From the timing diagram we can see that the regeneration time depends on the strength of the differential input.

The zoomed version of the regeneration phase(regen phase) is shown in Figure. 3.12.

During the beginning of the regeneration phase voltages of both output nodes V_{op} and V_{om} decrease as current flows through both branches of the latch till the current reaches maximum. Then the actual regeneration occurs. Depending on the differential input the current in one branch decreases while the current in the other branch increases thereby leading to regeneration.

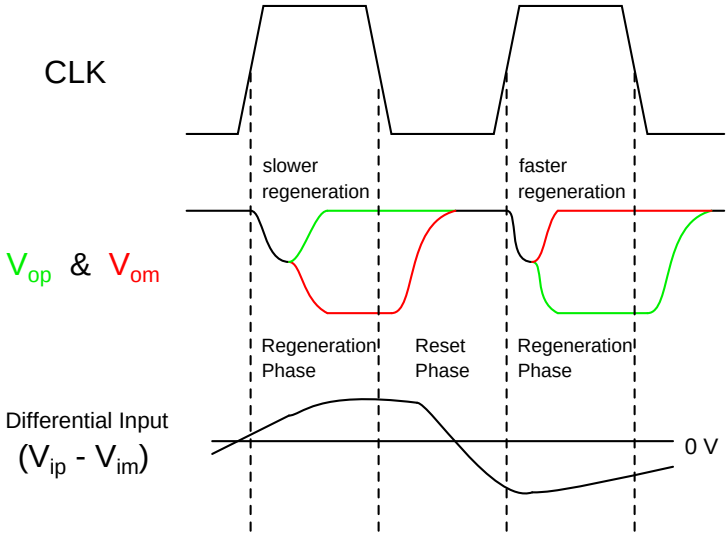


Figure 3.11: Clocked Regenerative Latch Timing Diagram

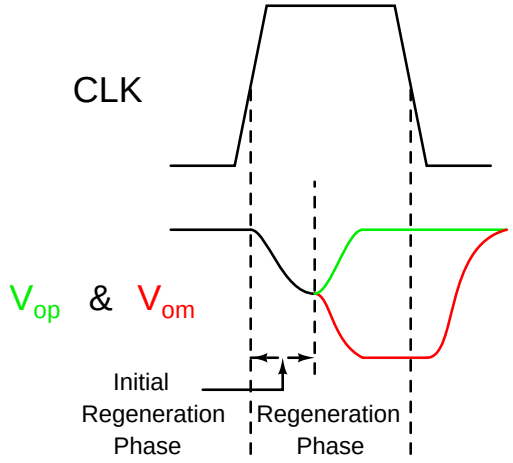


Figure 3.12: Regeneration Phase Timing Diagram

In the beginning of the regeneration phase, when voltages of both the nodes V_{op} and V_{om} decrease, there will be a slight difference in voltage between these two nodes depending on the differential input. Hence it is during this initial part of the regeneration phase that the input is sampled. There is no fixed sampling instant. Input changes after this initial window does not affect the decision of the latch.

The sizes of the transistors are chosen such that the regeneration completes within 0.25 T_s(< 96 ps) for a differential input of 5 mV across all corners and temperatures. SS corner at 70° C is the worst case scenario for delay.

The reset transistors M7 and M8 could have been made smaller. But if they are made smaller then at the end of the reset phase there is about 5 mV difference between V_{op} and V_{om}(depending on the previous decision) which affects the decision in the next cycle especially for small differential inputs. This dependance on previous cycle decision leads to hysteresis. Hence the reset transistors are not made smaller.

At the end of the regeneration phase the nodes V_{xp} and V_{xm} are 0 V(ground nodes of the back to back inverters). In the reset phase, when V_{op} and V_{om} are reset to logic 1(Vdd), the transistors M3 and M4 turn on and try to pull nodes V_{xp} and V_{xm} towards Vdd. But they will not reach Vdd due to the short duration of reset phase. There will be a small difference in the voltages between V_{xp} and V_{xm}. This also leads to hysteresis. This can be mitigated by using either of the schemes shown in Figure. 3.13.

The scheme shown in Figure. 3.13 b) is more effective since it tries to keep both nodes same during reset phase. But it is found that this hysteresis effect does not affect the performance of the latch. Hence neither of these schemes are used.

3.4 SR Latch

The clocked regenerative latch outputs reset to Vdd in the negative phase of the clock. So there is a need to store the decision before this reset occurs. Flip-flops can be used to store these outputs. But we need to generate clocks for these flip-flops and also make sure that these clock are timed such that the decision is stored before reset. This

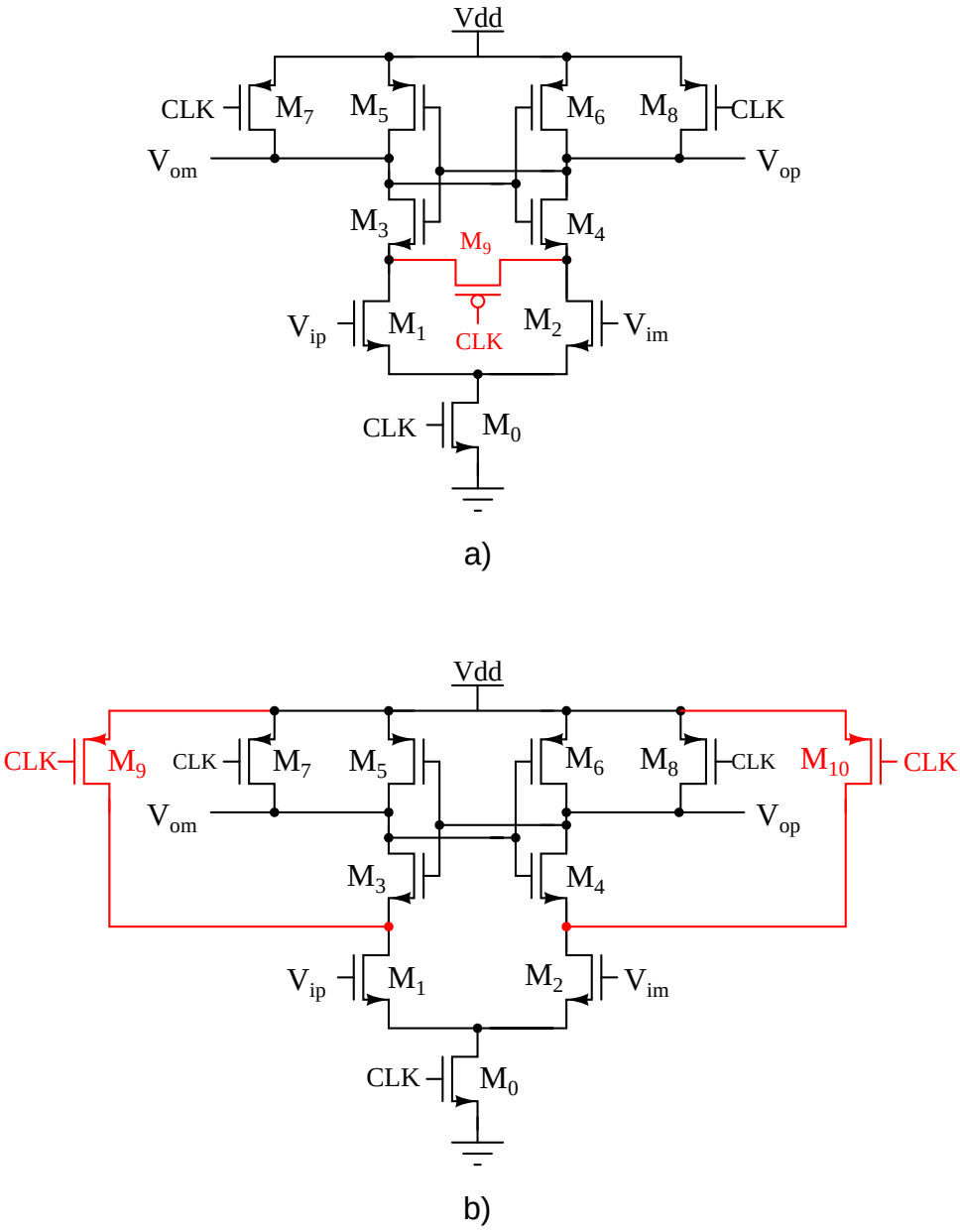


Figure 3.13: Clocked Regenerative latch Modifications

becomes difficult and adds to power consumption. A SR latch that does not respond to both inputs being logic 1(Vdd) can be used. NAND gate based SR latch shown in Figure. 3.14 does just what is required.

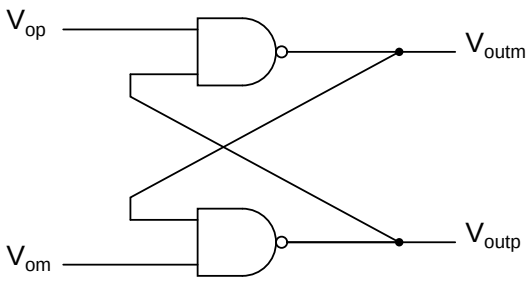


Figure 3.14: NAND gates based SR latch

The truth table of the latch is shown in Table. 3.3.

Table 3.3: SR Latch Truth Table

Inputs		Outputs	
V_{op}	V_{om}	V_{outp}	V_{outm}
0	1	0	1
1	0	1	0
1	1	Previous Value	Previous Value
0	0	1	1

When both inputs are logic 0 then the outputs are invalid(both logic 1). But the regenerative latch never produces such a combination.

The transistor level schematic of the SR latch is shown in Figure. 3.15.

If we directly cascade the regenerative latch with the SR latch then the SR latch provides an unequal load to the regenerative latch. As shown in figure. 3.16, at the start of the regeneration phase when SR latch outputs are $V_{outp} = logic\ 1$ and $V_{outm} = logic\ 0$, the gate of M_6 contributes less capacitance while the gate of M_7 provides significant capacitance.

This unequal capacitance affects the decision of the regenerative latch when the differential input is small. This is also a form of hysteresis since the decision stored in SR latch affects the next decision made by the regenerative latch. To avoid this issue a

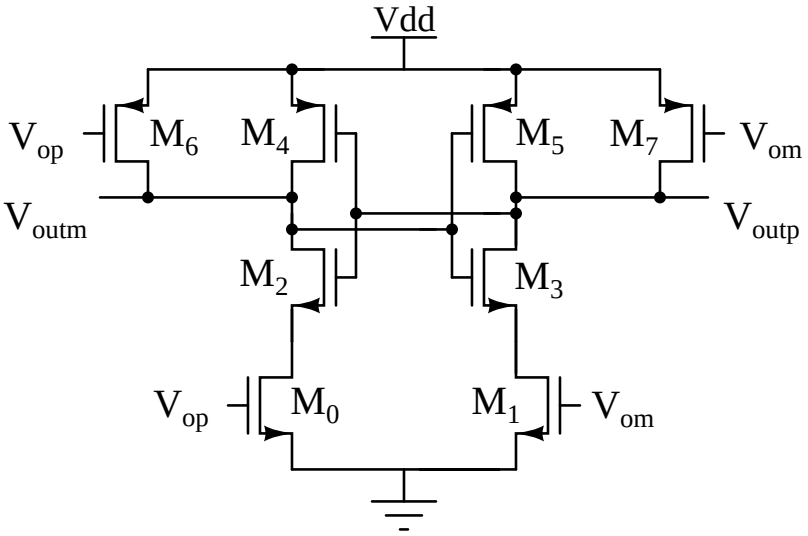


Figure 3.15: SR latch - Transistor Level

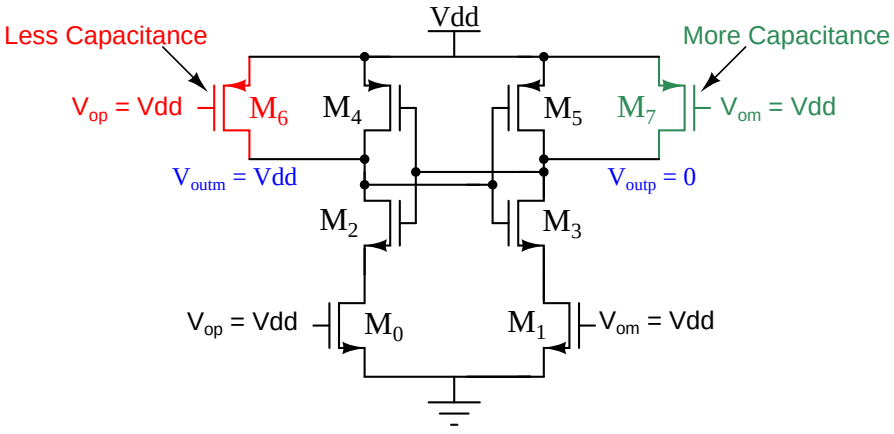


Figure 3.16: SR latch unequal input capacitance

pair of buffers can be used between regenerative latch and SR latch as shown in Figure. 3.17.

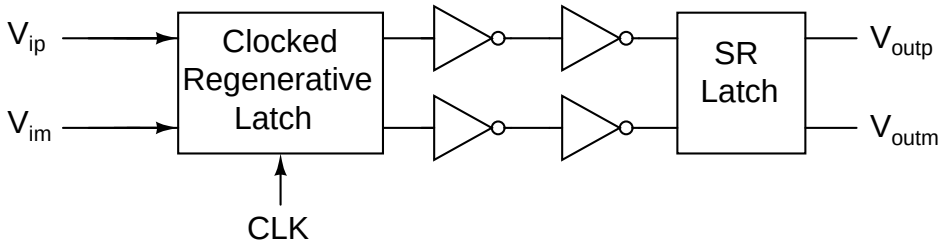


Figure 3.17: Buffers between Regenerative latch and SR latch

The sizes of the transistors in the SR latch are shown below. All transistors are of minimum length(60 nm) for high speed operation.

Transistor	Width(μm)
M_0, M_1	1
M_2, M_3	2.7
M_4, M_5	0.4
M_6, M_7	2.25

The sizes of the transistors in SR latch are chosen such that the entire delay from the clock edge to the SR latch outputs is less than 0.4 $\mathrm{T_s}$ (150 ps) for differential input of 10 mV.

3.5 Reference Generation

The input of ADC swings differentially from -1 V to +1 V(peak to peak of 2 V). For a 17 level mid-tread ADC, the step size is

$$\Delta = \tfrac{2}{16} = 0.125 \text{ V}$$

The 16 reference(threshold) voltages(single ended picture) are separated by the step size and range from -7.5Δ to 7.5Δ. The common mode voltage of the differential references is chosen to be same as the common mode voltage(0.8 V) of the ADC differential

inputs. The differential references ride over this common mode. The two reference strings are shown in Figure. 3.18.

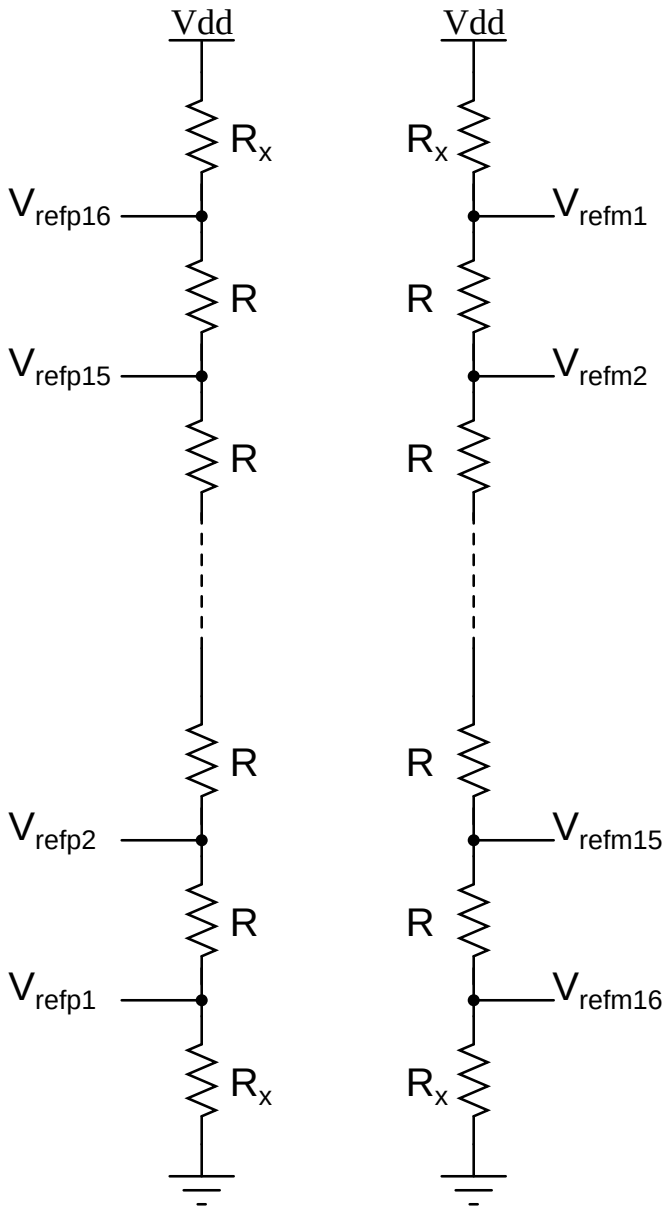


Figure 3.18: Differential Reference Generation - Resistive strings

Since these are DC nodes, a capacitor(1.57 pF) is placed from each reference node to ground in order to prevent dip in the DC voltage when current is drawn. In the resistive string R_x and R are related by $R_x = 5.3R$. R is chosen to be $634\,\Omega$ in order to charge the capacitor of the subtraction circuit with in the time allotted for charging(8 $\mathrm{T_s}$ = 3.07 ns). Each resistive string draws a static current of $98.5\,\mu\mathrm{A}$ and has static power

dissipation of $157\,\mu\mathrm{W}$.

3.6 Common mode voltage generation

The common mode voltage of the regenerative latch is chosen to be 0.6 V which is the midpoint of the ADC supply voltage(1.2 V). The common mode voltage is generated by using two resistors($1\,\mathrm{K}\Omega$ each) between digital supply and ground as shown in Figure. 3.19. A capacitor(3.76 pF) is placed between the common mode node and ground in order to prevent dip in the DC voltage when current is drawn.

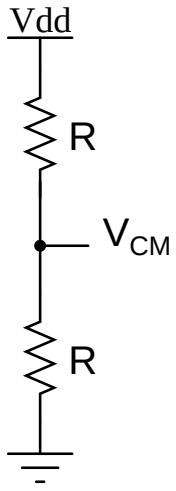


Figure 3.19: $\mathrm{V_{CM}}$ Generation

Since the ADC is about $240\,\mu\mathrm{m}$ long two such blocks are used one at each end of the ADC. Each block draws a static current of $600\,\mu\mathrm{A}$ and has static power dissipation of $720\,\mu\mathrm{W}$.

3.7 Layout

The layout of a single comparator is shown in Figure.3.20. The layout is obtained from the layouts of two half circuits to ensure that the layout is fully differential.

The dimensions of a comparator are $52\,\mu\mathrm{m}$ x $15\,\mu\mathrm{m}$.The length of the comparator is

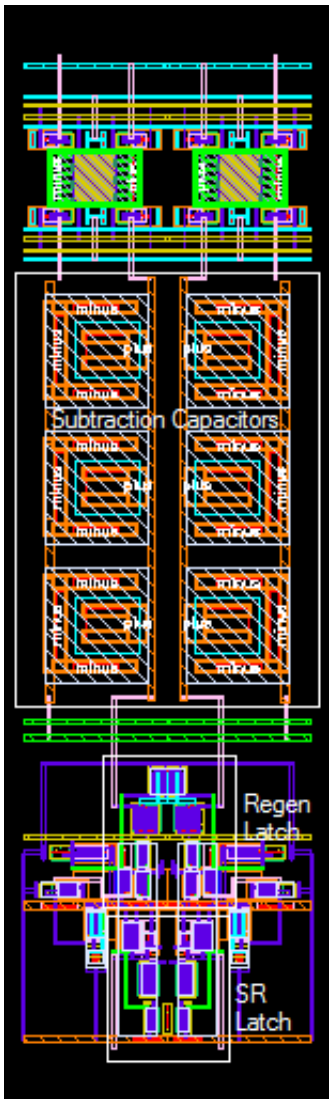


Figure 3.20: Layout of a Comparator

made as small as possible in order to reduce the length of the ADC. Linear capacitors(used in subtraction circuit) in the ST 65 nm library have a minimum dimensional limit. This determines the comparator length.

The layout of the 16 comparators put together are shown in Figure. 3.21.

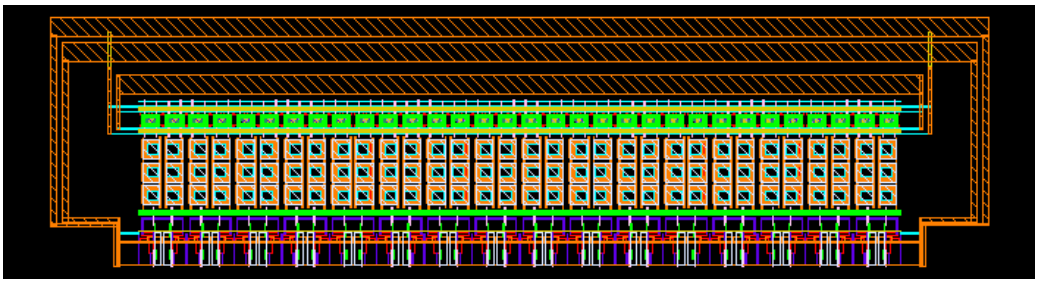


Figure 3.21: Layout of 16 Comparators

The layout of the entire ADC(16 comparators along with reference and V_{CM} generation) is shown in Figure. 3.22.

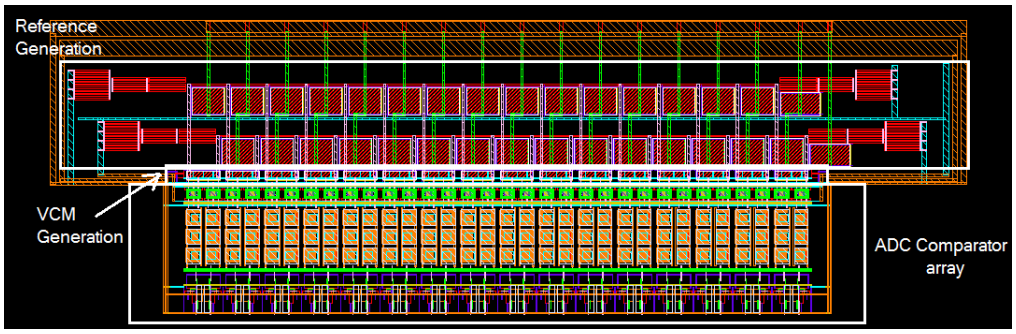


Figure 3.22: Layout of ADC

The dimensions of the ADC are $100\,\mu\mathrm{m}$ x $350\,\mu\mathrm{m}$ and the ADC occupies an area of $0.035\,\mathrm{mm}^2$.

3.8 Sumulation Results

3.8.1 Comparator

The comparator threshold voltage varies due to mismatch in the subtraction circuit and the regenerative latch. The offset in the threshold voltage(extend of threshold voltage variation) is determined by simulating it for 100 runs of Monte Carlo simulation. The threshold is determined by giving a slowly varying ramp to the comparator and detecting transition in the comparator output. The results are shown in Figure. 3.23 for a rising ramp and in Figure. 3.24 for a falling ramp. The accuracy of the simulation is $\pm 400\,\mu$V.

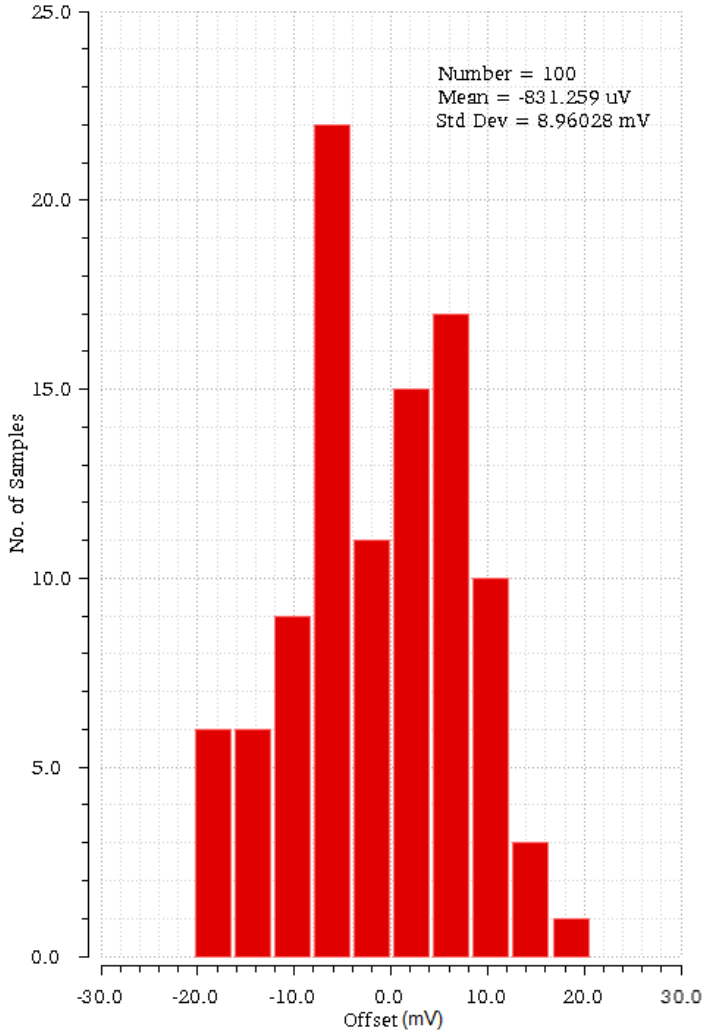


Figure 3.23: Offset in threshold for rising Ramp input

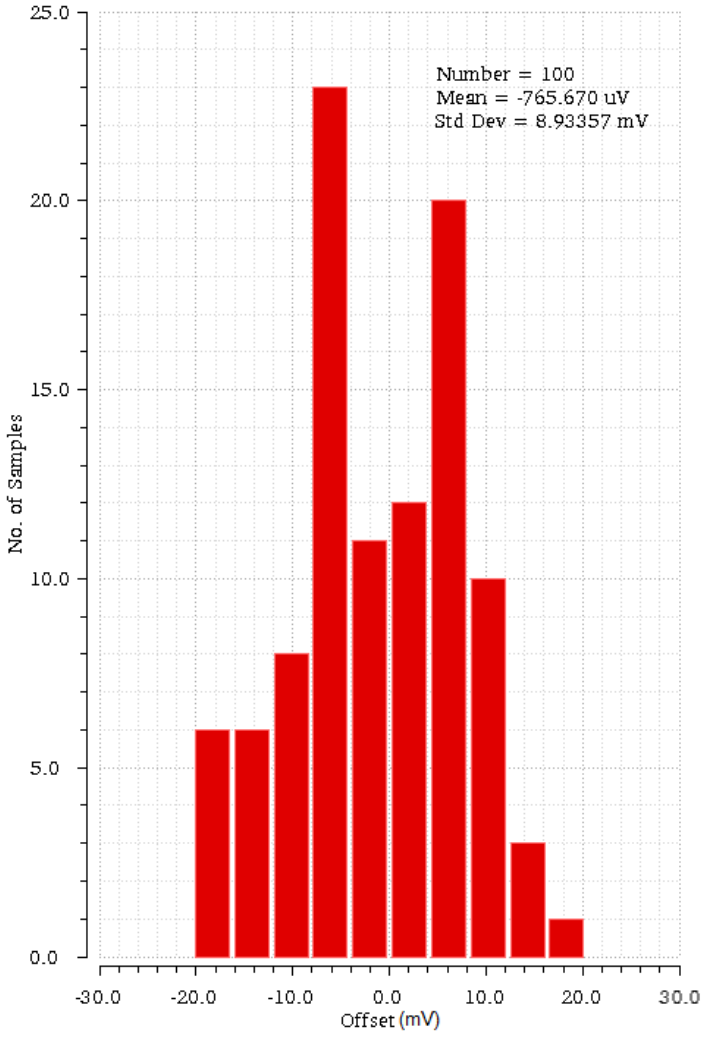


Figure 3.24: Offset in threshold for falling Ramp input

The standard deviation(sigma) of the offset is about 9 mV which is lesser than 0.1 LSB(12.5 mV).

3.8.2 ADC

INL

A figure of merit of ADC is INL. It is determined from the step characteristic of the ADC by giving a rising ramp input and plotting the output of an ideal DAC whose inputs are the ADC. The ADC characteristic is plotted in Figure. 3.25

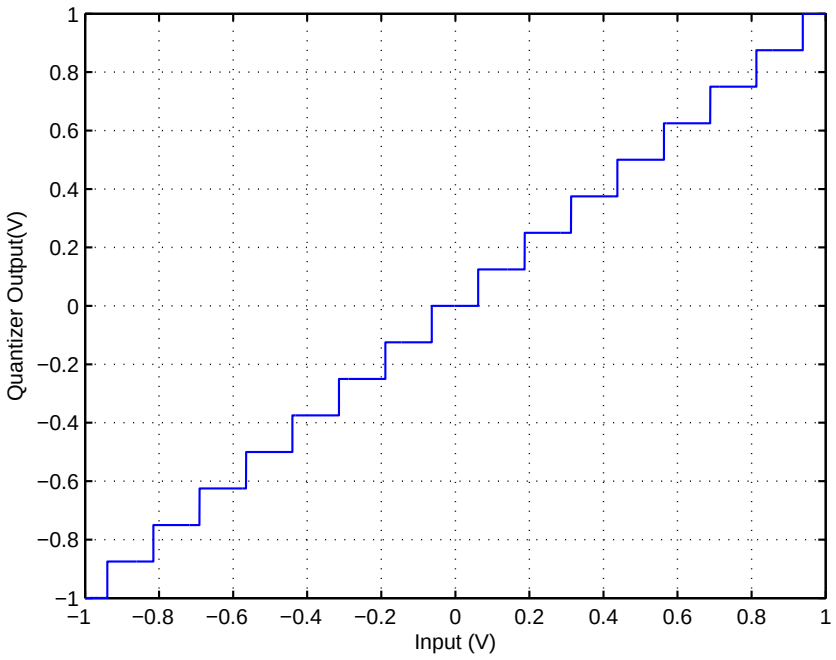


Figure 3.25: ADC Characteristics

The INL values are tabulated in Table. 3.4.

From the table it is seen that the magnitude of INL is well below 0.1 LSB. The INL degrades after R+C+CC extraction. This is due to the fact that there is a delay in the clock from the 1st to the 16th comparator. Even after degradation the INL is less than 0.2 LSB which along with the sigma offset determined in Subsection. 3.8.1 is less

Table 3.4: INL

Comparator	INL(in terms of LSB)
1	-0.017
2	-0.023
3	-0.023
4	-0.02
5	-0.02
6	-0.0138
7	-0.010
8	-0.008
9	-0.005
10	-0.001
11	0.001
12	0.001
13	0.007
14	0.007
15	0.005
16	0.007

than 0.3 LSB. The CTDSM loop can tolerate threshold variations upto 0.45 LSB while maintaining SQNR above 80 dB.

ADC Open loop SQNR

The ADC in open loop is simulated with an input frequency of 1.292 GHz(close to $f_s/2$). The SQNR is tabulated in Table. 3.5 for the worst case scenario(SS corner at 70° C). The SQNR is calculated after passing the output of ADC through an ideal DAC.

Table 3.5: SQNR of open loop ADC

Type of Netlist	SQNR(dB)
Schematic	26
C+CC	25.2
R+C+CC	25

The output spectrum for schematic in SS corner at 70° C is shown in Figure. 3.26.

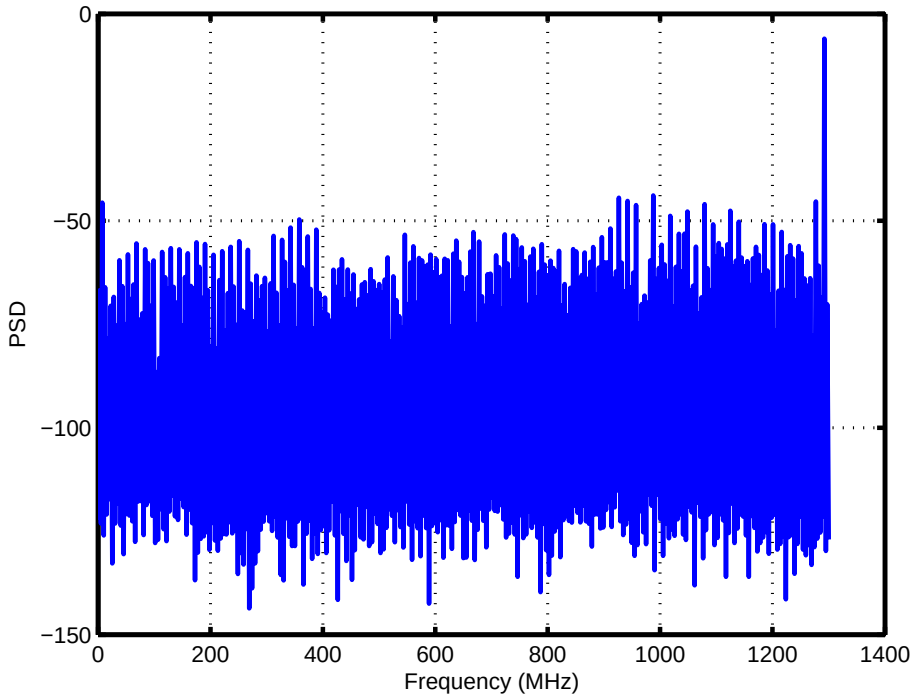


Figure 3.26: Spectrum of ADC output

SQNR of CTDSM loop

The ADC is embedded in the CTDSM loop and simulated with an input frequency of 28 MHz. The SQNR is tabulated in Table. 3.6. The SQNR is calculated after passing the output of ADC through an ideal DAC.

Table 3.6: SQNR of CTDSM with ADC

Type of Netlist	SQNR(dB)
Schematic	87
C+CC	85
R+C+CC	83

The CTDSM output spectrum for schematic in SS corner at 70° C is shown in Figure. 3.27.

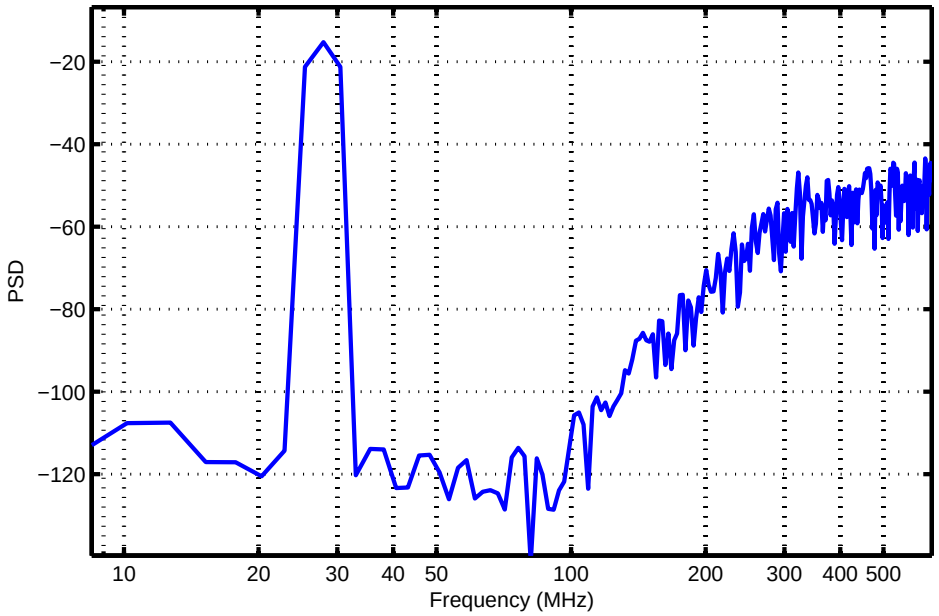


Figure 3.27: Spectrum of CTDSM output

Power Dissipation

The ADC has an open loop power consumption of 9.3 mW for input frequencies around $f_s/2$. The power consumption of the ADC inside the CTDSM loop is 5.4 mW.

CHAPTER 4

New High Speed DWA Implementation

4.1 Mismatch of DAC elements

One of the major concerns in multi-bit CTDSM is the issue of mismatch of DAC elements[8]. The CTDSM is especially sensitive to the mismatch present in the slow path DAC(highest order path). This DAC output has same transfer function to the CTDSM output as the CTDSM input. The mismatch components at lower frequencies directly affects the SQNR by increasing the noise floor and causing harmonics.

DAC mismatch causes the levels of the DAC to be different from their nominal values as shown in Figure. 4.1 for a 4 level DAC.

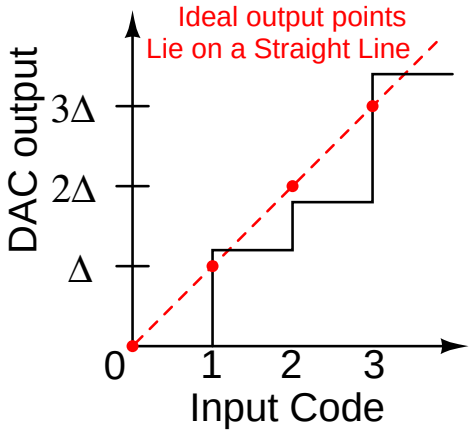


Figure 4.1: Mismatch of DAC elements

The difference in levels can be thought as non-linear distortion from the straight line. So this non-linearity leads to harmonics and rise of noise floor. When the input code is fixed the DAC elements that are active also gets fixed. Whenever the input code is 1, the 1st element is enabled(and it is always the 1st element that gets enabled). When the input code is 3, the first three elements are enabled. If the input is usually small then

the last few DAC elements rarely get enabled. Hence there is a clear input dependence to this non-linearity.

4.2 Ways to tackle mismatch of DAC elements

Mismatch can be reduced by increasing the sizes of the DAC elements. But this increases area and it is not possible to increase area beyond a limit. Hence other ways to deal with mismatch needs to be considered.

The first method that was tried in literature was randomization[9, 10]. The DAC element selection is random. Even for a fixed input code, the DAC elements that get enabled are different for different clock cycles. Since the selection is random it does not depend on the input code. So the harmonics disappear. The mismatch effect is modified to a white noise effect. Hence the noise floor increases thereby decreasing the SQNR.

Another way to address the issue is to think on lines similar to the DSM noise shaping idea. The IBQN was reduced by shaping the Q_{Noise} out of the signal band. A method can be devised to similarly shape the effect of the mismatch of DAC elements out of the signal band[11, 12]. This is in principle similar to having a CTDSM loop in the DAC path to shape mismatch. Generally a 1st order mismatch shaping is sufficient to reduce the mismatch below the IBQN level. There are several methods to achieve 1st order mismatch shaping. The most efficient and easy to implement method is Data Weighted Averaging(DWA).

4.3 DWA concept

DWA method selects the DAC elements in a circular fashion. There is a pointer which points to a DAC element. This pointer increments based on the input code. The elements covered by the pointer are enabled. The new position of the pointer is from where the increment happens in the next cycle. If the pointer reaches the last DAC element it circles back to the first element. This technique is shown in Figure. 4.2 for a 5 level

DAC with 4 DAC elements(numbered from 0 to 3).

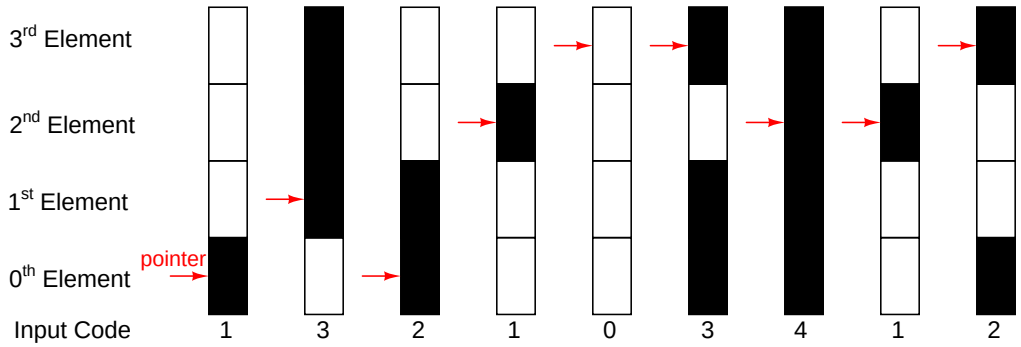


Figure 4.2: DWA example using 5 level DAC

From this it is observed that the input code only determines the number of DAC elements to be enabled. It is the pointer that decides which elements get enabled. And it is seen that a set of adjacent elements(in a circular fashion) get selected. In other words the thermometer code is shifted according to the pointer value and then applied to the DAC elements.

4.4 Conventional DWA Implementation

The conventional DWA implementation is shown in Figure. 4.3 for a L+1 level ADC and DAC where $L = 2^N$.

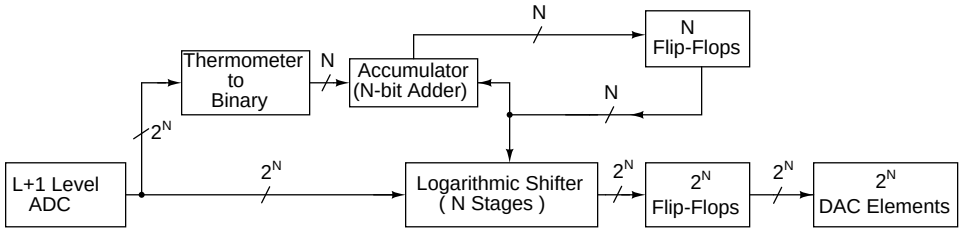


Figure 4.3: Conventional DWA Implementation

Flip-flops are used to store the pointer value that is used to control a shifter that is placed between the ADC and DAC. The output of ADC is converted to binary and then added with the pointer value to obtain the new pointer value that is used to control the shifter in the next cycle.

Since the DWA operates every clock cycle there is only a clock cycle available to generate the new pointer value. The thermometer to binary can be achieved in two ways. One method is to add individual bits. Another method is to convert thermometer code to one-hot code using a transition detect and then convert this one-hot code to binary. Both these involve significant delay. This binary code needs to be added to the current pointer value to obtain the next pointer value. This requires an accumulator(N-bit adder). For a 17 level ADC a 4-bit adder is required. The 4-bit adder itself has delay of close to 0.5 ns in 65 nm technology. Hence this conventional DWA method is unusable for high speeds(> 1 GHz).

4.5 New DWA Implementation

In conventional DWA, the ADC output was added to the current pointer to determine the new pointer. But an interesting fact is that the current DWA output itself has information about the next pointer. The position of the 10 transition(going from LSB to MSB) in the DWA output(taken circularly) directly denotes the next pointer value. An example is shown in Equation. 4.1 for a 9 level ADC.

$$\begin{aligned}
&\text{ADC Output(thermometer Code)} = 00000011(\text{Decimal } 2)\\
&\text{Current Pointer decimal value} = 1\\
&\text{Next Pointer decimal value} = 1 + 2 = \mathbf{3}\\
&\text{DWA Output} = 00000110\\
&\text{Position of 10 transition(going from LSB to MSB) in DWA otuput} = \mathbf{3}
\end{aligned} \tag{4.1}$$

This new implementation is shown in Figure. 4.4 for a L+1 level ADC and DAC where $L = 2^N$.

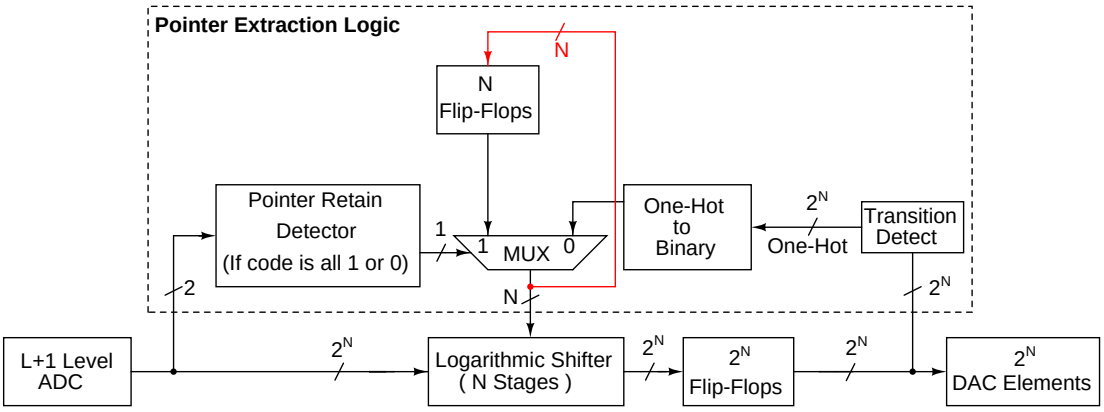


Figure 4.4: New High Speed DWA Implementation

4.5.1 Advantages

- Works at high speeds.
- Reduces the complexity and hence reduces the area and power dissipation of the DWA block, since there is no accumulator block.

4.5.2 Disadvantage

The time available to generate the next pointer value is reduced due to the fact that the DWA output is used instead of the ADC output.

4.6 Implemetation of individual blocks of the high speed DWA

4.6.1 Shifter

For a 17 level ADC, the shifter has 16 inputs, 16 outputs and a 4-bit control input. Based on the control input the shifter should be able to circularly shift the inputs. Hence each input needs to have a path to each of the 16 outputs through switches. This can be achieved by brute force by using a single stage of 256(16x16) switches. This is known as Barrel shifter. Though this method uses only single stage of switches it is not

advisable since each input and each output node is loaded with 16 switches. Moreover the routing is difficult and it requires 16 one-hot coded control lines(one for each shift). So the control input needs to be in one-hot code format. If more than one control line is enabled it leads to catastrophic errors in the output due to the fact that a single input is connected to multiple outputs and vice versa. Due to all these issues the barrel shifter is rarely used for large number of inputs. The commonly used shifter is logarithmic shifter. It consists of 4 stages of shifter each stage controlled by one bit of the control(pointer) input. The first stage can shift by $2^0 = 1$. The second stage can shift by $2^1 = 2$. The third stage can shift by $2^2 = 4$ and the final stage can shift by $2^3 = 8$. Since each stage can shift by a power of 2 it is called logarithmic shifter. Each stage uses 32(2x16) switches or 16 single bit multiplexers and can connect each of its input to 2 outputs via switches. The logarithmic shifter uses 128(2x16x4) switches.

The switch can either be simple transmission gate or $\mathrm{C^2MOS}$ clocked inverter. In this design clocked inverter is used due to its enhanced driving capability. Though the output of the clocked inverter will be inverted, since there are 4 stages, the final output of the logarithmic shifter is not inverted.

4.6.2 Pointer extraction Logic

The output of the DWA has the information of the next pointer value. The control logic blocks shown by dotted lines in Figure. 4.4 extracts this pointer value.

10 Transition detect

The transition detect can be implemented using a single stage of NAND gates. It detects 10 transition(going from LSB to MSB) and not the 01 transition. Since NAND gates are used the outputs are inverted one-hot code(only one of the output line is 0).

One-hot to binary converter

There are several methods to convert one-hot code to binary. The FAT tree structure[13] shown in Figure. 4.5 is used since it is a one of the fastest methods.

Since OR gates are slower, the FAT tree structure is implemented using NAND and NOR gates as shown in Figure. 4.6

Since the output of transition detect is already inverted, the FAT tree block can be directly cascaded.

Pointer Retain Detect

If the thermometer code is all 0's or all 1's corresponding to the two extreme levels, the new pointer value is same as the old pointer. Hence this condition needs to be detected and the old pointer value needs to be retained since there is no 10 transition in these two cases. This can be simply detected from the extreme two comparator outputs. If the 1st comparator(LSB of thermometer code) output is 0 then the thermometer code is all 0's. If the last comparator(MSB of thermometer code) output is 1 then the thermometer code is all 1's. By using a single inverter and a single NAND gate this condition can be detected as shown in Figure. 4.7. The NAND gate output is stored using a flip-flop.

Multiplexer

A 4-bit 2 to 1 multiplexer is used to select either the output of One-hot to binary converter or the previous pointer value based on the pointer retain detector output. The previous pointer value is stored using 4 Flip-flops as shown in Figure. 4.4.

The multiplexers are implemented using C^2MOS clocked inverters.

The output of the multiplexer is buffered and goes as control input to the logarithmic shifter.

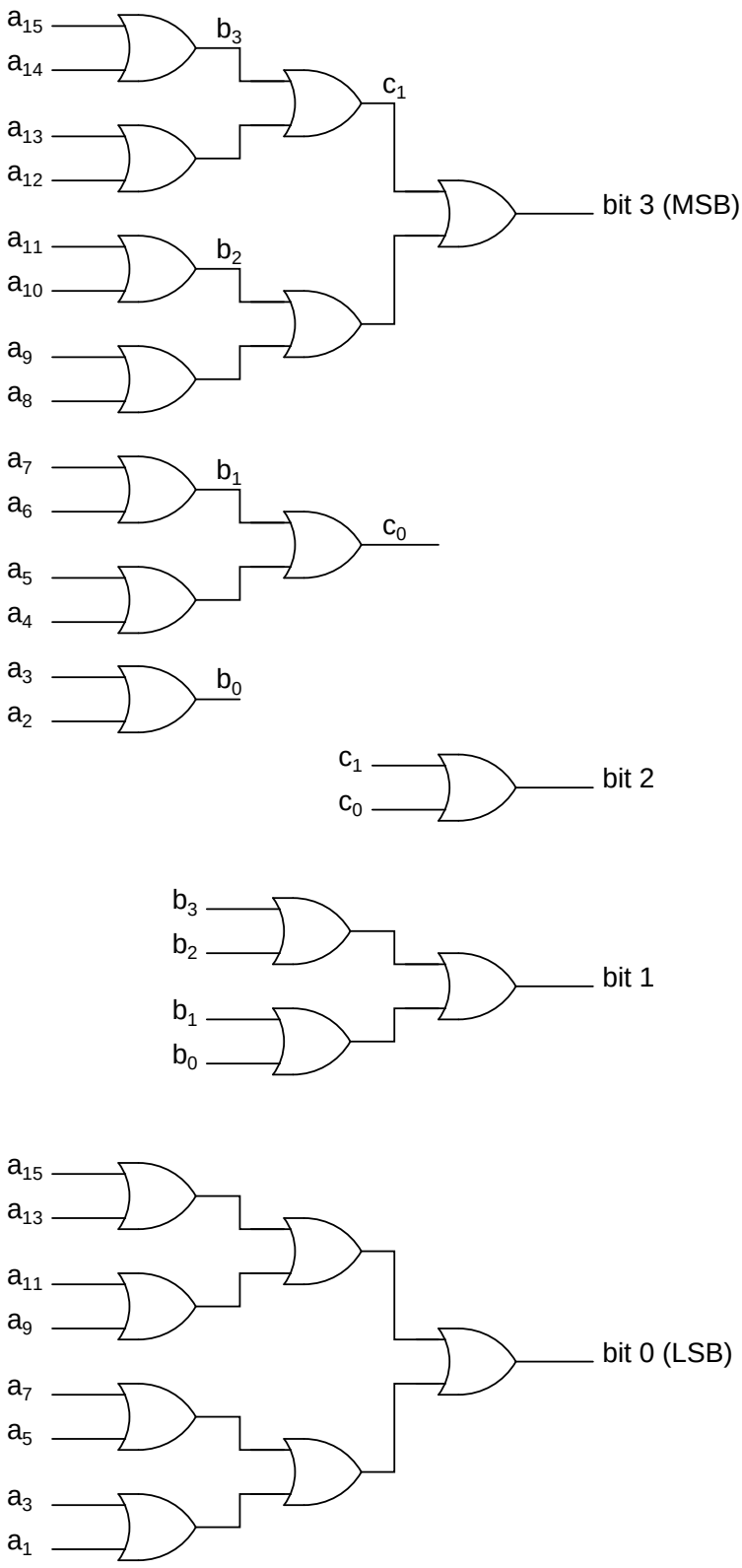


Figure 4.5: FAT tree - One hot to binary

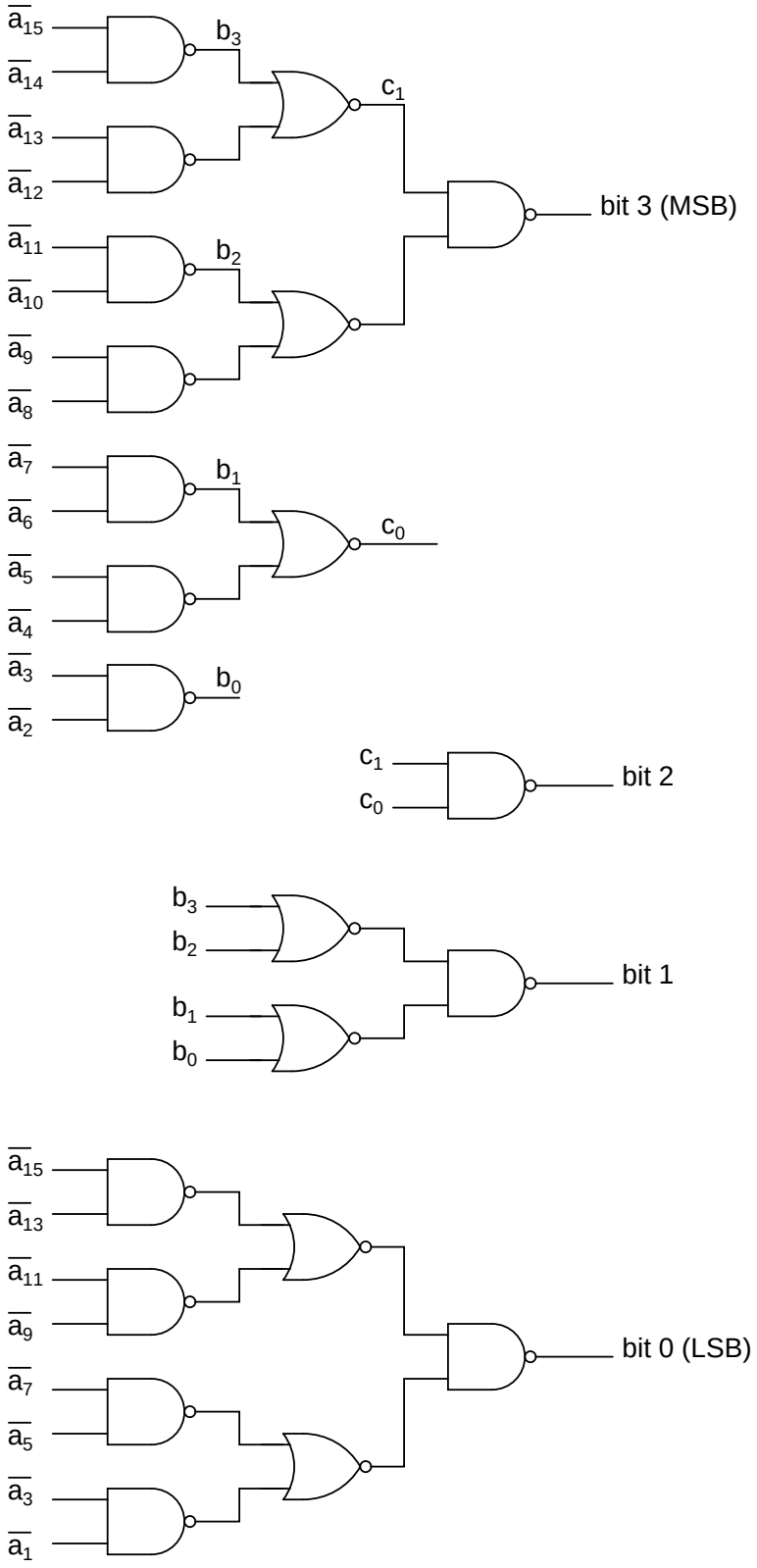


Figure 4.6: FAT tree Implementation

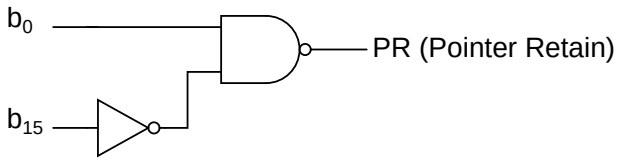


Figure 4.7: Pointer Retail Detector

4.7 Layout

The layout of the DWA is shown in Figure. 4.8

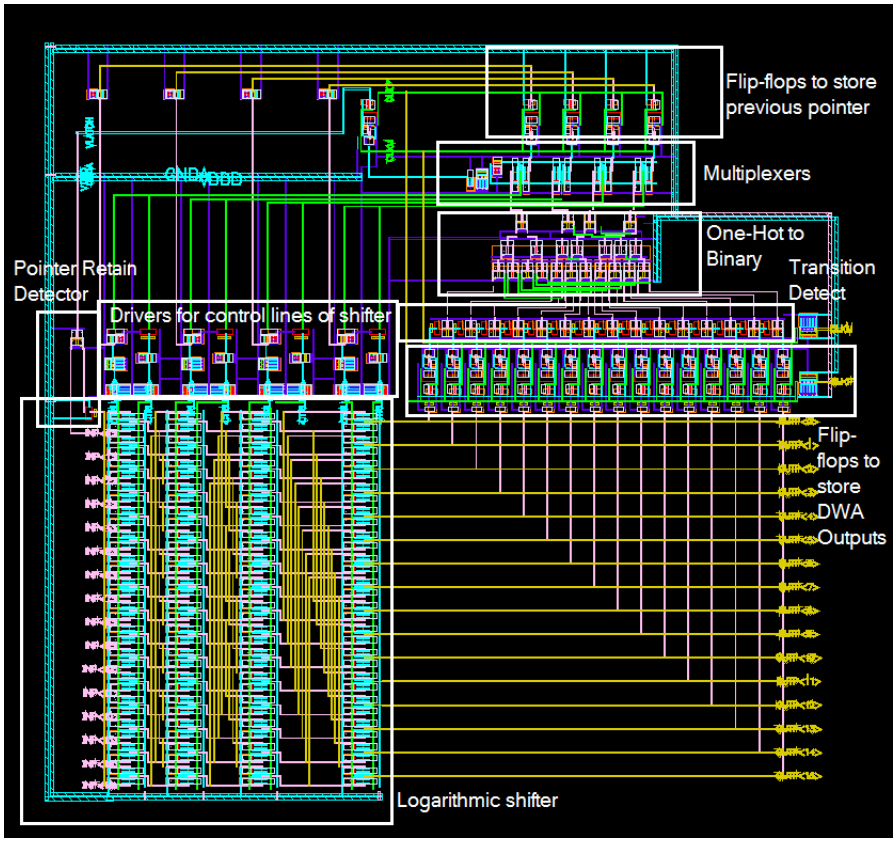


Figure 4.8: Layout of DWA

The dimensions of the DWA are $130\,\mu\mathrm{m}$ x $135\,\mu\mathrm{m}$ and the DWA occupies an area of $0.0175\,\mathrm{mm}^2$.

4.8 Simulation Results

4.8.1 Mismatch Simulation

The CTDSM without DWA was subjected to Monte-Carlo simulation to determine the effect of the mismatch of the slow path DAC resistors. Among the 10 runs, 4 runs returned SQNR below 78 dB(with significant harmonics). After including the DWA, the SQNR improved to 86 dB for all cases.

4.8.2 Open loop SQNR

The ADC along with DWA in open loop is simulated with an input frequency of 1.292 GHz(close to $f_s/2$). The SQNR is tabulated in Table. 4.1 for the worst case scenario(SS corner at 70° C). The SQNR is calculated after passing the output of ADC through an ideal DAC.

Table 4.1: SQNR of open loop ADC with DWA

Type of Netlist	SQNR(dB)
Schematic	26
C+CC	25.2
R+C+CC	24.6

The output spectrum for schematic in SS corner at 70° C is shown in Figure. 4.9.

4.8.3 SQNR of CTDSM loop with ADC-DWA

The ADC along with the DWA is embedded in the CTDSM loop and simulated with an input frequency of 28 MHz. The SQNR is tabulated in Table. 4.2 for SS corner at 70° C. The SQNR is calculated after passing the output of ADC through an ideal DAC.

The CTDSM output spectrum for schematic in SS corner at 70° C is shown in Figure. 4.10.

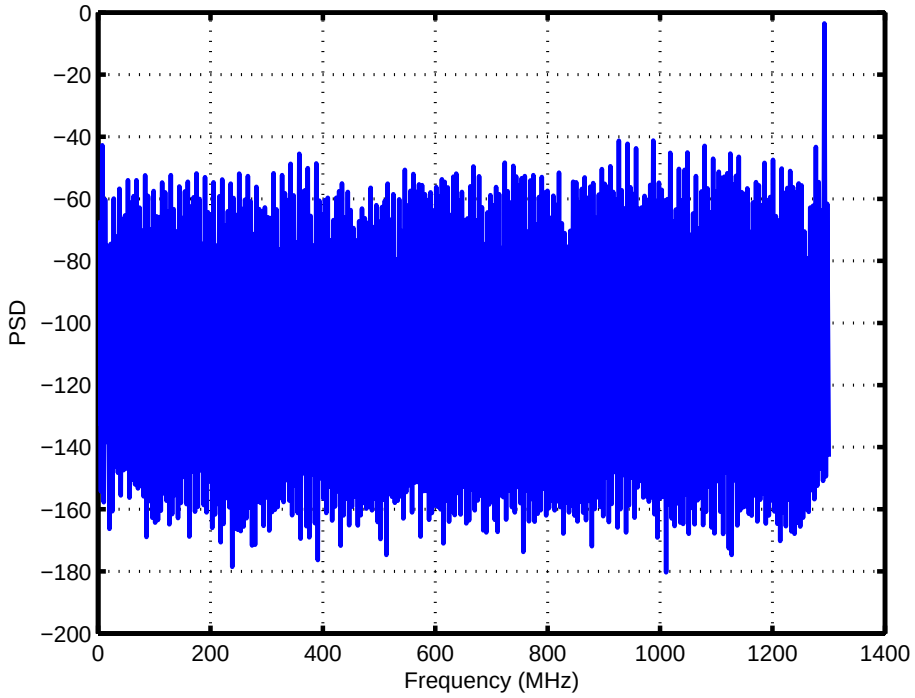


Figure 4.9: Spectrum of DWA output

Table 4.2: SQNR of CTDSM

Type of Netlist	SQNR(dB)
Schematic	87
C+CC	85
R+C+CC	82.7

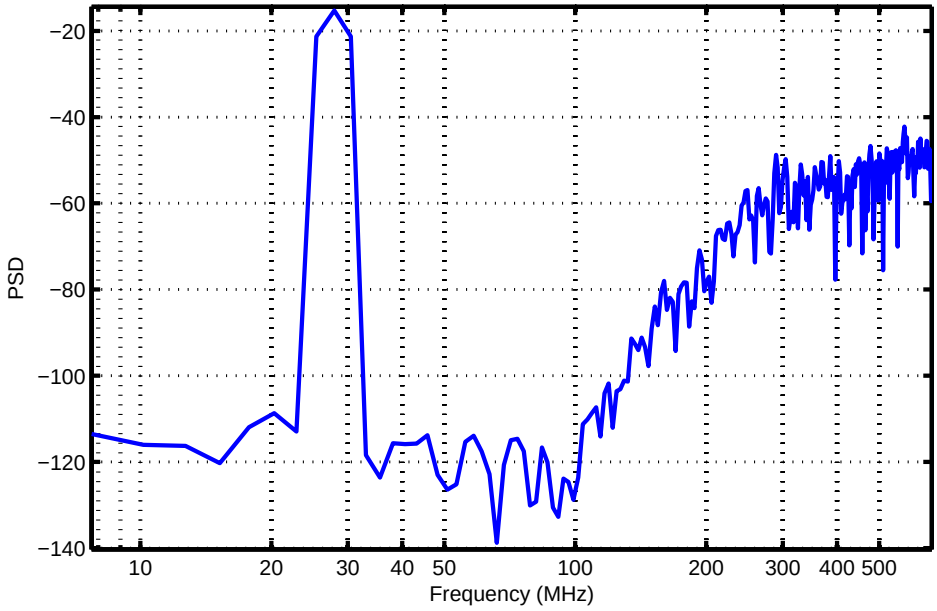


Figure 4.10: Spectrum of CTDSM output with DWA

4.8.4 Power Dissipation

The DWA draws an average current of 4 mA and has a power consumption of 4.8 mW.

CHAPTER 5

Clock Generation and Output Driver Network

5.1 Clock Generation Network

The block diagram of clock generation network is shown in Figure. 5.1.

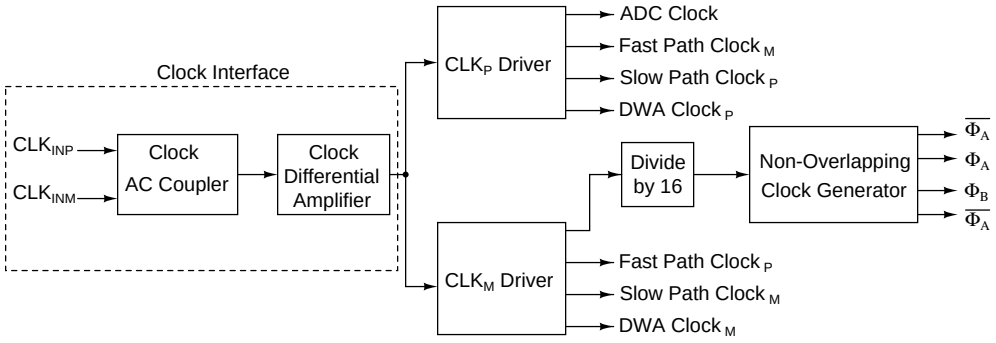


Figure 5.1: Clock Generation Network

5.1.1 Clock AC coupler

The clock AC coupler serves as a high pass filter that filters the noisy external differential clock. The input differential impedance is chosen to be $100\,\Omega$ in order to match the characteristic impedance of the probe that is used to bring the external clock. The clock AC coupler is shown in Figure. 5.2

5.1.2 Clock Differential Amplifier

Due to the high frequency of operation(2.6 GHz) even if the external differential clock has a differential amplitude of 1 V, it will get attenuated severely on passing through the

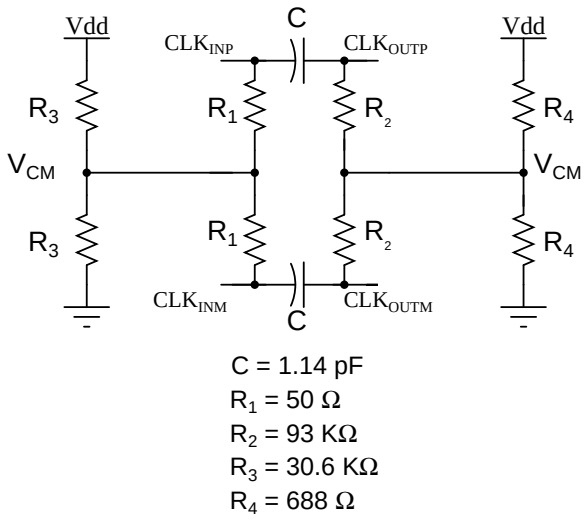


Figure 5.2: Clock AC Coupler

probe and pad rings and it could very well be less than 100 mV once inside the chip. It can also have some residual common mode content that does not get removed by the AC coupler. Moreover even if the external clock is a square wave, due to attenuation it could very well be close to a sine wave on reaching the active part of the chip. These necessitate the need for a differential amplifier for the clock that acts as a sine to square converter.

The differential amplifier is a two stage amplifier with both stages identical as shown in Figure. 5.3.

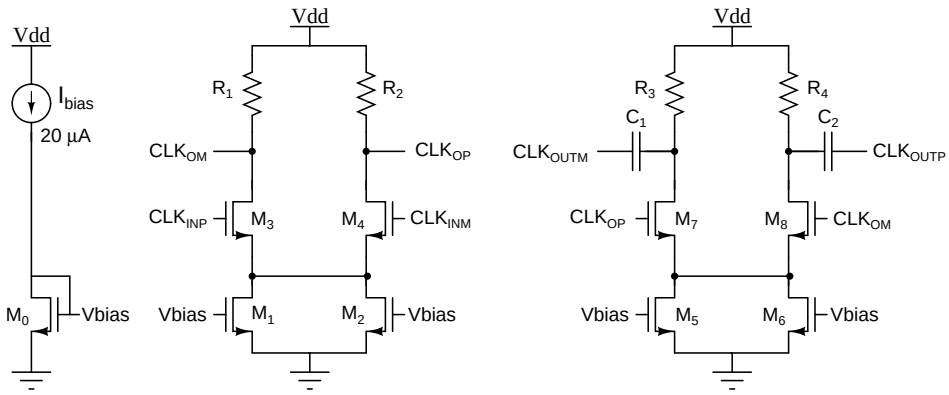


Figure 5.3: Clock Differential Amplifier

All resistors have a resistance of $410\,\Omega$ and both the capacitors have a capacitance of

$185\,\mathrm{pF}$. The sizes of the transistors are given below

Transistor	Width(μm)	Length(μm)
M_0	1.6	0.2
M_1, M_2, M_5, M_6	121.6	0.2
M_3, M_4, M_7, M_8	60	0.1

Each stage of the differential amplifier provides a gain of 3.5 at the clock frequency of 2.6 GHz. The overall gain is around 12 across corners.

5.1.3 Clock Drivers

The clock drivers are a set of inverter chains that generate the various clocks that are used in the CTDSM. The first inverter that interfaces with the differential amplifier is self biased with resistive feedback in order to set the V_{CM} around the midpoint of the supply. The various clocks and the relationship between them are given in Table. 5.1. The ADC clock is used as reference.

Table 5.1: Relationship between different clocks in CTDSM

Type of Clock	Delay with respect to ADC clock
ADC clock	0
Clock to synchronize inputs to fast path NRZ DACs	0.45 $\mathrm{T_s}$
Clock for current steering RZ DACs	0.45 $\mathrm{T_s}$
Slow path clock to synchronize inputs to slow path NRZ DAC	1.1 $\mathrm{T_s}$
DWA clock	0.9 $\mathrm{T_s}$

5.1.4 Clock Divide by 16

Since the clocks to the subtraction circuit is slower($\frac{f_s}{16}$), the f_s clock needs to be divided. This can be achieved by using T-Flip-flops as shown in Figure. 5.4.

The T-Flip-flops have their T inputs tied to 1. Such T-Flip-flops can be made using D-Flip-flops in the fashion shown in Figure. 5.5.

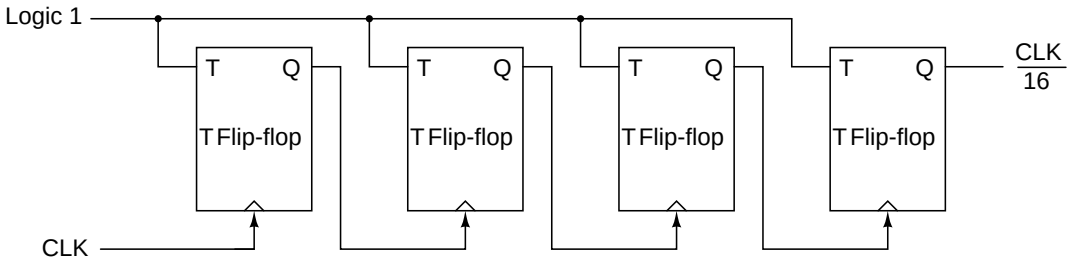


Figure 5.4: Divide by 16 using T-Flip-flops

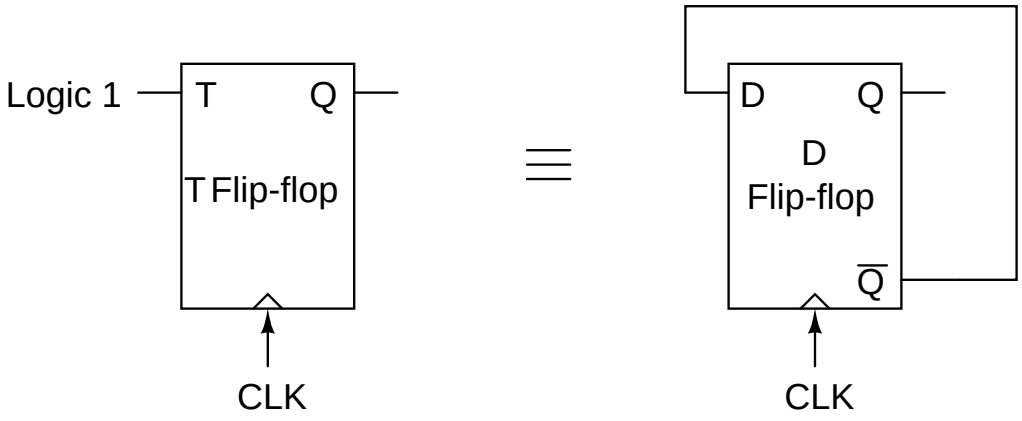


Figure 5.5: T-flip-flop using D-Flip-flop

The D-Flip-flops are implemented using True Single Phase Clock(TSPC) architecture shown in Figure. 5.6.

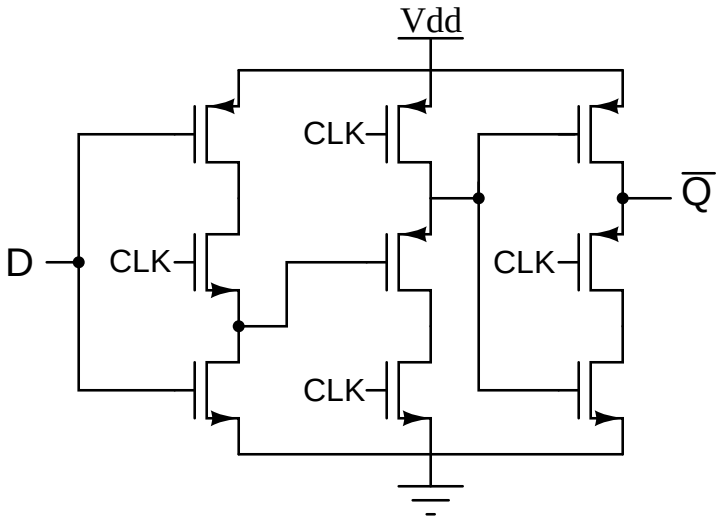


Figure 5.6: TSPC D-Flip-flop

It uses only single phase clock and since its output is already inverted it is easy to obtain a T-Flip-flop by shorting $\overline{Q}$ output and D input.

5.1.5 Non-Overlapping Clock

Since non-overlapping clocks are needed for the subtraction circuit they are generated using the circuit shown in Figure. 5.7.

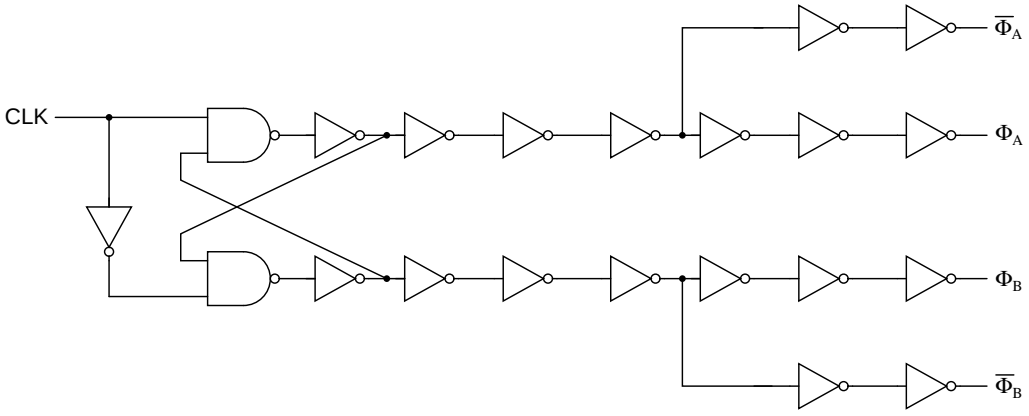


Figure 5.7: Non-Overlapping Clock Generation

5.1.6 Layout

The layout of the clock interface network is shown in Figure. 5.8.

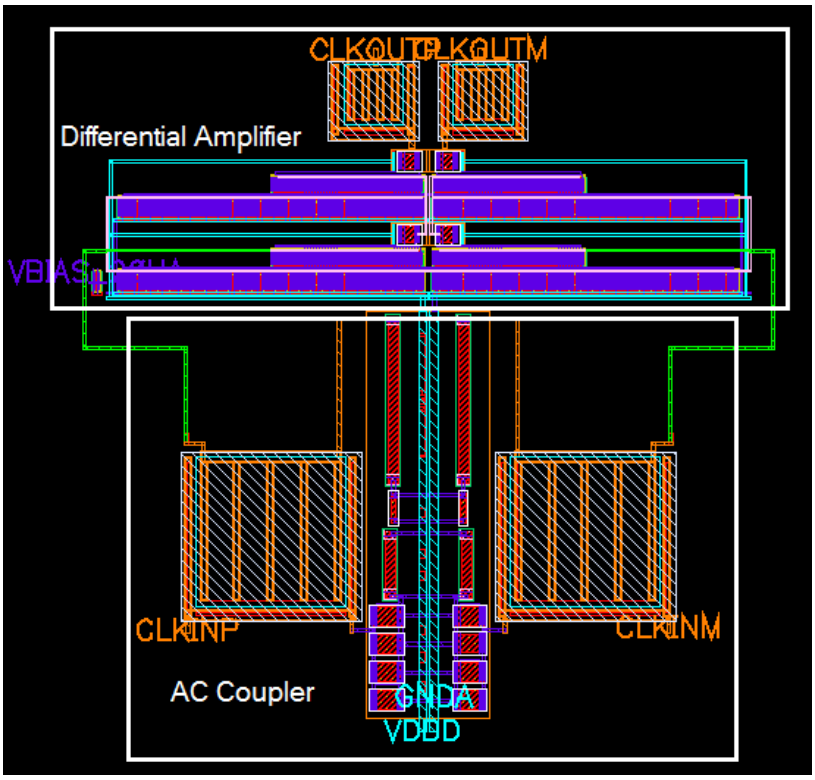


Figure 5.8: Clock Interface Layout

The layout of the clock divide by 16 along with the non-overlapping clock circuit is shown in Figure. 5.9.

5.1.7 Simulation Results

The CTDSM was simulated with the clock generation network. There is no degradation in SNR when the ideal clock network was replaced by real extracted components.

The clock interface networks consumes 8.5 mW of power. The rest of the clock generation network consumes 4.2 mW of power.

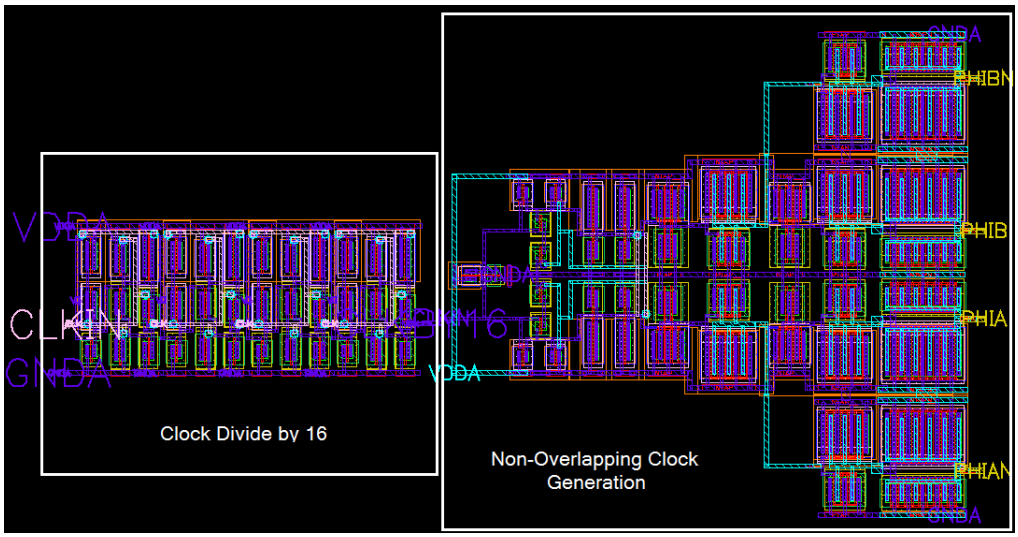


Figure 5.9: Layout of the Non-Overlapping $\frac{f_s}{16}$ Clock Network

5.2 Output Driver Network

5.2.1 Fast Path Driver

The fast path consists of 5 DACs of which 3 are NRZ and two are RZ CS DACs.

For the NRZ DACs its inputs need to be synchronized at 0.45 T_s. For this purpose we can use a flip-flop. But since a flip-flop can be considered as a cascade of two latches, SR latch is already present in the comparator. Hence we require only a D-Latch for synchronization. A C^2MOS D-Latch is used. The D-latch output is buffered and given to the NRZ DACs.

The two RZ CS DACs have their own 0.45 T_s clock. So the input to these RZ DACs need not be synchronized. So the ADC comparator outputs are just buffered before reaching the RZ DACs.

The fast path driver consumes 12.7 mW of power. In order to interface with the loop filter, which has supply voltage of 1.6 V against the ADC supply voltage of 1.2 V, the last stage inverters of the driver have a supply voltage of 1.6 V. Since the input logic 1 to these inverters are 1.2V, the P-MOS transistors of these inverters do not turn off completely, leading to increase in leakage currents and hence increase in power

dissipation.

5.2.2 Slow Path Driver

The slow path consists of a single NRZ resistive DAC. Its inputs need to be synchronized at 1.1 T_s. This is achieved using transmission gates based flip-flops as shown in Figure. 5.10. Transmission gate flip-flops are used instead of C^2MOS flip-flops for their cleaner output(devoid of clock feed-through). To prevent the intermediate node(V_x) from floating, during the negative phase of clock, the output of the flip-flop is fed back.

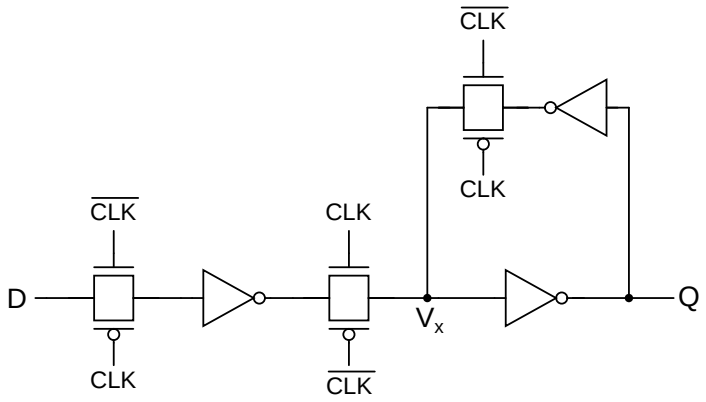


Figure 5.10: Transmission gate based Flip-flop with feedback

The slow path driver consumes 5.67 mW of power. Here also the P-MOS transistors of last stage inverters do not turn off completely, leading to increase in leakage currents and hence increase in power dissipation.

5.3 Overall CTDSM Power Consumption

The CTDSM power consumption split up is given in Table. 5.2.

The clock interface network that consists of Clock AC Coupler and Clock Differential Amplifier(sine to square converter) can generally be excluded from the CTDSM power consumption.

Table 5.2: CTDSM Power Consumption

Block	Power Consumption (in mW)
ADC	5.4
DWA	4.8
Loop filter and DACs	37
Output Drivers	18.3
Clock Generation network	12.7
Clock Generation network(excluding Clock Interface)	4.2
CTDSM	78.2
CTDSM(excluding Clock Interface)	69.7

CHAPTER 6

Novel Clockless Return-to-Zero DAC

6.1 RZ DAC Element

RZ DACs are commonly used in CTDSMs to enhance the feedback content and stabilize the CTDSM loop by compensating for the excess loop delay. RZ DACs are clocked and the schematic of a Current Steering(CS) RZ DAC element is shown in Figure. 6.1.

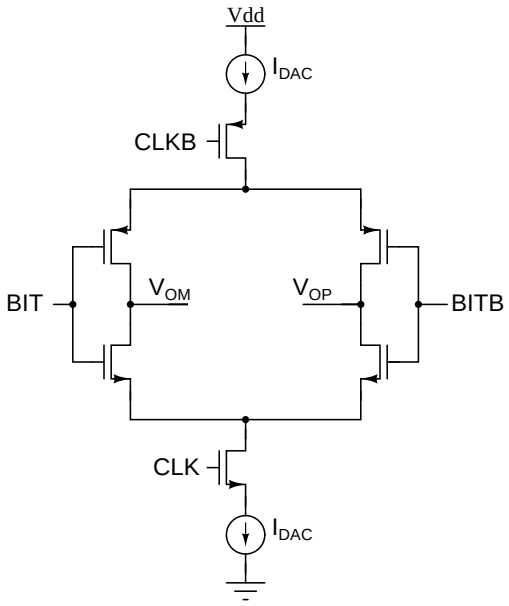


Figure 6.1: Current Steering RZ DAC Element

When the clock is OFF the current sources enter triode region. When the clock turns back ON, these sources take some time to come back to saturation and provide the required current. At high speeds this time taken is significant and leads to non-linearities. Hence the current sources are retained in saturation by providing a path for the current when the clock is OFF as shown in Figure. 6.2.

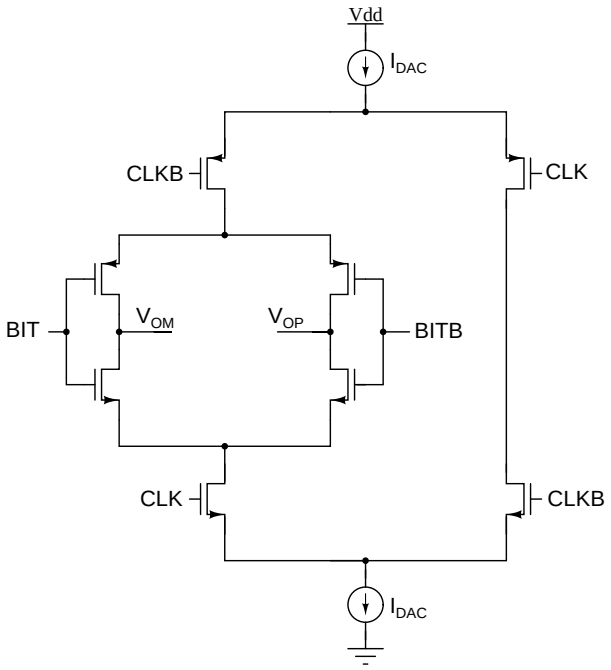


Figure 6.2: Modified Current Steering RZ DAC Element

For a multi-bit CTDSM there will be several such DAC elements. Generating and driving clocks for such DAC elements becomes cumbersome and adds to power dissipation. An RZ Current Steering DAC without explicit clock seems to be an exiting alternative.

6.2 Clockless RZ DAC motivation

The clocked regenerative latch outputs remains valid for about half time period and automatically gets reset to logic 1(Vdd). A method can be devised to use these outputs to drive the CS DAC and achieve RZ effect.

6.3 Clockless RZ DAC Implementaion

There are many ways to implement Clockless RZ current steering DACs controlled by the regenerative latch outputs. One method is to use the existing RZ current steering DACs and generate the clock using NAND gate controlled by the outputs of a regener-

ative latch. In the regeneration phase the NAND gate output is 1 and in the reset phase the NAND gate output is 0(since both inputs to the NAND gate are reset to 1) there by acting as a clock. But this method requires extra circuitry for NAND and buffering of the clock. Moreover the clock so generated may come earlier than the Data for some DAC elements thereby requiring extra circuitry to avoid such problems.

Another implementation that avoids the need of extra circuitry is shown in Figure. 6.3. The DACs are controlled by the Regenerative latch outputs(which reset to logic 1 after half clock cycle thereby producing the RZ effect).

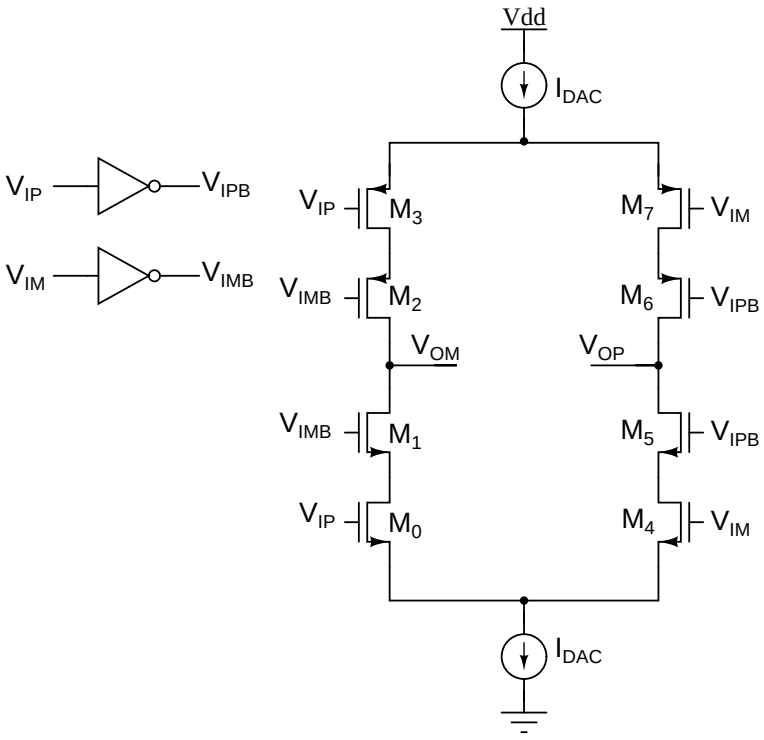


Figure 6.3: Clockless Current Steering RZ DAC Element

The Table. 6.1 explains the working of the Clockless Current Steering(CS) Return to Zero(RZ) DAC.

The current sources can be retained in saturation by using the modification shown in Figure. 6.4.

Table 6.1: Clockless RZ CS DAC operation

Input Combination		Result
V_{IP}	V_{IM}	
1	0	Current pushed into V_{OP}, Current pulled from V_{OM}
0	1	Current pushed into V_{OM}, Current pulled from V_{OP}
1	1	Both V_{OP} and V_{OM} disconnected from current sources

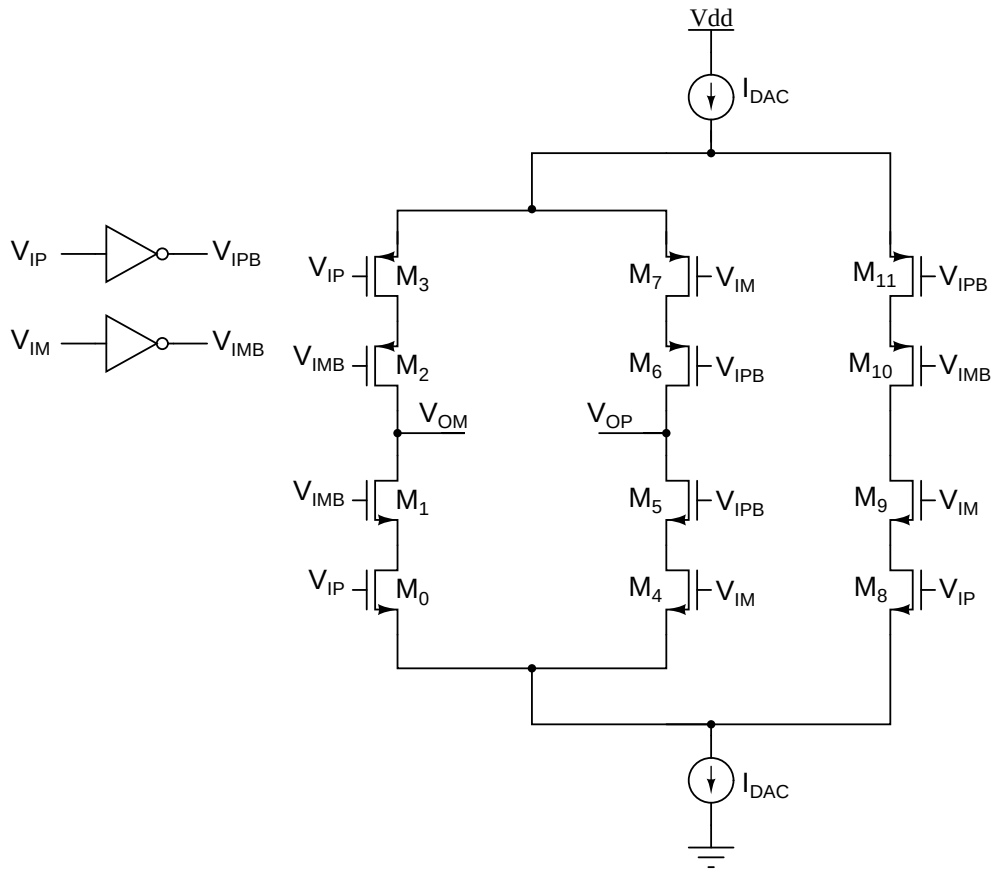


Figure 6.4: Modified Clockless CS RZ DAC Element

6.4 Non-Ideality in Clockless RZ DAC

The differential inputs to the 16 comparators after reference subtraction are different. The comparator that sees a larger differential input regenerates faster. Hence the corresponding clockless DAC element turns on earlier than other elements. But all the clockless DAC elements turn off around the same time since the resets are synced. Some sample DAC pulses of the Clockless DAC elements are shown in Figure. 6.5. This non-ideality can be thought of as an error pulse similar to the quantization error which can be approximated to have a white(flat) spectrum. Since this error gets added to the inner most loop it gets shaped and most of its content is swept out of the signal band. But since a small portion remains it can lead to rise of noise floor.

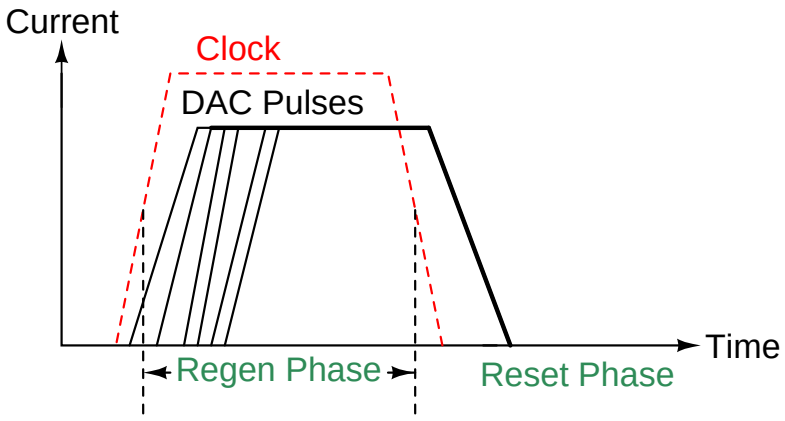


Figure 6.5: Clockless RZ DAC Element pulses

6.5 Simulation Results

The clocked RZ DAC elements are replaced with the clockless RZ DAC elements and the CTDSM is simulated with an input frequency of 28 MHz. The loop is stable. The spectrum is shown in Figure. 6.6.

There is no harmonic distortion(2nd and 3rd harmonics). There is no out of band peaking. But the noise shaping in the signal band is not clearly visible due to rise in the noise floor. Due to the noise floor rise the SQNR has come down to 80 dB. By changing the

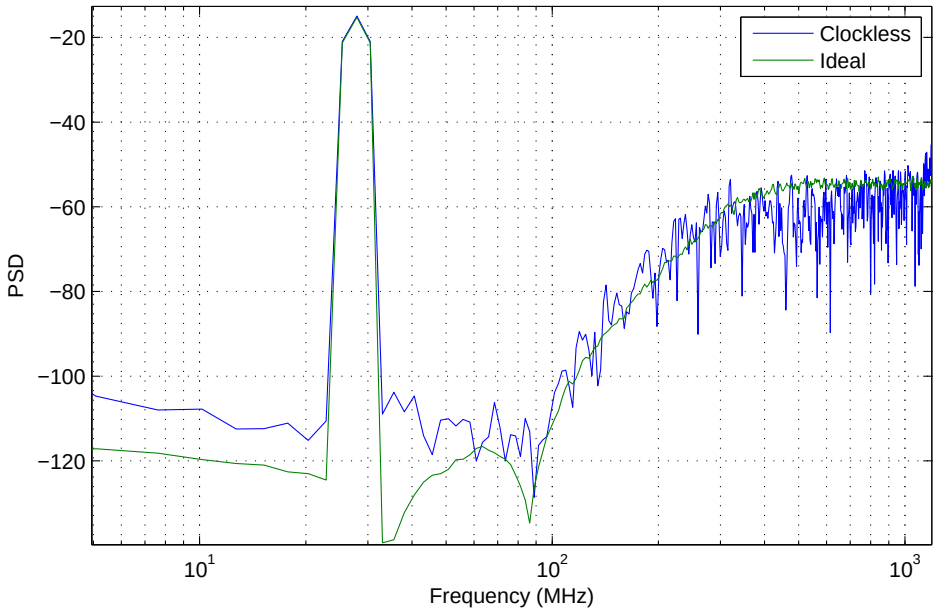


Figure 6.6: CTDSM Spectrum with Clockless RZ DAC

currents of the clockless RZ DAC elements the SQNR improved to 82 dB.

CHAPTER 7

Conclusion and Future work

In this work the implementation of a high speed ADC working at 2.6 GHz is presented. The open loop SQNR of the ADC is 25 dB. The ADC is embedded in a CTDSM loop designed for a wide bandwidth of 100 MHz at a sampling Frequency of 2.6 GHz. The SQNR of the CTDSM loop for a -3 dBFS input signal at 30MHz frequency is 83 dB. A new high speed, power efficient implementation of Data Weighted Averaging(DWA) is also presented. The inclusion of DWA did not affect the SQNR of the CTDSM loop. It improved the performance of the CTDSM when there is mismatch of DAC elements. The power dissipation of the CTDSM is 78.2 mW . The power dissipation of the combination of ADC, DWA, clock generation and output driver network is 41.2 mW. A novel clockless RZ DAC is also presented. The feasibility of its inclusion in the CTDSM loop is verified. Though the CTDSM loop is stable there is SQNR degradation due to noise floor rise.

The Clockless RZ concept can further be analyzed and a technique to model the varying rise times of the current pulses can be determined. This model can then be used to fit the NTF to improve the SQNR.

REFERENCES

[1] Bolatkale, Breems, Rutten, and Makinwa, "A 4-GHz Continuous-Time Delta Sigma ADC with 70-dB DR and -74 dBFS THD in 125-MHz BW," IEEE Transactions on Circuits and Systems-II:Express Briefs, vol. 46, no. 12, September 2011.

[2] P. Shettigar and S.Pavan, "A 15 mW 3.6 GS/s CT-Delta Sigma ADC with 36MHz Bandwidth and 83dB Dynamic Range in 90 nm CMOS," IEEE International Solid State Circuits Conference (ISSCC) ,San Francisco, February 2012.

[3] R. Schreier and G. Temes, "Understanding Delta-Sigma Data Converters," IEEE Press, New Jersey, 2005.

[4] W. L. Lee, "A novel higher-order interpolative modulator topology for high resolution oversampling A/D converters," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, June 1987.

[5] F. Chen and B. Leung, "A high resolution multibit sigma-delta modulator with individual level averaging," IEEE Journal of Solid-State Circuits, vol. 30, no. 4, pp. 453-460, April 1995.

[6] Baradwaj.V, "Design of 4 bit flash ADC for use in a Delta Sigma Modulator," B.Tech thesis, Indian Institute of Technology Madras, Chennai, 2008.

[7] J. Kim, B. S. Leibowitz, J. Ren, and C. J. Madden, "Simulation and Analysis of Random Decision Errors in Clocked Comparators," IEEE Transactions on Circuits and Systems-I: Regular Papers, vol. 56, no. 82, pp. 1844-1857, August 2009.

[8] S. Pavan and N. Krishnapura, "Oversampling Analog-to-Digital Converter Design," 21st International Conference on VLSI Design(VLSID), pp. 7-7, 2008.

[9] L. Carley, "A noise-shaping coder topology for 15+ bit converters," IEEE Journal of Solid-State Circuits, vol. 24, no. 2, pp. 267 - 273, April 1989.

[10] D.-H. Lee, T.-H. Kuo, and K.-L. Wen, "Low-Cost 14-Bit Current-Steering DAC With a Randomized Thermometer-Coding Method," IEEE Transactions on Circuits and Systems-II:Express Briefs, vol. 66, no. 2, pp. 137 - 141, February 2009.

[11] M. Neitola and T. Rahkonen, "A Generalized Data-Weighted Averaging Algorithm," IEEE Transactions on Circuits and Systems-II:Express Briefs, vol. 57, no. 2, pp. 115 - 119, February 2010.

[12] I. Galton, "Why Dynamic-Element-Matching DACs Work," IEEE Transactions on Circuits and Systems-II:Express Briefs, vol. 57, no. 2, pp. 69-74, February 2010.

[13] D. Lee, J. Yoo, K. Choi, and J. Ghaznavi, “Fat tree encoder design for ultra-high speed flash A/D converters,” IEEE Midwest Symposium on Circuits and Systems (MWSCAS), August 2002.