8 Bit Current Mode ADC For

QIF Neural Model

A Project

Presented To

The Faculty of the Department of Electrical Engineering

In Partial Fulfillment of the Requirements for the Master of Science Degree in Electrical Engineering

By

Mandeep Singh

SJSU ID: 006521372

May 2011

8 Bit Current Mode ADC For

QIF Neural Model

Submitted By

Mandeep Singh

SJSU ID: 006521372

	Signature	Date
Dr. David W. Parent (Advisor)		5/13/2011
Dr. Eric Basham (Co-Advisor)		5/12/2011
Dr. Tri Caohuu (Graduate Co-ordinator)		5/16/2011

ABSTRACT

To provide digital data to a digital implementation of a Quadratic Integrate and Fire point source neural model form an analog source a data converter is required. To meet this need an eight bit current mode ADC has been designed. The eight bit ADC uses an algorithmic architecture in which each bit cell is designed using active current mirrors. Current mode technique eliminates the use of precision components including capacitors and resistors, typically employed in voltage mode ADC architecture. The ADC operates at 100 K Samples/sec and input signal to the ADC has a bandwidth of 25 KHz. The ADC is implemented in TSMC 0.25 µm technology.

ACKNOWLEDGEMENTS

. I would like to express my thanks to my Advisor Prof. David Parent and my Co-Advisor Prof. Eric Basham who guided me throughout and made this project a success. They have been so generous to take time out of their busy schedule and guide me whenever I got stuck. I would also like to thank my parents, without their support obtaining masters degree was a distant dream.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Circuit implementation of neural models help in emulating neurobiological systems [17]. Sensing external analog voltages and converting them into digital format for further processing is one of the important blocks in an artificial neural network. This conversion is carried out by an Analog to Digital Converter (ADC). Analog to Digital Conversion is the key building block, which sits in the middle of analog and digital blocks. Fig 1-1 shows the application of ADC where a digital processing blocks requires a digital input.

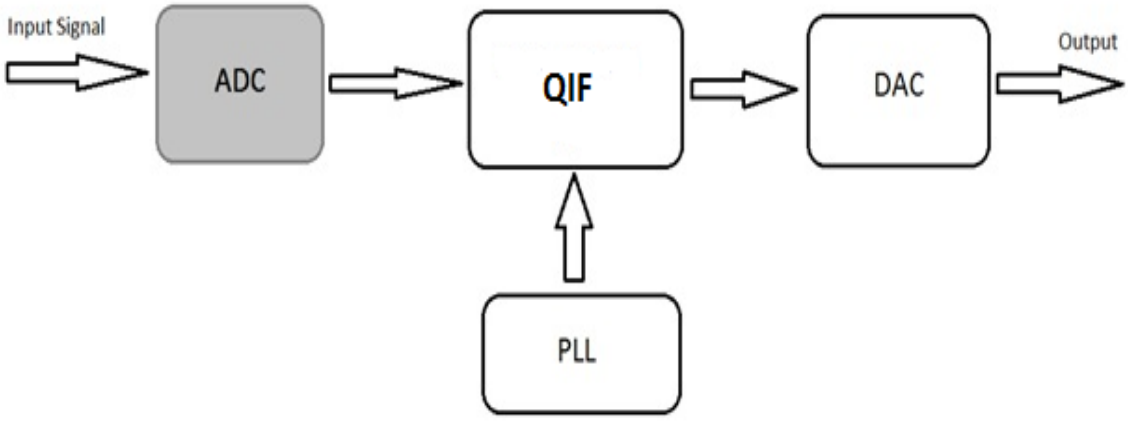

Fig 1-1: ADC application.

Since, the input signal is analog in nature an ADC is used to convert it into digital format, which goes to the QIF (Quadratic Integrate and Fire) block as shown in Fig 1-1. This project implements an ADC for QIF neural model block.

1.1 Specification for the desired application

The application for which the ADC is targeted is for QIF neural model. This application requires medium resolution and low sample rate type of ADC [2]. The input single frequency varies from 100 Hz to 25 KHz. Resolution of the ADC is 8 bits. The project is implemented using TSMC 0.25µm technology with 2.5 volt V_{dd}. Area usage is also very important to fit the entire project in a MOSIS research tiny chip.

1.2 Literature Review

The application demands a medium resolution and low sample rate therefore architecture that results in such specifications should be used. Fig 1.2 shows the

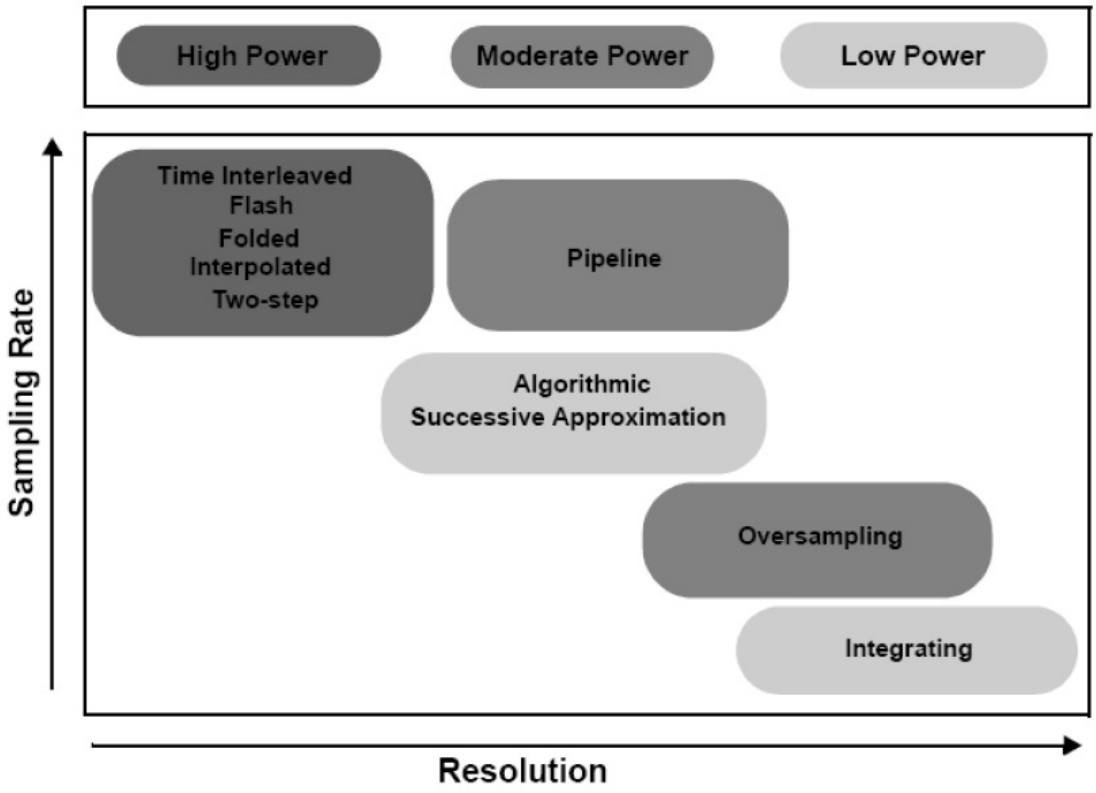

Fig: 1-2: Evolution of ADC architectures [18]

evolution of ADC from sampling rate and resolution point of view [18]. A flash ADC uses a resistor ladder and comparators in its design [19]. The resistor ladder is both area and power hungry so this architecture is not suitable for this application. The flash ADC has the advantage of operating it at high frequencies. The pipelined ADC comprises of an operational amplifier, a sample and hold, comparator blocks. This also results in high area consumption [19]. High resolution applications involve use of delta sigma architecture. This architecture involves complex circuitry which uses capacitors and filters. As neural applications do not require high resolution therefore, use of delta-sigma architecture is eliminated.

Successive approximation type ADC is suitable candidate for the desired application as it occupies less area, is medium resolution and the desired frequency is achievable using the SAR architecture. This architecture has been implemented in [20], [21] and [22] for bio-medical applications.

In [20] the area and power is less and sample rate as desired in achieved but, it uses switches for variable resolution. Control of switches may result in increase of complexity. In [21] the architecture occupies low area but the sample rate is only 20 K Sample/sec. The design presented in [22] meets the requirement from area, resolution and sample rate point of view but again it involves use of switches.

Analyzing the architectures mentioned above current mode algorithmic architecture was chosen because of its ease of implementation and simple architecture. The architecture eliminates the use of precision components for e.g. capacitor used in OTA for compensation and sampling capacitor.

1.3 Project contribution, motivation and report organization

This project implements a current mode algorithmic ADC that improves and eliminates the shortcomings of the ADCs previously implemented in [1] and [3] by introducing track and hold block at the input and D flip flops at the output.

As technology advances, it is important to minimize fabrication cost without sacrificing performance parameters. Depending on the application the resolution and sampling rate is specified therefore designers are left with optimizing the area and power. The main motivation of the algorithmic ADC implemented described here lies in the fact that it will occupy less area, which will in turn save fabrication cost.

The first part of this project report reviewed ADC's and covered prior literature. The second part of this report discusses the ADC design and implementation. The third part covers the physical implementation of the ADC. The fourth part describes about simulation results. Finally, chapter 5 discusses future improvements and concludes.

CHAPTER 2

CUREENT MODE ALGORITHMIC ADC DESSIGN

After literature review and application considerations, current mode algorithmic ADC architecture was chosen because of complicated simpler design and low area occupancy [1]. The main advantages of this architecture are:

- Easy implementation.
- No precision capacitors required.
- No precision clock required.
- Occupies less area as compared to other ADC architectures.

Current mode algorithmic ADCs comprise a series of 1 bit cells, track and hold block, current comparator, OTA and D- flip flop. The use of the architecture defined in [1] is limited because it functions only for one particular frequency. This limitation arises because every different input signal frequency results in a different conversion time. This in turn results in uncertainty of when we can expect the digital output for a particular signal level. In neural recording applications the signal frequency varies from 100 Hz to 6 KHz [2]. For this reason, the architecture should support a range of frequencies. The architecture used in [3] is pipelined version of the architecture in [1]. This architecture has increased throughput but requires sample and hold circuitry between every bit cell. To control each sample and hold block requires precision clock resulting in increase of complexity. It also results in increase of area.

The proposed design incorporates the ideas used in both [1] and [3] but overcomes their drawbacks. The input current goes into a current mode track and hold block, the output of track and hold block goes into 8 identical bit cells to generate the digital output. During the hold phase a constant current goes from the track and hold block into each one of the 8 bit cells thus digital output of each bit cell is also constant. Taking advantage of the hold phase the digital output is latched onto a D-flip flop in middle of hold phase. During the track phase the digital output of each bit cell won't be constant, depending on the input signal level, making it difficult to process the changing digital output. Implementing a D-flip flop at the output of each bit cell provides a useful function to stabilize the output. The output of D-flip flop remains the same during the track phase thus eliminating changing digital levels as a function of the input. Having current mode track and hold block at the input and D-flip flops at the output also results in synchronously clocked circuit.

2.1 Current mode algorithm

The algorithm for this current mode ADC implementation is based on two mathematical equations:

$$I_{out} = \begin{cases} 2I_{in} \\ 2I_{in} - I_{ref} \end{cases} if \begin{cases} 2I_{in} < I_{ref} \\ 2I_{in} > I_{ref} \end{cases}$$

Where I_{out} is the output current of bit cell, I_{in} is the input current to the bit cell and I_{ref} is the reference current. I_{ref} is also termed as the full scale current.

Input current I_{in} in the first bit cell is amplified to twice its value. This value is compared with a reference current. If twice the input current is greater than the reference current the bit cell generates logic high, otherwise the output of the cell will be logic low. If bit cell generates logic high the output current from the bit cell is difference of twice the input current and reference current. If the digital out is logic low then twice of input current is passed onto identical next stage. The bit cells are cascaded such that output of one cell is input of the next cell. The first bit cell that gets the input signal gives the MSB

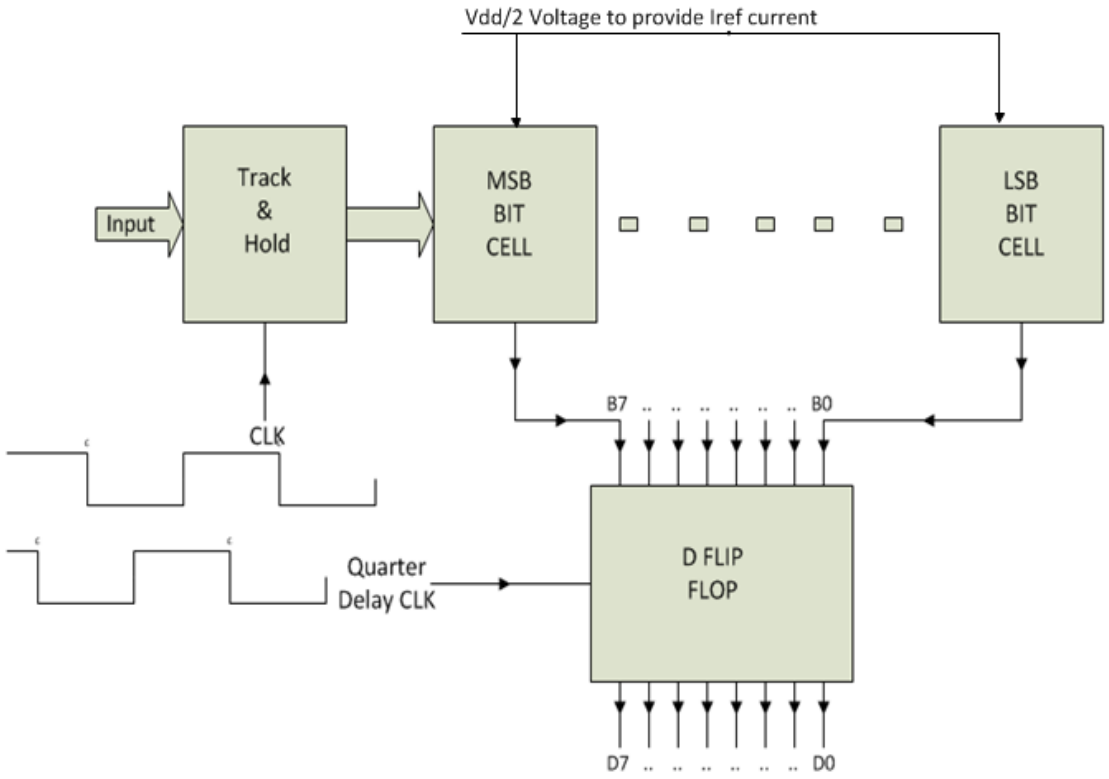

Fig. 2-1: Complete block diagram of the ADC

and the last bit cell in the chain gives LSB. The complete block diagram of the ADC is as shown in Fig. 2-1.

2.2 One Bit Cell

The circuit implementation of current mode architecture is shown in Fig 2-2. Input current I_{in} goes into the drain of nmos N1. Twice of the input current is then mirrored in nmos N2, whose width is twice as big as nmos N1. The size of N3 and N4 is the same as N1. A reference voltage, $V_{dd}/2$ is applied at gate of nmos N3 and N4 to generate reference current Iref. Twice of the input current generated from N2 is mirrored into pmos P2.

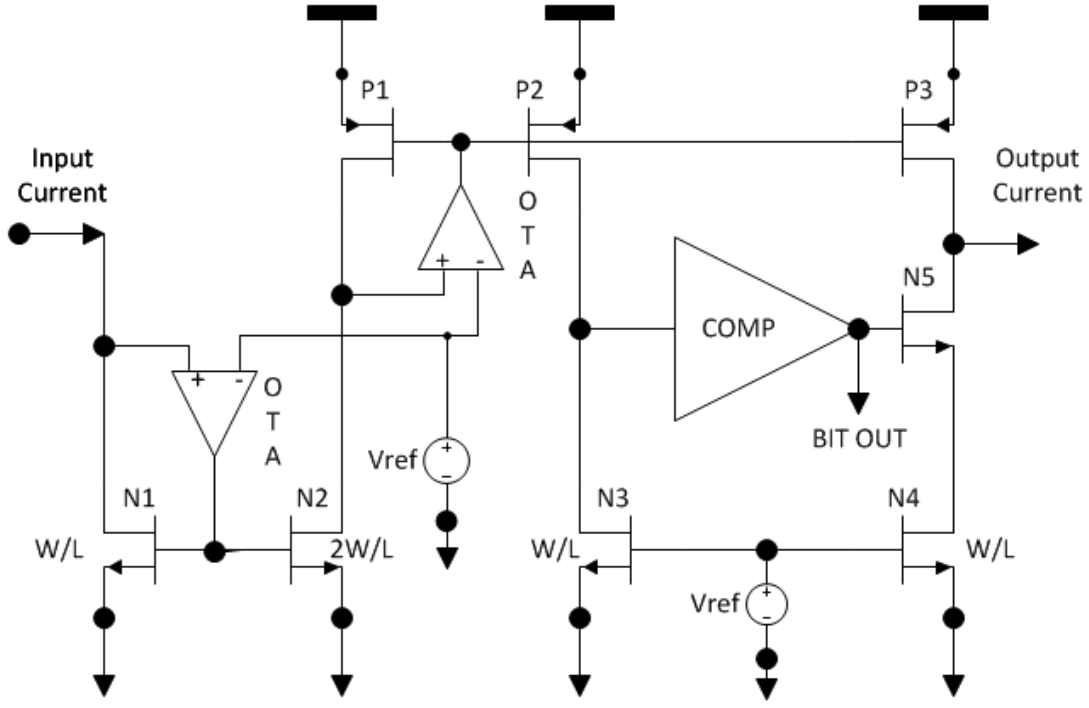

Fig 2-2: One Bit Cell

Current mirrored by pmos P2 and current mirrored by nmos N3 cancel each other thus resulting in a voltage drop across P2. Their drain is connected to digital buffer whose output is high if $2I_{in}$ is greater than I_{ref}. The output of buffer is low if $2I_{in}$ is less than $I_{ref.}$

Depending on the output of buffer, nmos N5 acting as a switch turns on or off. If it's on, the output current I_{out} is $2I_{in} - I_{ref}$. If the switch is off the output current is $2I_{in}$. The output of digital buffer, BIT OUT, acts as the digital output of ADC. Cascading of the 8 bit cells, with the analog output of one going to the analog input of the next cell, results in an ADC with 8 bit resolution.

It's very important to for the current mirroring operation to happen with little error. Thus, the current mirror branches in the bit cell should exhibit very good matching. From a physical implementation point of view the transistors should be matched and their characteristics should not vary drastically due to process variations. To have good current matching the transistors should have high output resistance [4]. Increasing the transistor length results in increased output resistance but, the circuit's operating frequency is also reduced. Noise performance also becomes poorer with increase in length. Therefore, choosing the optimal length plays a key role.

As explained in [1], a bit cell with normal current mirror where drain and gate are connected a maximum of 6 bit resolution is achieved because this circuit is susceptible to threshold voltage mismatch and finite output resistance effects. Cascoded current mirror architectures are limited to 7 bits of resolution because of their high threshold voltage mismatch even though the output resistance is increased.

To obtain 8 bit resolution, current mirror architecture should be used that does not introduce current mirroring error due to threshold voltage variations and low output impedance. In Fig 2-3 the drain of nmos N1 and nmos N2 are connected to non inverting terminal of an OTA. If inverting terminal of both the OTAs is connected to a reference voltage then the non inverting terminal will follow the voltage at the inverting terminal because of the virtual ground concept [5]. Reference voltage should be midpoint of V_{dd}. The gate voltage of nmos N1 will be given by:

$$V_{gs,N1} = A(V_{ds,N1} - V_{dd}/2)$$

Where, A is the open loop gain of OTA. $V_{ds,N1}$ is the drain potential of nmos N1. $V_{gs,N1}$ is the gate to source voltage of nmos N1 and V_{dd} is the supply voltage.

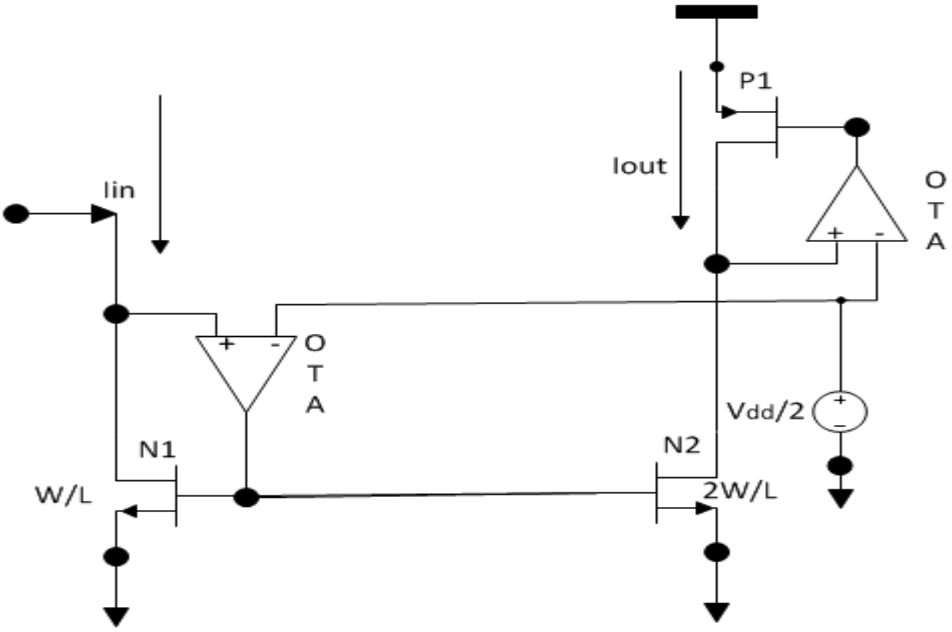

Fig 2-3: Active current mirror using OTA

The larger the gain of OTA, the smaller the difference between inverting and non-inverting terminals given an offset at the gate potential at nmos N1. Smaller difference between inverting and non-inverting terminals results in reduction of current mirroring error.

Connecting the mirroring branches as shown in Fig. 2-3 reduces current error due to threshold voltage variations between noms N1 and N2. Also, high output impedance is not required using this configuration. Keeping the drain voltages constant at input and output nodes irrespective of device gate voltage is the primary advantage of the active current mirror architecture.

2.3 Operational Transconductance Amplifier (OTA)

As explained in the previous section OTA is used to regulate the drain potential of

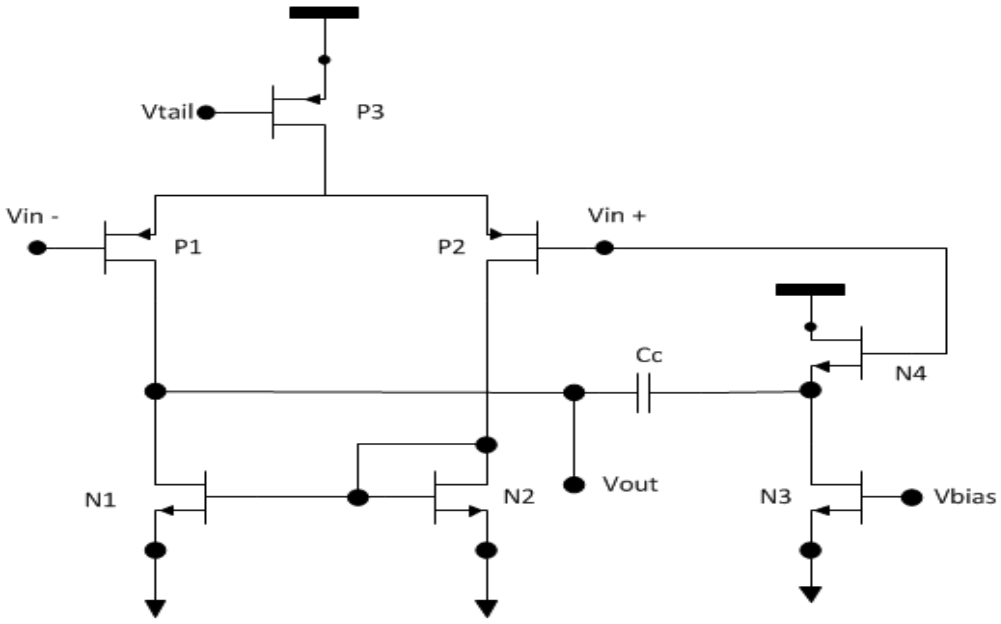

Fig 2-4: Operation transconductance amplifier

the mirroring devices to eliminate current mirroring errors. The OTA architecture used is shown in Fig 2-4. It comprises of a differential pair and a buffer stage. The capacitor Cc is used for compensation. The compensating capacitor helps in stabilization and settling time of the output. Since, the sampling frequency of the ADC is 100 KHz, the 3dB frequency of the OTA should be above 100 KHz for the circuit to operate properly. Also, the DC value of the OTA output should be Vdd/2.

2.4 Track and Hold Block

Track and Hold (T&H) block is used to sample the input signal and have the equivalent digital output of a particular sample sampled by the T&H block. Fig 2-5 shows the circuit diagram of T&H block.

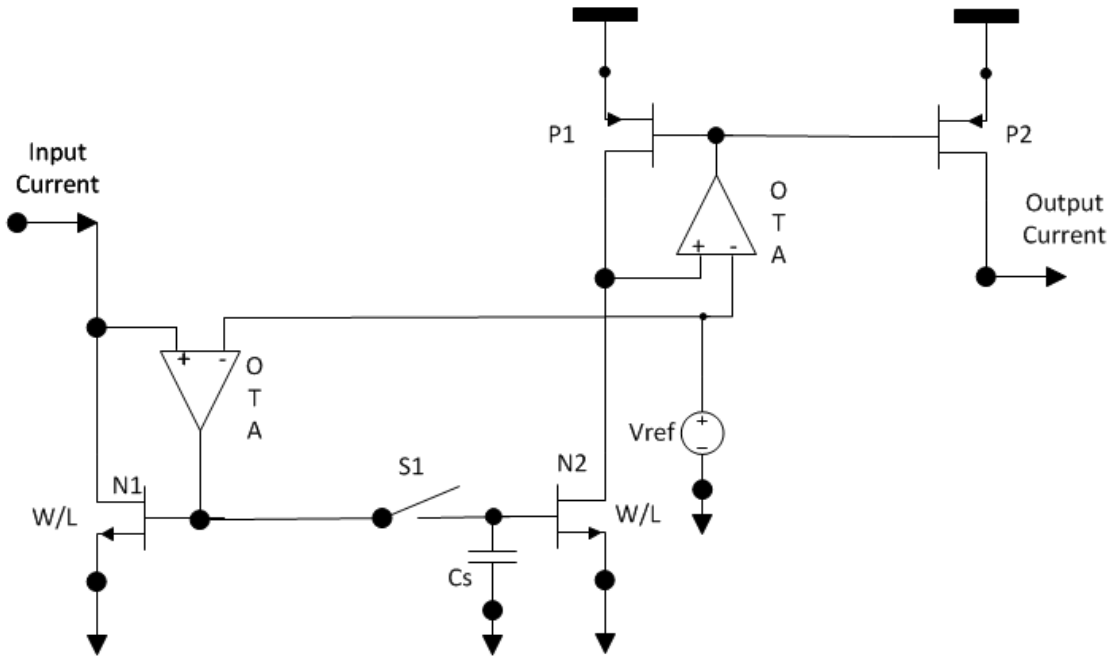

Fig2-5: Track & Hold circuit diagram

Noise filters (capacitors) are used to have clean current signal free from noise at the output. The switch used in T&H block is shown in Fig 2-6.

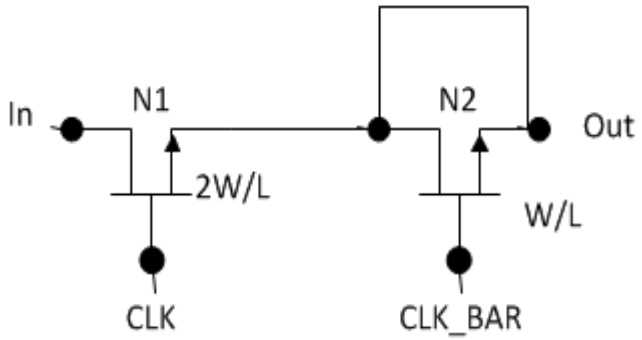

Fig 2-6: Nmos Switch

The operation of T&H block is fairly straight forward. When the CLK is high the output current follows the input current. The sampling capacitor, Cs, gets charged as a function of the input voltage. When the CLK goes low, sampling capacitor does not get more charge from the input. The charge is stored by the capacitor. Ignoring leakages the stored charge remains constant when the CLK is low. Constant charge provides constant voltage at the gate of nmos N2 that result in a constant current. Thus, output current is held constant from the point at which CLK is low. When the CLK goes high again the output current follows the input current. Noise filters (capacitors) are used to have clean current free from noise at the output to improve stability. Capacitors connected between gate of pmos P1 (P2) and ground helps to remove noise. The switch used in T&H block is shown

in Fig 2-6. It comprises two nmos N1 and N2. N2 acts as dummy to eliminate charge sharing and clock feed through.

2.5 Current Comparator

Fig 2-7 shows the current comparator circuit. Twice the input current is compared to a reference current. In the first branch shown in Fig 2-7, interaction between $2I_{in}$ and I_{ref} results in a voltage drop at the input of first inverter (gate of P1 and N1). The size of transistors used to generate $2I_{in}$ and I_{ref} are designed such that If $2I_{in}$ is equal to I_{ref} the voltage set at the input of first inverter will be $V_{dd}/2$.

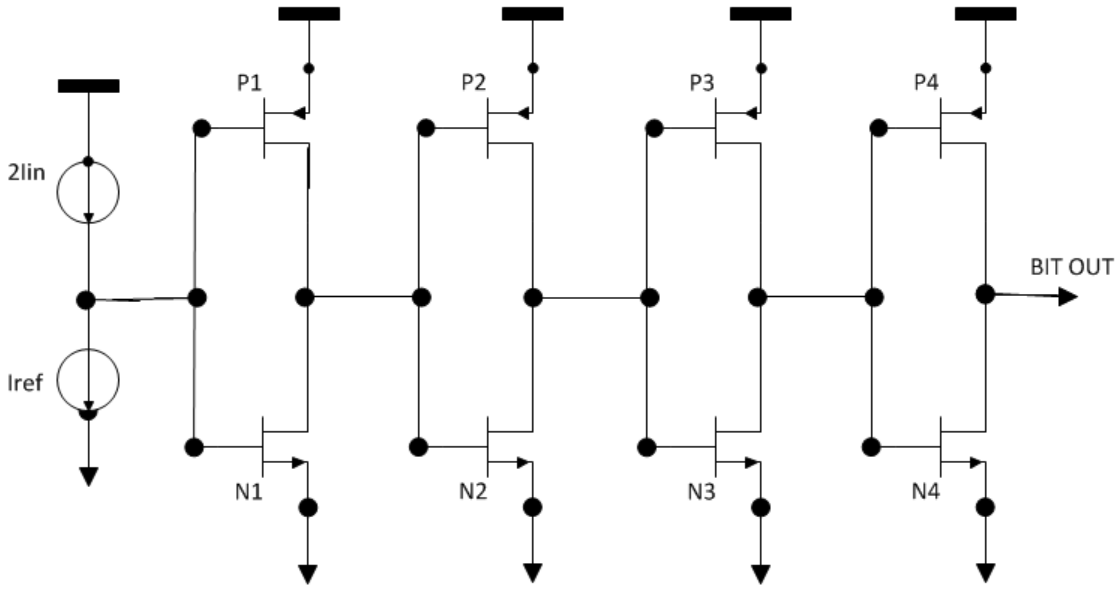

Fig 2-7: Current comparator

From noise point of view the first inverter is slow so that it shows hysteresis. Hysteresis is very important in this circuit because if there is input voltage at the first

inverter, which is close to $V_{dd}/2$ then noise at this point will result in the output going high to low, resulting in incorrect conversion from analog input to a digital level.

The 2nd, 3rd and 4th inverters are to drive the switch. Their sizes are greater than the size of the 1st inverter. The hysteresis shown by these inverters is comparatively less compared to the 1st inverter. The output of the comparator should switch to either high or low quickly. Also, if the input to first inverter is close to $V_{dd}/2$, then the voltage level of the 1st inverter's output may not rise to full V_{dd} resulting in current subtraction error at the switch. Therefore, another advantage of inverter chain is eliminating the current subtraction error by forcing the output to only two values, either V_{dd} or ground.

2.6 D Flip Flop

The D flip flop is used to keep the output stable even when the input is changing

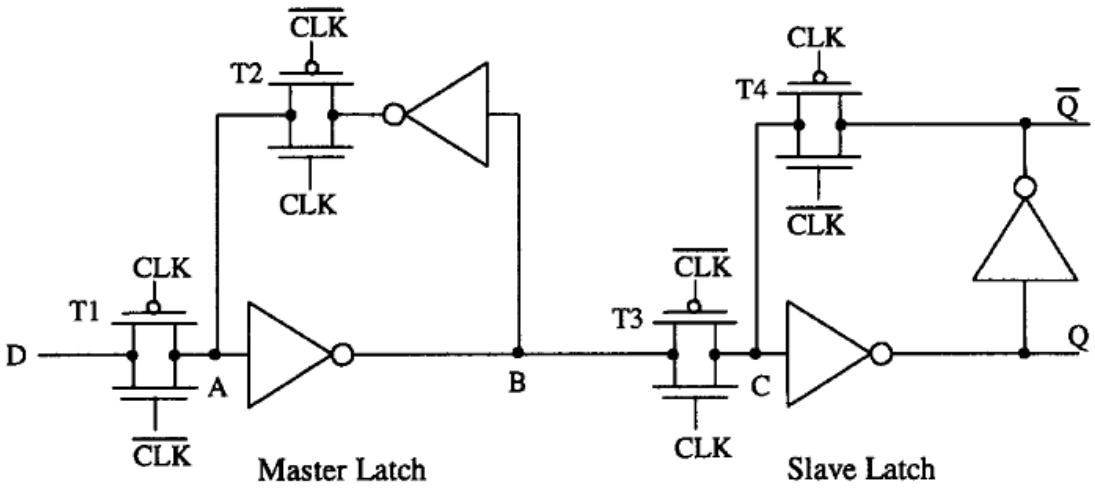

Fig 2-8: D flip flop [6]

during the track or sample phase. Fig. 2-8 shows the D flip flop architecture used [6]. This is simplest flip flop built using inverters and pass gates. The sizes of nmos and pmos used in the inverter are also used for the transistors in pass gate making the design simpler.

2.7 Beta multiplier circuit

To generate bias voltages beta multiplier circuit as shown in Fig. 2-9 was used [7]. The output voltage node requires 10 uF decoupling capacitors, which are off chip to overcome voltage variations due to switching currents levels in the 1 bit cell. External voltage control has significant advantage over internal biasing. The voltage required to generate reference current, if controlled externally, will help in having varied dynamic range of ADC. For e.g. if 6 bit resolution is required the reference current can be decreased resulting in low dynamic range.

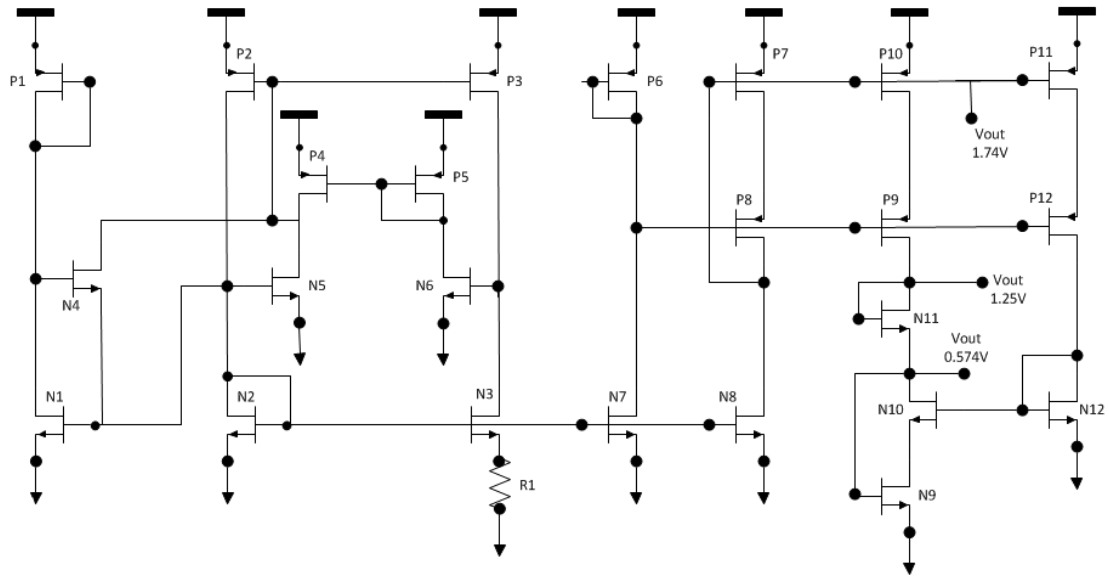

Fig 2-9: Beta multiplier circuit

The output voltages generated by the beta multiplier are 1.25V for the reference voltage to the operational transconductance amplifier, 1.74V bias voltage for the tail transistor in the operational transconductance amplifier and 574mV bias voltage for the nmos in common drain stage used in operational transconductance amplifier.

2.8 Clock generation circuit from 1 MHz reference clock

Since the ADC is integrated with other system level blocks like QIF, DAC and PLL, from system level perspective 1 MHz clock is the reference clock to the whole system. It's from this 1MHz clock a clock for the ADC will be generated. The ADC architecture require 100 KHz clock for track and hold circuit. A delayed clock of 100 KHz for the D flip flops is also required. A digital circuit as shown in Fig 2-10 is used to generate 200 KHz from 1 MHz clock.

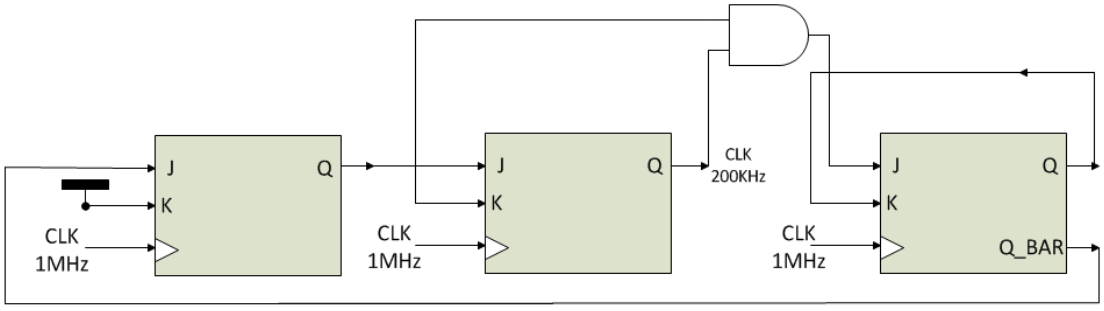

Fig 2-10: 200 KHz clock generator from 1 MHz clock

The 200 KHz clock signal generated from the circuit shown in Fig. 2-10 is an input for a block, which generates clock, clock bar and delayed clock. This circuit is shown in Fig. 2-11.

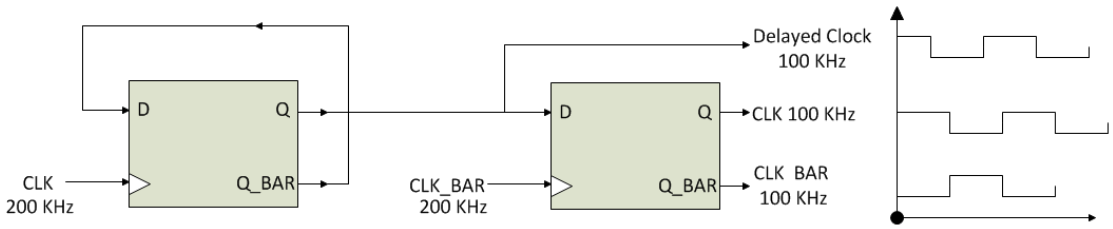

Fig 2-11: Clock, Clock Bar and Delay Clock generator

2.9 Parallel to serial shift register

For the parallel data generated by the ADC it has to be transferred serially to the QIF block. The requirement for the QIF block are such that it requires 8 bits to be transferred in 8 clocks, 9^{th} and 10^{th} clock should always transfer zero data. Also, the data should be in 2's complement form varying from -127 to 128. Latch enable signal should also be there that makes the QIF block know that new data is coming. The latch enable signal should be low during the 10^{th} clock. For first 9 clocks it should be high. The frequency at which this data should be delivered to the QIF block is 1 MHz.

This requirement can be easily implemented by an 8 bit parallel to serial shift register and a mod 10 counter. To generate 2's complement from the 8 bit unsigned ADC output will require complementing the MSB [8]. This will result in 8 bit unsigned values being level shifted from -127 to 128. Unsigned 00000000 corresponds to signed (-) 127 and unsigned 11111111 corresponds to signed (+) 128. An 8 bit unsigned ADC will act as 7 bit signed ADC by simply inverting the MSB.

Fig. 2-12 shows a circuit that generates the latch enable (active low) signal for the QIF block.

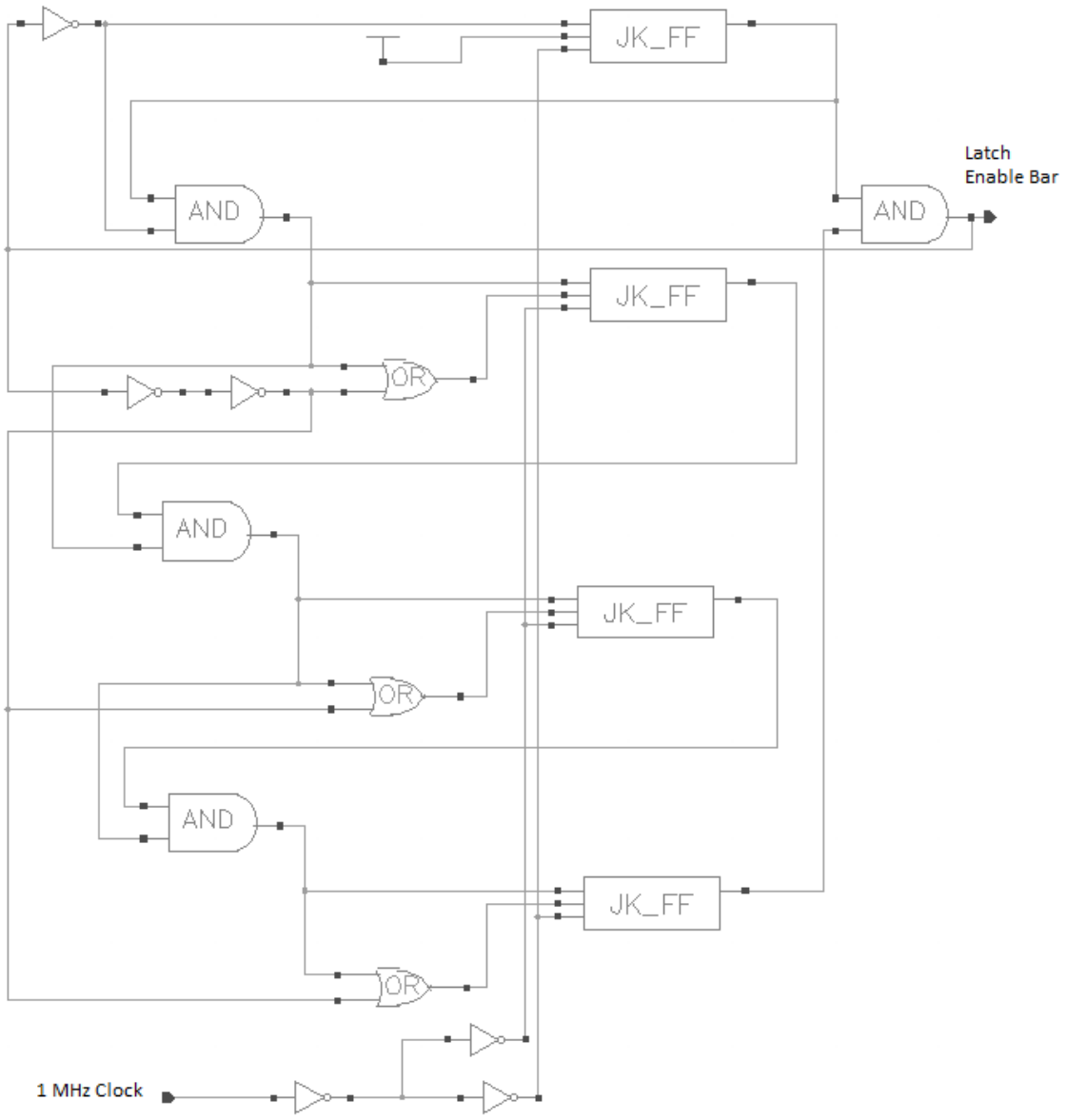

Fig 2-12: Latch Enable signal generator

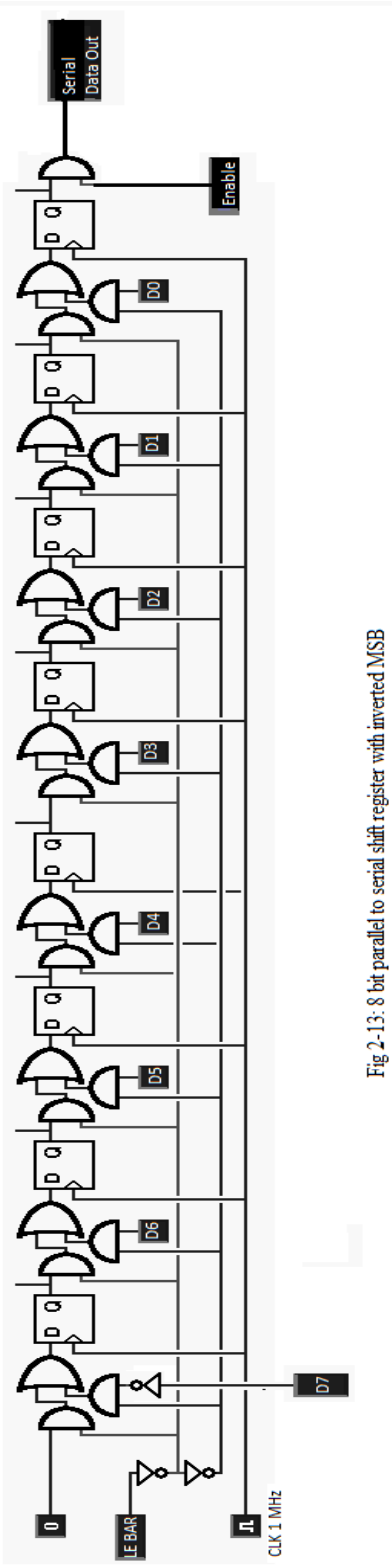

Fig 2-13: 8 bit parallel to serial shift register with inverted MSB

Fig. 2-13 shows the circuit diagram of 8 bit parallel to serial shift register [9]. The MSB is inverted to have 7 bit signed output ranging from -127 to 128. The LE BAR signal in the circuit is generated from the circuit shown in Fig.2-12. There is ENABLE signal at the final AND gate. This is used to have zero output when ENABLE pin is low. The clock frequency to the shift register is 1 MHz. The parallel inputs to this shift register come from the ADC output. As ADC operating frequency is 100 KHz and QIF block require 8 bits to be delivered in 10 cycles therefore the operating frequency of the shift register is 1 MHz that is 10 times the operating frequency of the ADC. Clock generation circuits for the ADC help the shift register and ADC itself to be in synchronization with each other.

2.10 Sizing the transistors used in ADC bit cell.

Sizing of transistors used in 1 bit cell was done by analyzing the Gm, Id, Rds, Vth graphs obtained by nmos and pmos characterization testbench. Fig 2-14 shows testbench used to obtain the above mentioned graphs. Obtaining the graphs is based on spectre simulator. The model file for TSMC 0.25µm process was obtained from MOSIS website [10]. Some settings to be followed from simulation point of view are as follows:

1) In the test bench schematic, the properties of nmos or pmos used should have Model Type option selected to User as shown in Fig. 2-15.
2) Define the path of the model file obtained from MOSIS in the model libraries setup option in Analog Design Environment (ADE).

3) Also define the path of a file, in the model libraries setup option in ADE, with extension scs, say nmos.scs, which contains the line “save N0:oppoint” where N0 is the instance name of the noms used in the schematic.

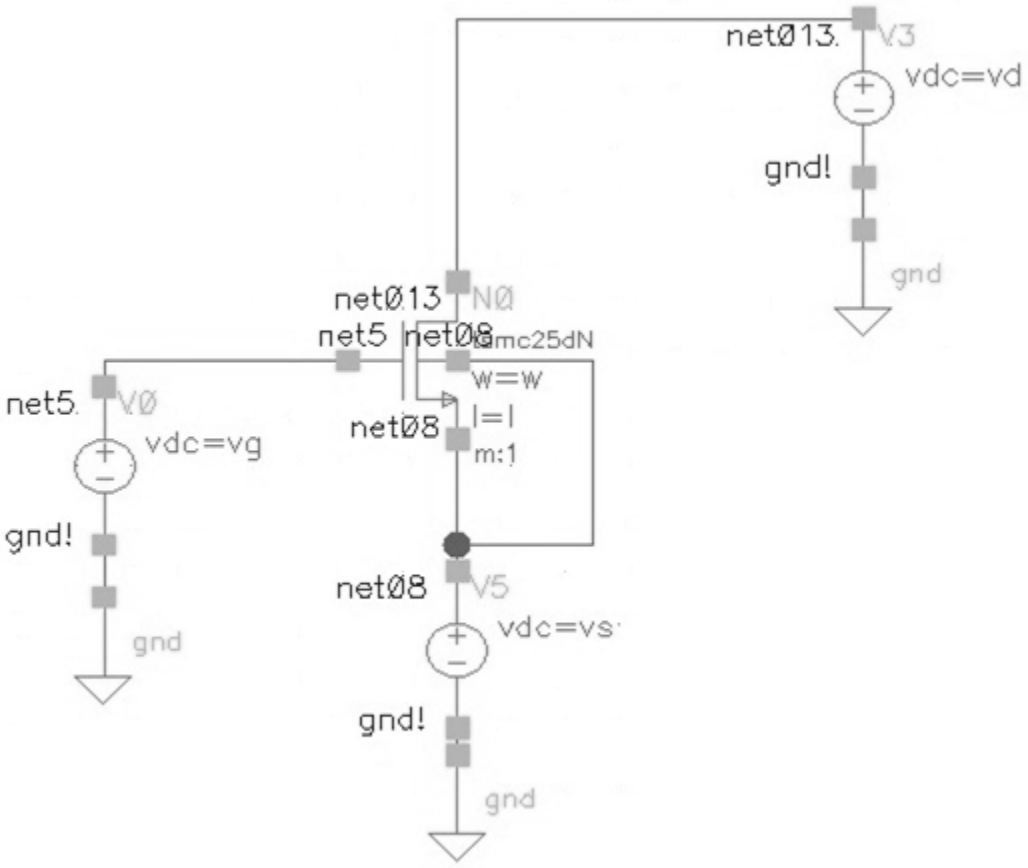

Fig2-14: nmos characterization

Now to obtain a graph of Gm vs any independent variable DC analysis has to be done. For e.g. if Vth variations have to be seen with change in length, then define all the bias voltages and fix the width of nmos. Do the DC analysis by varying length. Now in the

result browser the graph of Vth vs length is already generated. The graph can be obtained by selecting appropriate graph from the list as shown in Fig.2-16. Similarly, other graphs

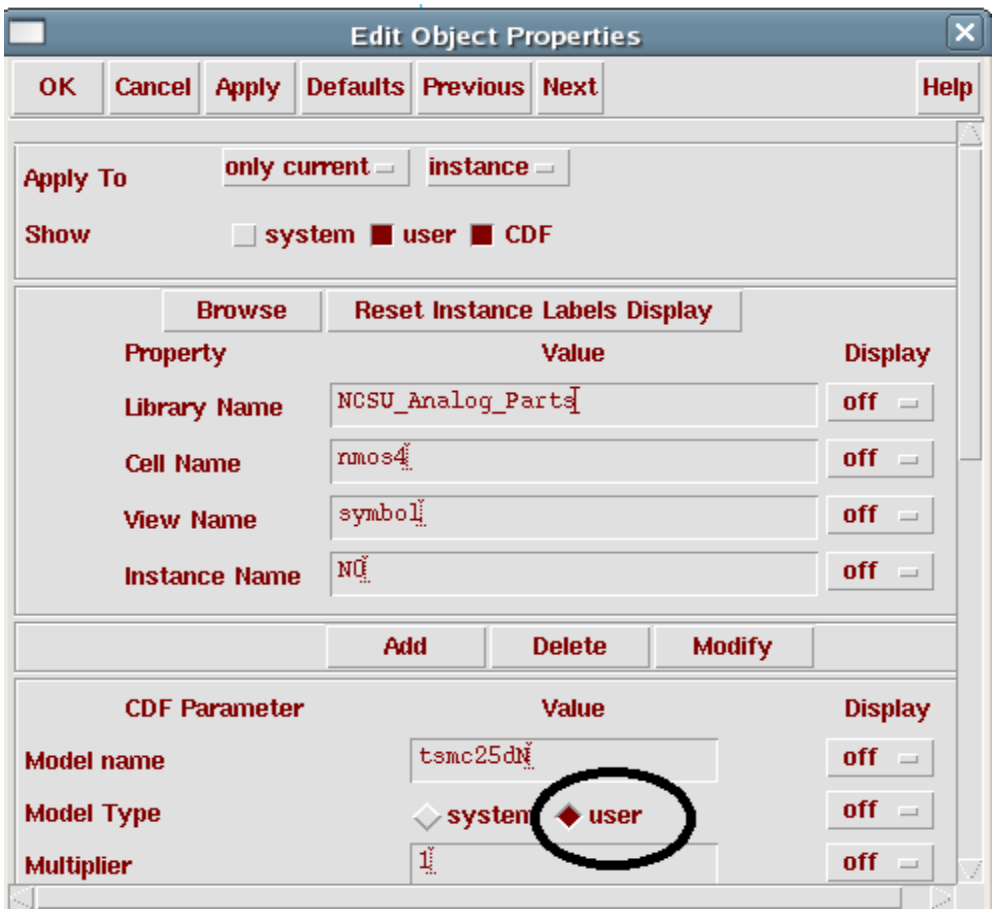

Fig 2-15: Model type selected to user

such as Gm, Id, Rds are generated by setting the testbench as mentioned in the steps.

Now, from design point of view, the ADC bit cell is sensitive to process variations like threshold voltage changes. To make the circuit less sensitive to threshold variations length should be increased but taking in consideration of noise point of view. The current error should be less than 1 LSB for the algorithmic ADC to work properly.

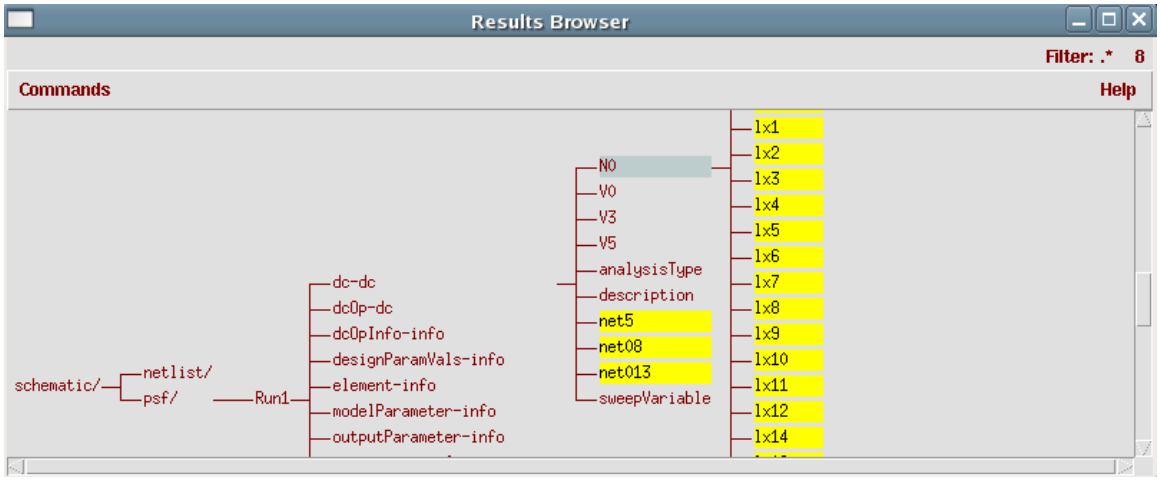

Fig 2-16: Graphs listed in the result browser window of ADE.

Fig 2-17 shows the variation of threshold voltage vs length for nmos. As observed from the graph keeping length of 4µm will give minimum threshold variations even if there is some change in length around that value. Therefore, a length of 4µm was chosen. Similarly, for pmos the length chosen was 2µm.

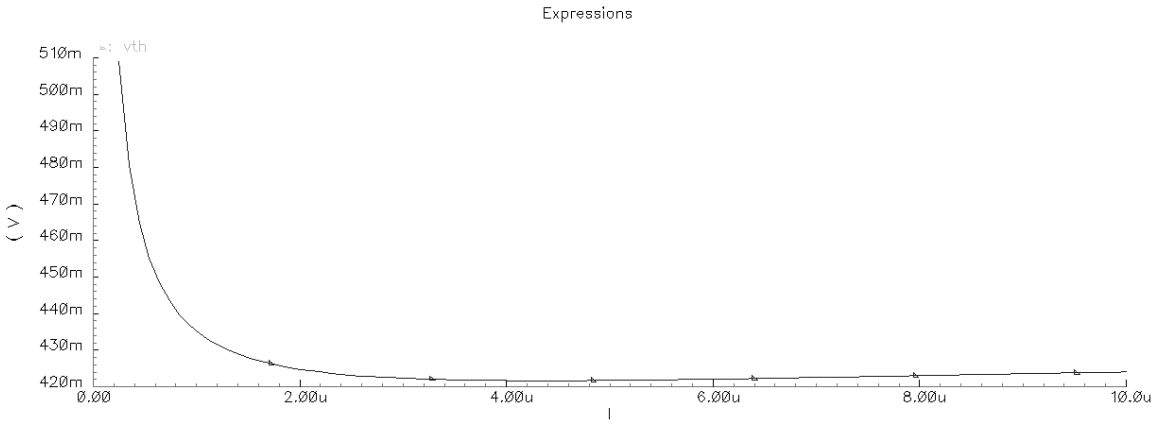

Fig 2-17: Threshold voltage versus length

The bias voltages at gate and drain of nmos and pmos were kept at $V_{dd}/2$. Therefore, width of the transistor was chosen based on the value of reference current. For the ADC designed the reference current chosen is around 100 uA. For the test bench mentioned in Fig 2-14 keeping length 4µm, drain and gate voltages at 1.25 V and source voltage at ground potential DC sweep of width from 1µm to 50µm is carried out. Graph of Id vs width as shown in Fig. 2-18 is plotted. The graph shows at width of 6µm a current of 100uA will flow through the nmos. Similarly, for the same amount of current to flow the size of pmos was obtained from similar graph. The width of pmos comes out to be 18µm.

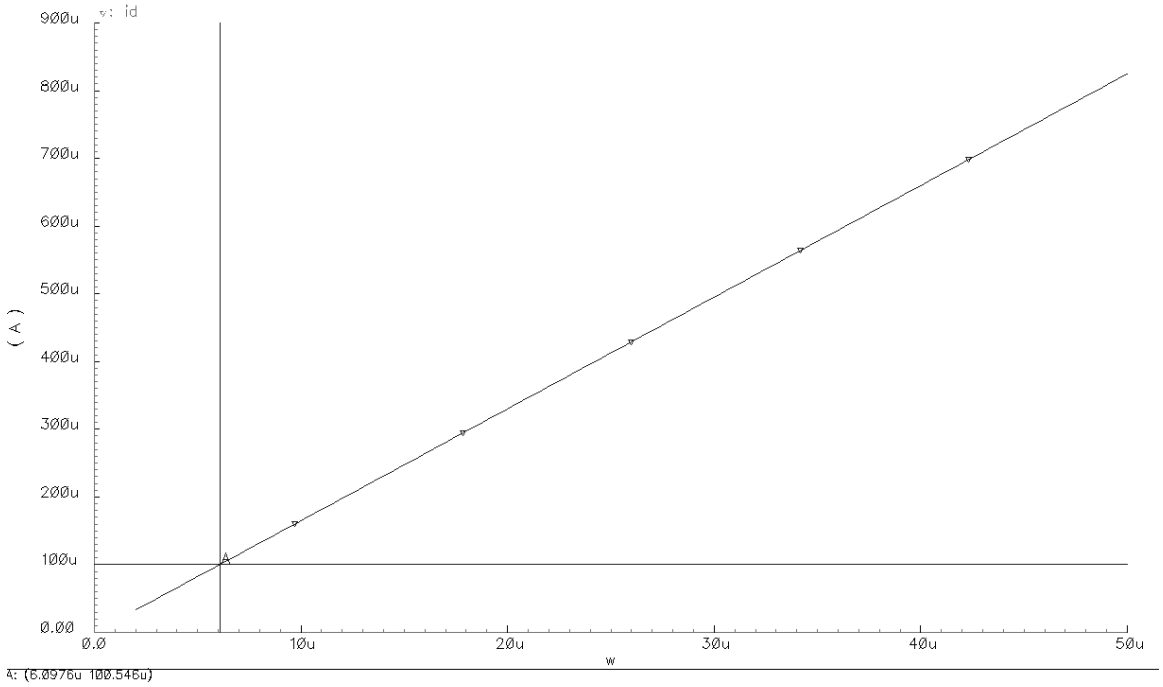

Fig 2-18: Current versus width

For doubling the mirror current, width of nmos and pmos in that branch is doubled. Since, pmos used always handles twice the input current their sizes are doubled i.e.

36µm. Connecting the devices together after sizing them will result in current value similar or close to as expected when the transistor was placed alone in the test bench shown in Fig. 2-14. Minor tweaking may be required after that. After that the noise analysis of the complete bit cell is carried out to make sure the size chosen will work and does not inject significant noise.

If 100uA is the full scale current, 1 LSB comes out to be ~390 nA current. The minimum detected signal by the circuit should be 4 times smaller than 1 LSB (~ 98 nA) [3]. Input referred noise is given by

Sqrt((minimum signal^2) / B.W.) = Sqrt((98 e-9)^2/100e3) = 309.9 pA/$\sqrt{Hz}$ (2.1)

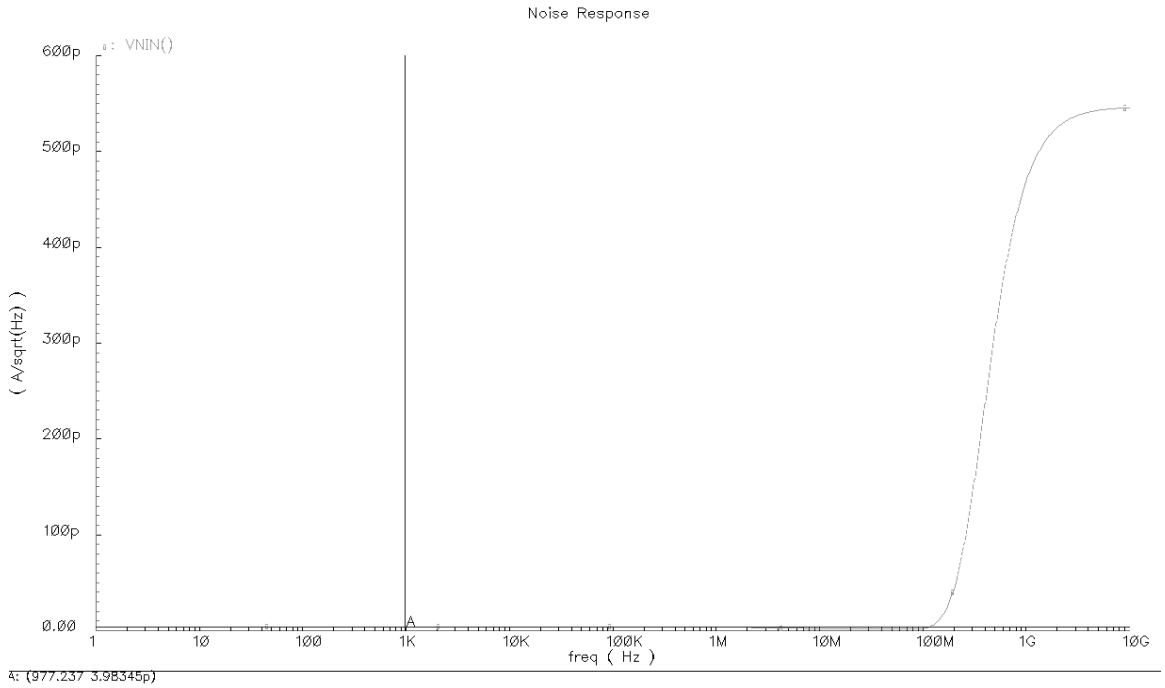

Fig 2-19: Input referred noise of bit cell

The simulated input referred noise of 1 bit cell is as shown in Fig. 2-19. Comparing the simulated value with the one obtained in equation 2.1, the simulated value is less, which confirms that the sizes chosen for the bit cell are right as they result in not so significant noise generation.

2.11 Transistor sizes

Transistor sizes used for each circuit block are as shown below. Table 2-1 shows the transistor sizes used in 1 bit cell circuit shown Fig 2-2.

NMOS			**PMOS**		
Transistor	**Width**	**Length**	**Transistor**	**Width**	**Length**
N1	5.88µm	4.02 µm	P1	35.88µm	1.98µm
N2	11.76µm	4.02 µm	P2	35.88µm	1.98µm
N3	5.88µm	4.02 µm	P3	35.88µm	1.98µm
N4	5.88µm	4.02 µm			
N5	60µm	240nm			

TABLE 2-1: Transistor sizes used in 1 bit cell.

Table 2-2 shows the transistor sizes used in operational transconductance amplifier circuit shown in Fig 2-4.

Table 2-3 shows the transistor sizes used in track and hold circuit shown in Fig 2-5.

Table 2-4 shows the transistor sizes used in NMOS switch shown in Fig 2-6 used in track and hold circuit

NMOS			**PMOS**		
Transistor	**Width**	**Length**	**Transistor**	**Width**	**Length**
N1	600 nm	5.52µm	P1	4.98µm	780 nm
N2	600 nm	5.52 µm	P2	4.98µm	780 nm
N3	6 µm	1.98 µm	P3	7.98µm	780 nm
N4	6 µm	1.98 µm			
Compensation capacitor Cc = 500 fF					

TABLE 2-2: Transistor sizes used in OTA

NMOS			**PMOS**		
Transistor	**Width**	**Length**	**Transistor**	**Width**	**Length**
N1	5.88µm	4.02 µm	P1	17.94µm	1.98µm
N2	5.88µm	4.02 µm	P2	17.94µm	1.98µm
Sampling Capacitor Cs = 1 pF					

TABLE 2-3: Transistor sizes used in Track and Hold circuit

NMOS		
Transistor	**Width**	**Length**
N1	3 µm	240 nm
N2	1.5 µm	240 nm

TABLE 2-4: Transistor sizes used in NMOS switch

NMOS			PMOS		
Transistor	**Width**	**Length**	**Transistor**	**Width**	**Length**
N1	360 nm	240 nm	P1	1.08 µm	240 nm
N2	1.92 µm	240 nm	P2	4.32 µm	240 nm
N3	5.04 µm	240 nm	P3	10.08 µm	240 nm
N4	5.04 µm	240 nm	P4	10.08 µm	240 nm

TABLE 2-5: Transistor size used in current comparator

Table 2-5 shows the size of current comparator circuit as shown in Fig 2-7.

The nmos transistors used in the D-FF has width of 5.04 µm and pmos transistors used has width of 10.08 µm. Length of all the pmos and nmos is 240 nm. The AND and OR gates used are from standard cell library of TSMC 0.25 µm from OSU [16].

NMOS			PMOS		
Transistor	**Width**	**Length**	**Transistor**	**Width**	**Length**
N1	19.98 µm	4.02 µm	P1	10.02 µm	100.02 µm
N2	19.98 µm	4.02 µm	P2	30 µm	1.98 µm
N3	40.02 µm	4.02 µm	P3	30 µm	1.98 µm
N4	19.98 µm	1.98 µm	P4	30 µm	1.98 µm
N5	19.98 µm	4.02 µm	P5	30 µm	1.98 µm
N6	19.98 µm	4.02 µm	P6	15 µm	10.08 µm
N7	19.98 µm	4.02 µm	P7	30 µm	1.98 µm

N8	19.98 µm	4.02 µm	P8	30 µm	1.98 µm
N9	19.98 µm	4.02 µm	P9	30 µm	1.98 µm
N10	19.98 µm	4.02 µm	P10	30 µm	1.98 µm
N11	19.98 µm	4.02 µm	P11	30 µm	1.98 µm
N12	15 µm	16.02 µm	P12	30 µm	1.98 µm
Resistor R1 = 3.7 K ohm					

TABLE 2-6: Transistor sizes used in beta multiplier

CHAPTER 3

PHYSICAL IMPLEMENTATION ADC

3.1 Layout of ADC

1 bit cell layout is shown in Fig. 3-1. Inter digitated technique is used in the layout to match the mirroring currents with least amount of error due to process variations. To

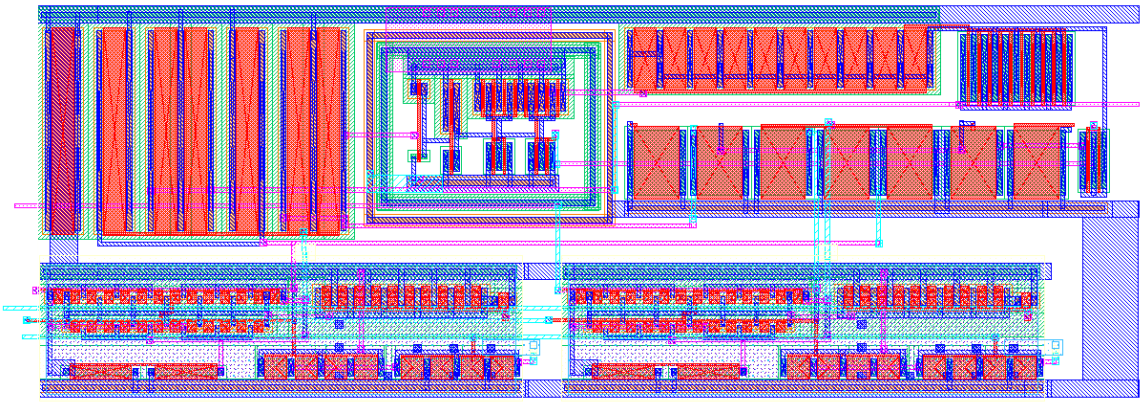

Fig 3-1: 1 bit cell layout

improve matching dummy transistors were placed adjacent to main transistors. As the comparator is a sensitive part of every bit cell, the transistors for comparator circuit were placed inside ntap and ptap guard rings. The OTA transistors were also laid out using inter digitated technique. The compensation capacitor used in the OTA was laid out over the transistors itself as it is MIM (Metal Insulator Metal) cap resulting in low area consumption. Fig 3-2 shows the track and hold layout. Fig 3-3 shows the D flip flop circuit used in the ADC. The cell height used for the layout of digital circuits is 12 µm.

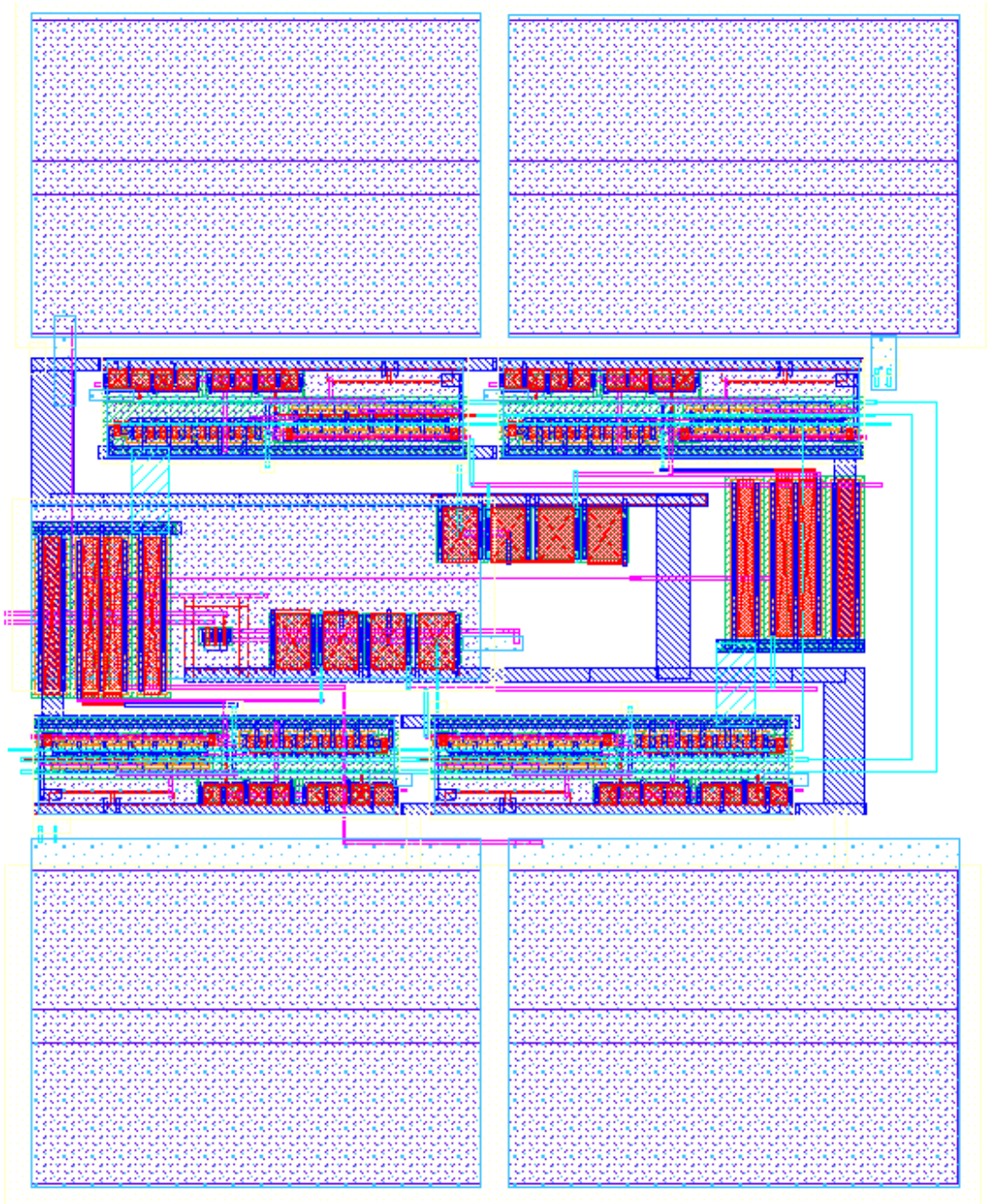

Fig 3-2: Track and hold circuit layout

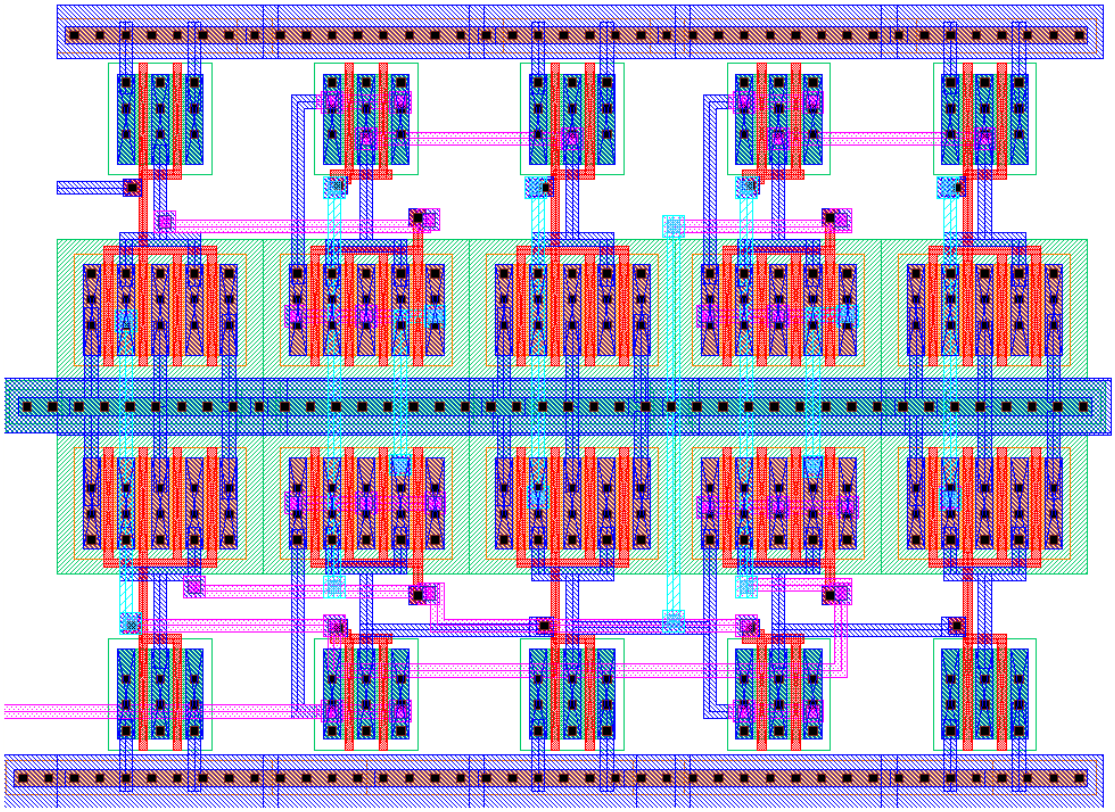

Fig 3-3: D flip flop layout

Complete layout of ADC is as shown in Fig 3-4. The digital and analog blocks were

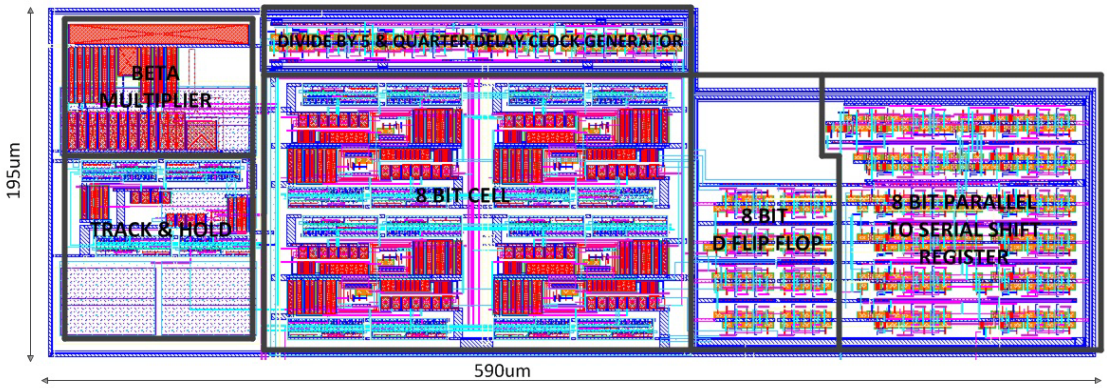

Fig 3-4: Complete layout of ADC

shielded by majority carrier guard rings. Table 3-1 shows area occupied by each analog and digital blocks used in the ADC.

Block	**Area**
8 One bit cells	145µm X 216µm
Track & Hold	117µm X 148µm
Divide by 5 and clock delay generator	22.5 µm X 233 µm
Beta multiplier	72µm X 104µm
8 Bit D Flip-Flop	75µm X 88µm
8 Bit parallel to serial shift register	133µm X 144µm

CHAPTER 4

POST LAYOUT SIMULATION RESULTS

Through characterization of ADC is required to make sure the circuit will work as expected after it is fabricated. ADCs are characterized using the following tests, which are basic to any ADC:

1) DNL
2) INL
3) SNDR
4) SFDR

4.1 Differential Nonlinearity (DNL)

Differential non-linearity of an ADC is defined as the difference between actual code width of real ADC and the Ideal ADC [11]. Fig 4-1 shows an example of ADC output with DNL graph. For each stair case DNL can be calculated by the formula:

$$\text{DNL}=\text{Actual step width} - \text{Ideal step width} \qquad (4.1)$$

As seen in Fig 4-1, 6^{th} (region between 5/8 and 6/8) transition is shooting up, the DNL is -1LSB at that point. If the transition height was greater than the one shown in 6^{th} transition then the DNL will be less than -1LSB resulting in missing code. The 2^{nd} transition makes the DNL go to +1LSB. Irrespective of the width of the staircase it cannot be guaranteed if there will be missing code or not because the transition did not happen and it cannot be predicted if transition happens it will jump by 1LSB or 2LSB.

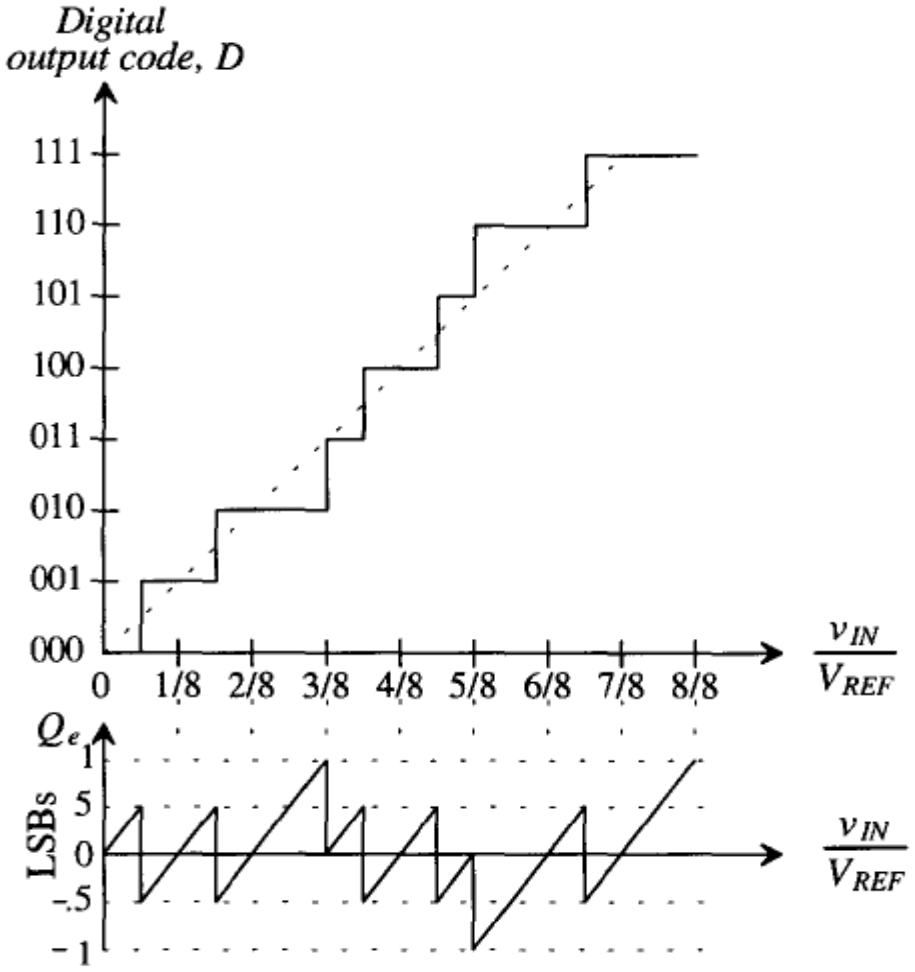

Fig4-1: Transfer curve for real ADC and quantization graph explaining DNL [11]

4.2 Integral Nonlinearity (INL)

INL is calculated from the transfer curve of ADC by drawing a straight line, which touches the first and last code transition. INL is then given by difference between ADC code transitions and the straight line. The INL graph is as shown in Fig 4-2.

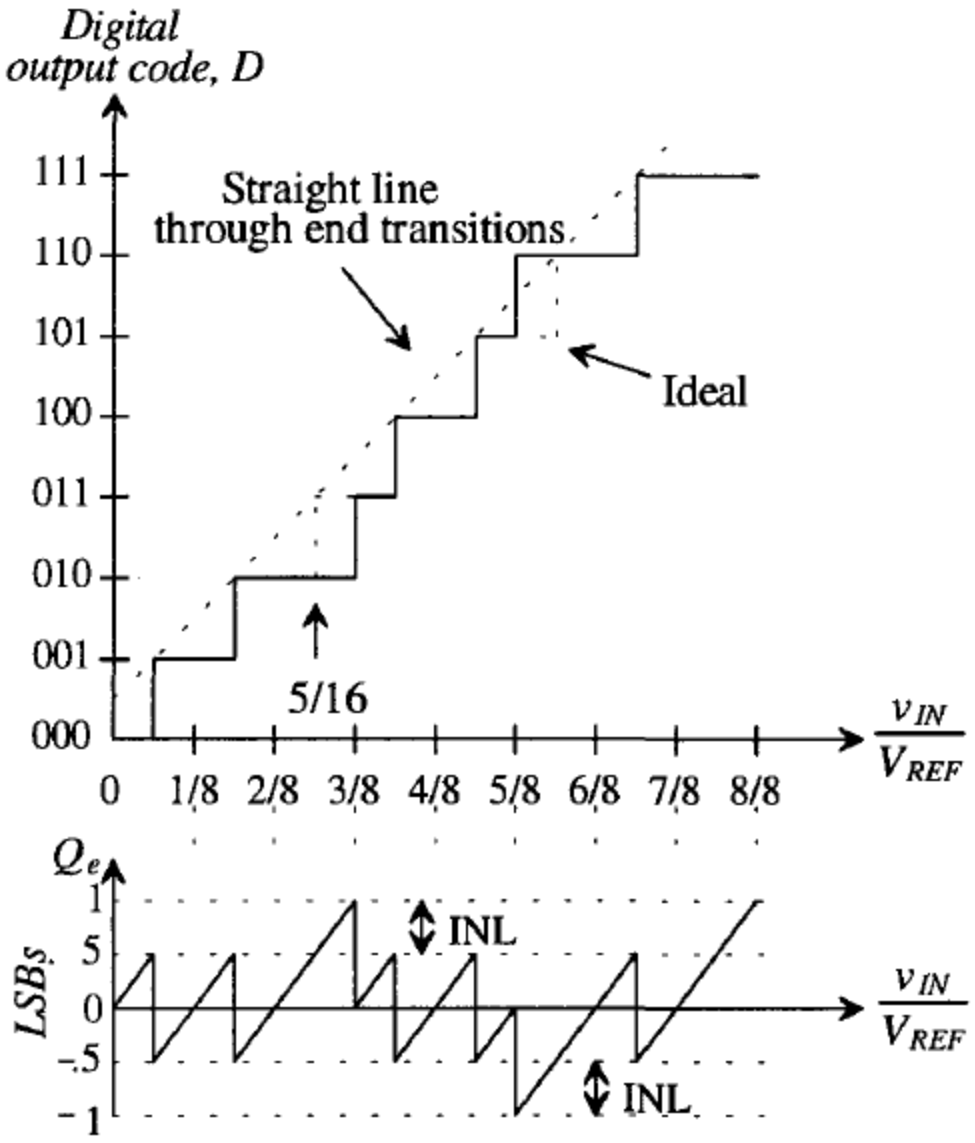

Fig 4-2: Transfer curve for real ADC and quantization graph explaining INL [11]

4.3 Test Bench for INL and DNL

Both the INL and DNL graphs can be obtained with single testbench as shown in Fig. 4-3 [12]. Since, the resolution of designed ADC is 8 bit. Thus, there are 256 digital levels. For this test bench a ramp input is given to designed ADC such that the input is

zero at zero time and reaches its full scale value at time equal to sampling period times number of digital levels. The input should reach its full scale value at 2.56ms (256 X 10us). The output of real ADC should go to ideal DAC of the same resolution.

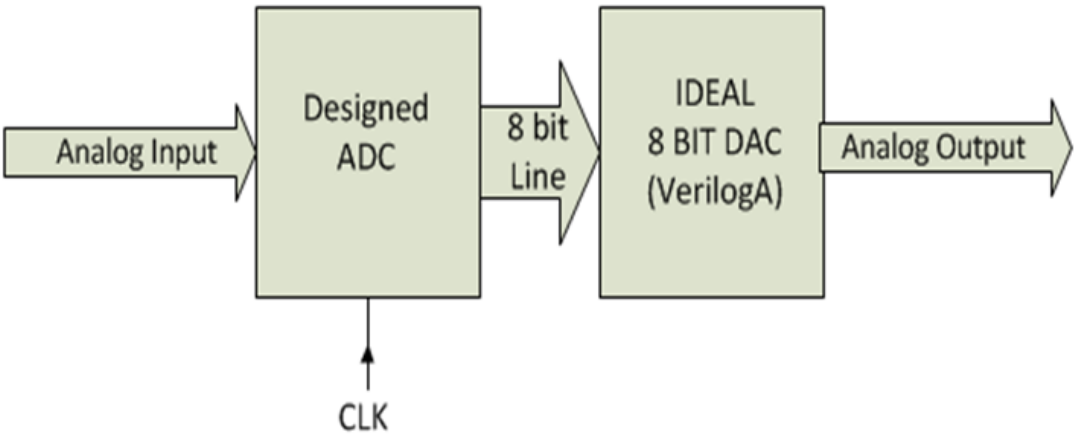

Fig4-3: Test bench to find INL and DNL

The input of ADC as a ramp and output of DAC as staircase

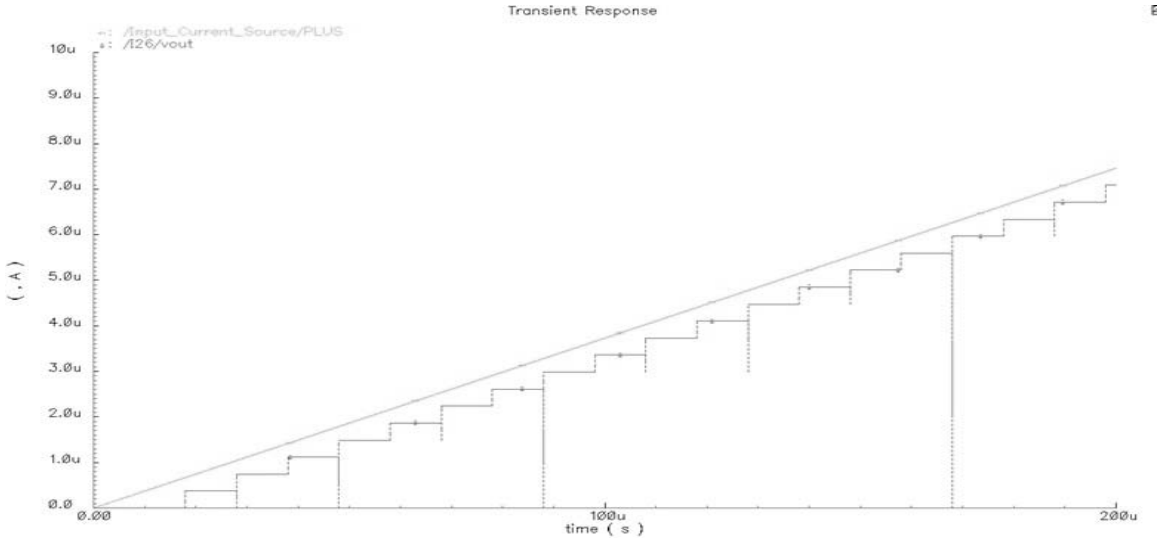

Fig 4-4: Ramp input to real ADC and staircase output of ideal DAC.

can be seen as shown in Fig 4-4. The ideal DAC with 8 bit resolution is designed using verilogA. The verilogA code for the ideal DAC can be seen in Appendix I. Subtraction of output from input will give INL and DNL graph as shown in Fig 4-5.

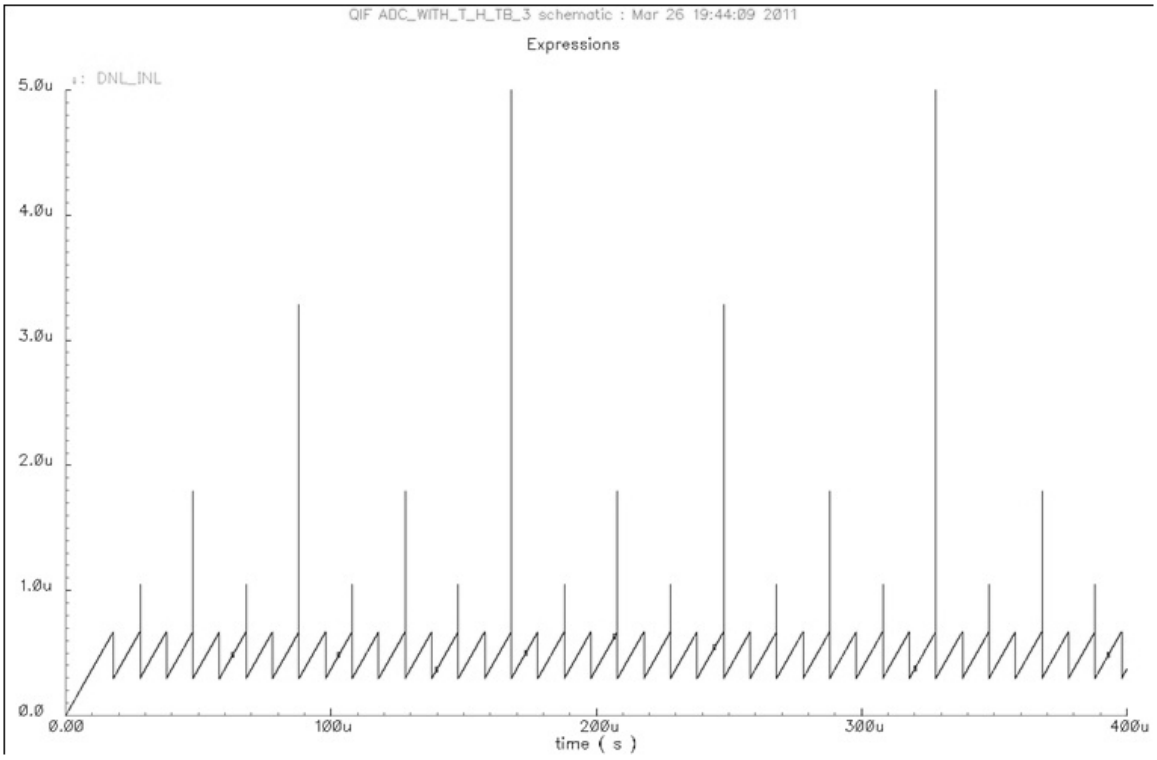

Fig 4-5: INL and DNL of the designed ADC.

The spikes in the graph are due to the designed DAC. It is designed in such a way that even if there is minute change in the output of ADC it will be reflected at the output of DAC. The parallel output generated comes from D flip flop and all the bits cannot be generated at the same time due to clock skew and jitter, which results in spikes. The duration of these spikes is around 60ps. The DC shift of the graph in Fig 4-5 is due to latency of the ADC. Fig 4-6 shows complete time scale of 2.56ms with no INL or DNL

error.

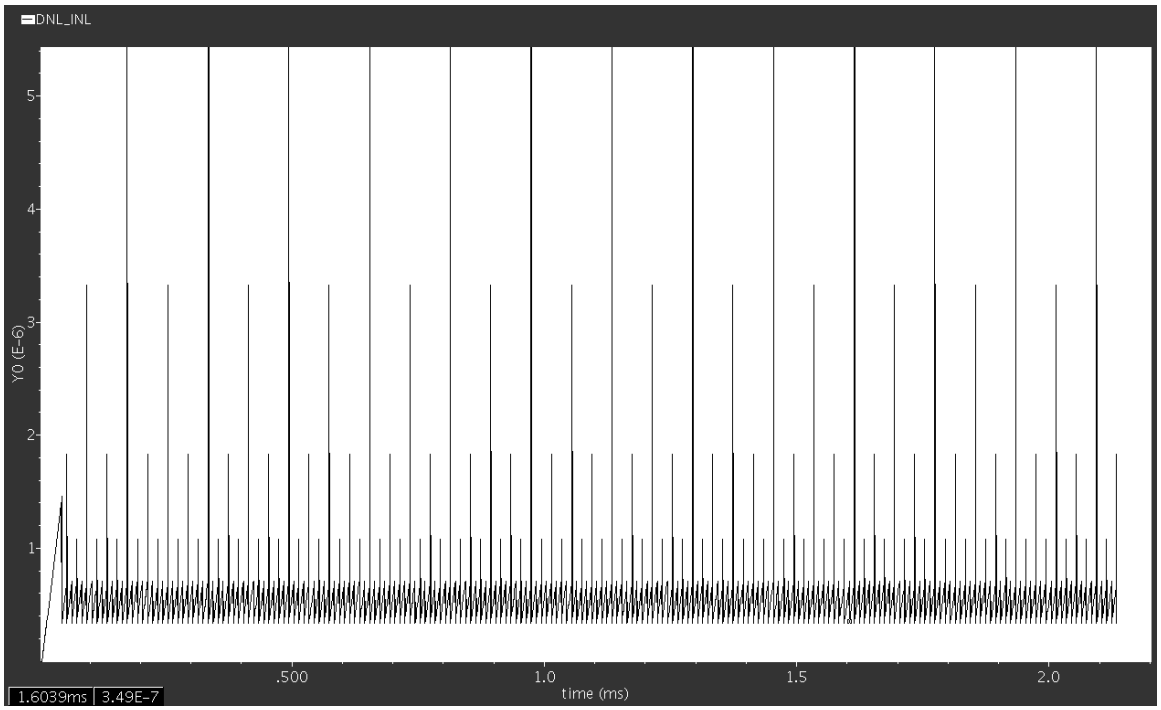

Fig 4-6: INL and DNL complete time scale.

4.4 Test Bench for SNDR

For current mode ADC the signal to noise ratio is given by the formula

$$SNR_{ideal} = 20.\ Log\ (I_P/\sqrt{2}\ )/(I_{LSB}/\sqrt{12}\ ) \qquad (4.2)$$

Where, I_p is full scale current. SNR_{real} is given by

$$SNR_{real} = (0.5(I_{REF+} - I_{REF-})/\sqrt{2}\ )/\ I_{Qe,RMS} \qquad (4.3)$$

Where, $I_{Qe,RMS}$ is the quantization error current and is calculated by the test bench as shown in Fig 4-7. A sinusoidal input is given to the designed ADC. The output of

designed ADC goes into ideal verilogA DAC. The same sinusoidal input signal goes into an ideal Sample and Hold circuit. There is delay added in the input signal to the ideal S&H (verilogA model in Appendix II) and its clocked in such a way that output of S&H block and output of Ideal DAC changes at the same time. In this way the fundamental frequency component is eliminated during the subtraction of S&H output from the DAC output. The resulting signal is the quantization error current [14]. Taking rms of this signal will give $I_{Qe,RMS.}$

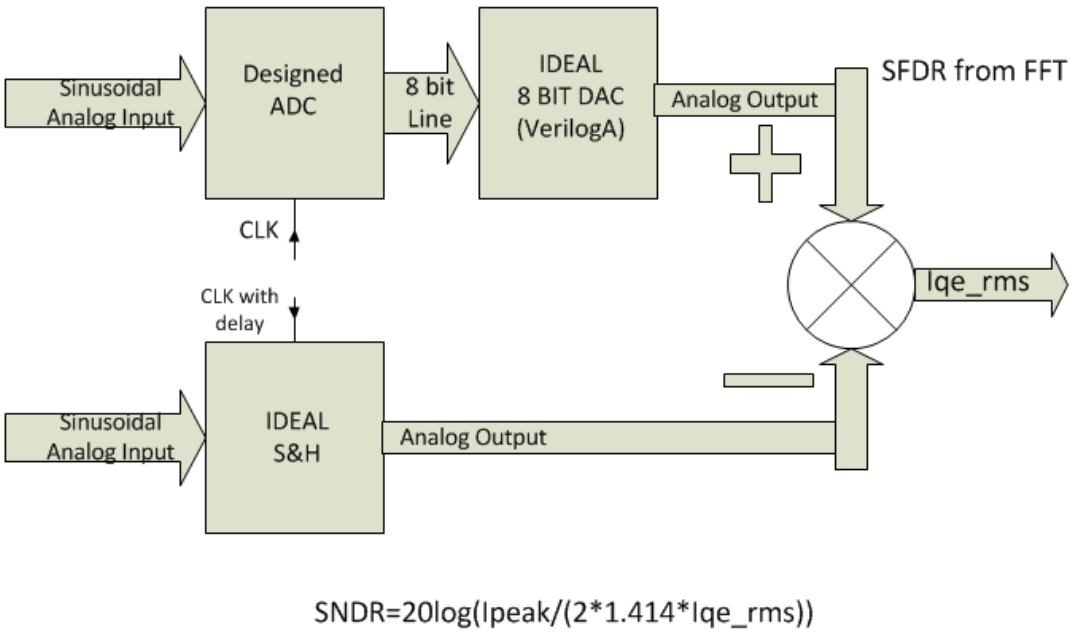

Fig 4-7: SFDR and Quantization error current test bench.

In real ADC there is always distortion in the output because of non-linearity and mismatch in the data converter circuitry [13]. Therefore, the quantization error calculated also contains distortion. As this error current is used to find SNR shown in equation 4.3, the SNR is actually called as SNDR (signal to noise plus distortion ratio). Fig 4-8 shows

the output when 10 KHz sinusoidal is given at the input of the test bench shown in Fig 4-7. $I_{Qe,RMS}$ is given by subtracting 3rd wave from the 2nd and taking rms of the resulting wave. At 10 KHz the $I_{Qe,RMS}$ is 308.9 nA.

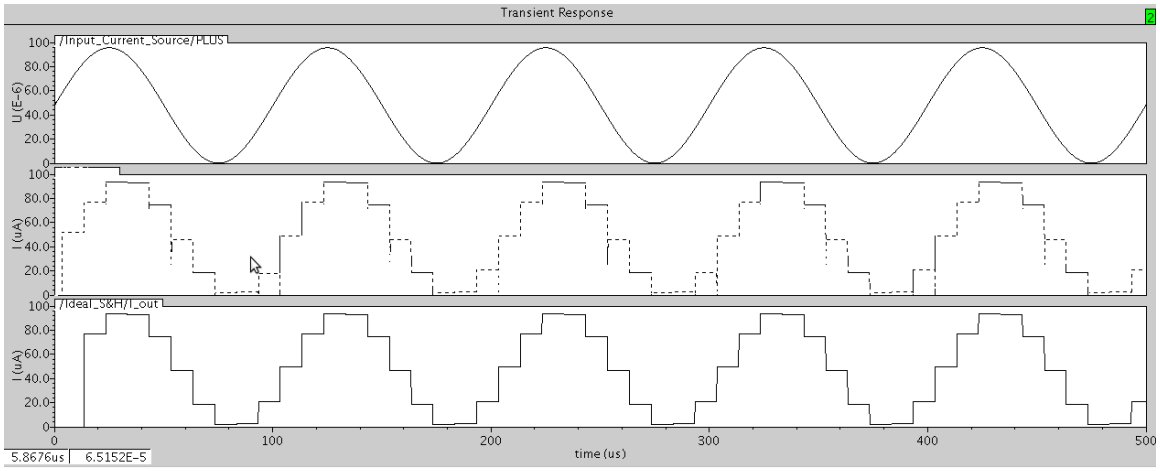

Fig 4-8: Output of DAC and S&H with 10 KHz signal. a) input signal b) output of ideal DAC c) output of ideal S&H signal

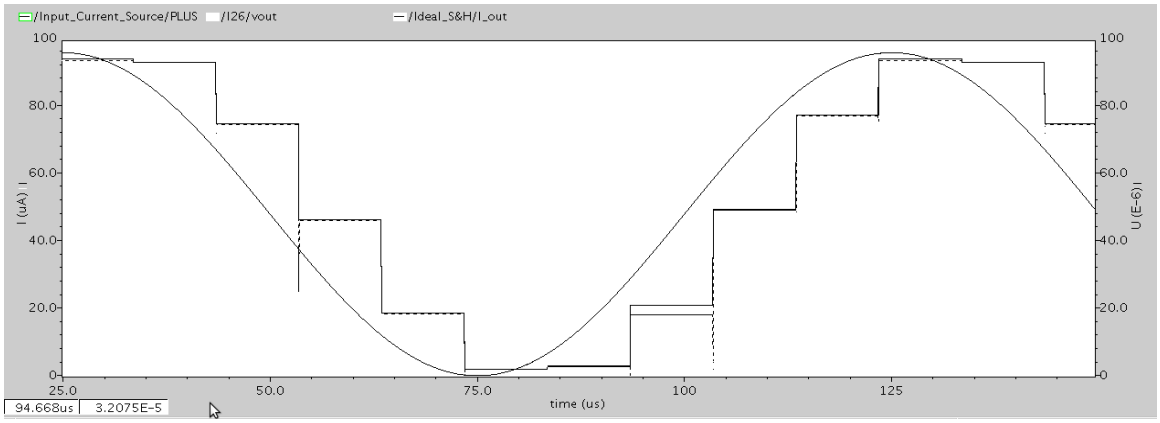

Fig 4-9: DAC output and S&H output overlapped at 10 KHz. Solid lines S&H output, dotted lines DAC output and continuous sine wave is the input signal

Using this value in equation 4.3 and taking I_{REF+} equal to 96uA and I_{REF-} equal to 0 the SNDR for the designed ADC at 10 KHz comes out to be 40.81dB. Fig 7-9 shows both DAC output and S&H output when they are overlapped on each other. It can be seen that there is difference between S&H output and DAC output and the difference is called as quantization error.

4.5 Test Bench for SFDR

Spurious free dynamic range (SFDR) is calculated from the same test bench shown in Fig 4-7. Taking DFT of the output of DAC will give an output spectrum of ADC. SFDR is defined as the difference of signal level between the fundamental frequency and the largest tone in the output spectrum up to the Nyquist frequency. The DFT of output signal of DAC was calculated using cadence calculator, which requires special simulation settings [15]. Fig 4-10

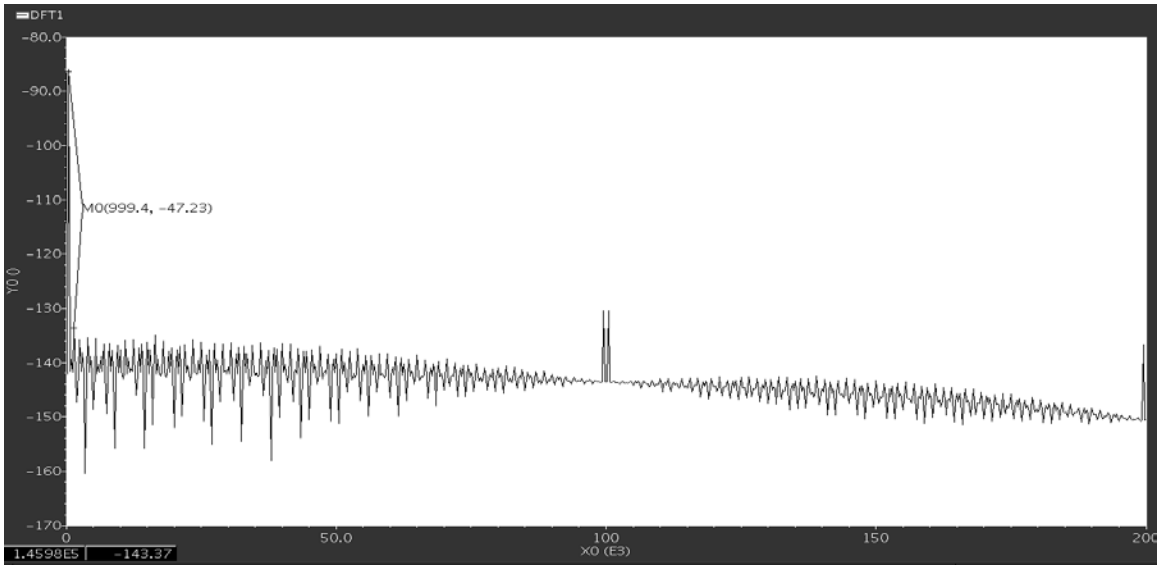

Fig 4-10: 47.23 dB SFDR at 500 Hz

Table 4-1 shows the SNDR and SFDR at different frequencies. These values were obtained by giving different frequency signal to the testbench in Fig 4-7. The SNDR is low at low frequencies because the quantization error increases. The reason for quantization error getting increased is the sampled signal does not rise much and what happens is for two continuous samples (low rising signal) the digital output only changes once. As the signal level remains in a particular LSB range the digital output will be the same for that particular LSB range.

Frequency (Hz)	SNDR (dB)	SFDR(dB)	ENOB
100	29.57	51.71	4.61
500	36.24	47.23	5.73
1K	36.17	47.07	5.71
5K	40.27	51.1	6.39
10K	40.81	40.3	6.48
20K	39.23	38.24	6.22

TABLE 4-1: SNDR, SFDR and ENOB at different frequencies

4.6 Power consumption

The rms current drawn from voltage source of 2.5V is 3.47 mA when a sinusoidal signal of 1 KHz was applied at the input. Therefore, the power consumed is 8.675 mW.

4.7 Summary

Designed ADC performance specifications are shown in Table 4-2

Process	TSMC 0.25 µm
Power supply	2.5 V
Reference Current	95.57 uA
Sampling Rate	100 KHz
SNDR	40.81 dB (max), 29.57 dB (min)
SFDR	51.71 dB (max), 38.24 dB (min)
ENOB	6.48 (max), 4.61(min)
Power dissipation	8.675mW
Input referred noise	309.9 pA/$\sqrt{Hz}$
Area	195µm X 590µm (with serializer) 195µm X 454µm (without serializer)

TABLE 4-2: ADC performance summary

4.8 Comparison among other ADCs implemented

Table 4-3 shows the comparison between different ADCs implemented.

Type of ADC	**Tech.** (µm)	**Bits**	**Area**	**Power**	**Sampling Rate (Samples/sec)**	**ENOB**
Dual Slope [23]	0.25	8	710 µm X 630 µm	5 mW	2M to 45.32 M	-
Flash [24]	0.09	8	1.1 mm X 0.6mm	207 mW	1.25 G	7
SAR [21]	0.25	8	150 µm X 230 µm	680 nW	20 K	6.65
SAR [22]	0.18	8	250 µm X 320 µm	7.75 µW	500 K	7.5
Current Mode [3]	0.18	6	7479 μm^2	6 µW	130 K	5.5
This Work	0.25	8	195 µm X 454 µm	8.67 mW	100 K	6.48

Table 4-3: Comparison between different ADCs

CHAPTER 5

CONCLUSION

5.1 Future Work

The performance specifications of the designed ADC were achieved as expected. The ADC specifications can be improved by reducing power. The ADC architecture can be made more immune to noise with minor modifications. To reduce power consumption, sub threshold design can lead to low power of the order of micro watts.

For better noise performance, the output of current comparator can be connected to D flip flop and the output of D flip flop should go to nmos switch as shown in Fig 5-1. By this the nmos is either ON or OFF thus ensuring proper subtraction operation of reference current.

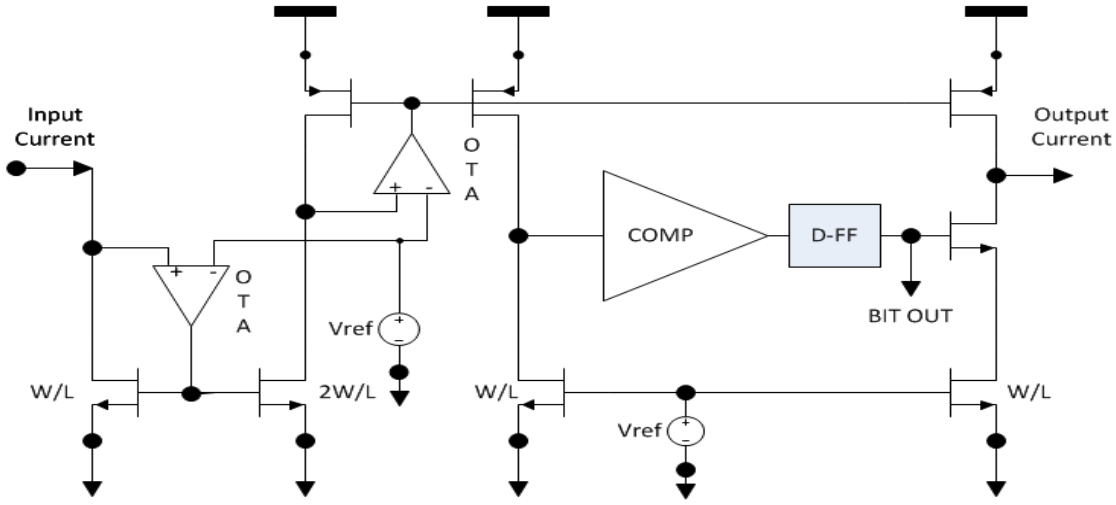

Fig 5-1: Bit cell improvement.

5.2 Conclusion

Current mode algorithmic ADC with 8 bit resolution was successfully implemented using TSMC 0.25 µm technology. Post layout simulation was carried out to characterize the ADC. As expected the current mode architecture occupies less area. Design simplicity and ease of implementation is added advantage of this architecture. The architecture does not require precision capacitors or resistors, which vary by process variations. Implementation of parallel to serial architecture was also done, which makes the ADC output available to QIF block.

REFERENCES

[1] 12. Nairn, D.G.; Salama, C.A.T.; , "Current-mode algorithmic analog-to-digital converters," *Solid-State Circuits, IEEE Journal of* , vol.25, no.4, pp.997-1004, Aug 1990.

[2] Najafi, K.; Wise, K.; , "Implantable multielectrode array with on-chip signal processing," *Solid-State Circuits Conference. Digest of Technical Papers. 1986 IEEE International* , vol.XXIX, no., pp. 98- 99, Feb 1986.

[3] Agarwal, A.; Kim, Y.B.; Sonkusale, S.; , "Low power current mode ADC for CMOS sensor IC," *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on* , vol., no., pp. 584- 587 Vol. 1, 23-26 May 2005 doi: 10.1109/ISCAS.2005.1464655

[4] R. Jacob Baker. 2007. *CMOS: Circuit Design, Layout, and Simulation (IEEE Press Series on Microelectronic Systems)*. Revised 2nd edition. Wiley-IEEE Press. Chapter 20, Section 20.2.

[5] A.S. Sedra, K.C. Smith, “Microelectronic Circuits”, Oxford University Press: Fifth Edition 2004, ISBN 9780195142518. Chapter 2

[6] R. Jacob Baker. 2007. *CMOS: Circuit Design, Layout, and Simulation (IEEE Press Series on Microelectronic Systems)*. Revised 2nd edition. Wiley-IEEE Press. Chapter 13, Section 13.3

[7] R. Jacob Baker. 2007. *CMOS: Circuit Design, Layout, and Simulation (IEEE Press Series on Microelectronic Systems)*. Revised 2nd edition. Wiley-IEEE Press. Chapter 20, Section 20.3

[8] R. Jacob Baker. 2009. *CMOS: Mixed-Signal Circuit Design*. 2nd edition. Wiley-IEEE Press. Chapter 4, Section 4.1.3

[9] http://www.play-hookey.com/digital/shift-out_register.html

[10] http://www.mosis.com/Technical/Testdata/tsmc-025-prm.html

[11] R. Jacob Baker. 2007. *CMOS: Circuit Design, Layout, and Simulation (IEEE Press Series on Microelectronic Systems)*. Revised 2nd edition. Wiley-IEEE Press. Chapter 28

[12] Phillip E. Allen, Douglas R. Holberg. CMOS Analog Circuit Design. Second Edition. Chapter 10, Section 10.5

[13] R. Jacob Baker. 2009. *CMOS: Mixed-Signal Circuit Design*. 2nd edition. Wiley-IEEE Press. pp-176

[14] R. Jacob Baker. 2009. *CMOS: Mixed-Signal Circuit Design*. 2nd edition. Wiley-IEEE Press. pp-171

[15] http://cmosedu.com/cmos2/MSD_Matlab/MSD_MATLAB.htm

[16] http://vcag.ecen.okstate.edu/?page_id=12#

[17] Basham, E.J.; Parent, D.W.; , "An analog circuit implementation of a quadratic integrate and fire neuron," *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE* , vol., no., pp.741-744, 3-6 Sept. 2009

[18] D.A. Johns and K. Martin, *Analog Integrated Circuit Design*, John Wiley and Sons,Inc, New York, NY, 1997

[19] R. Jacob Baker. 2007. *CMOS: Circuit Design, Layout, and Simulation (IEEE Press Series on Microelectronic Systems)*. Revised 2nd edition. Wiley-IEEE Press. Chapter 29

[20] O'Driscoll, S.; Meng, T.H.; , "Adaptive resolution ADC array for neural implant," *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE* , vol., no., pp.1053-1056, 3-6 Sept. 2009

[21] Sun-Il Chang; Euisik Yoon; , "A low-power area-efficient 8 bit SAR ADC using dual capacitor arrays for neural microsystems," *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE* , vol., no., pp.1647-1650, 3-6 Sept. 2009

[22] You-Kuang Chang; Chao-Shiun Wang; Chorng-Kuang Wang; , "A 8-bit 500-KS/s low power SAR ADC for bio-medical applications," *Solid-State Circuits Conference, 2007. ASSCC '07. IEEE Asian* , vol., no., pp.228-231, 12-14 Nov. 2007

[23] Cheng-Ta Chiang; Li-Lung Kao; Yu-Chung Huang; , "A Low-Cost CMOS Integrated Dual-Mode Dual-Slope ADC with Synchronous Rectification Circuit for AC/DC Signal Measuring," *Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE* , vol., no., pp.165-169, 12-15 May 2008 doi: 10.1109/IMTC.2008.4547024

[24] Hairong Yu; Chang, M.-C.F.; , "A 1-V 1.25-GS/S 8-Bit Self-Calibrated Flash ADC in 90-nm Digital CMOS," *Circuits and Systems II: Express Briefs, IEEE Transactions on* , vol.55, no.7, pp.668-672, July 2008 doi: 10.1109/TCSII.2008.921596

APPENDIX I

VerilogA code for DAC

```
// Ideal current mode DAC model using VerilogA

`include "constants.vams"
`include "disciplines.vams"

module I_DAC(iout,din);
output iout; //Analog current out
electrical iout;
input [7:0] din; // 8 bit Digital input
electrical [7:0]din;
parameter real trise = 10e-12; //Rise time of output
parameter real tfall = 10e-12; //Fall time of output
parameter real tdel = 10e-12;  //Delay
real scale;
real sample;
real vtran;
real Iref;
analog begin
vtran=1.25;
Iref=95.57e-6; //Reference current
scale=0;
scale=scale + ((V(din[7]) > vtran) ? 128:0);
scale=scale + ((V(din[6]) > vtran) ? 64:0);
scale=scale + ((V(din[5]) > vtran) ? 32:0);
scale=scale + ((V(din[4]) > vtran) ? 16:0);
scale=scale + ((V(din[3]) > vtran) ? 8:0);
scale=scale + ((V(din[2]) > vtran) ? 4:0);
scale=scale + ((V(din[1]) > vtran) ? 2:0);
scale=scale + ((V(din[0]) > vtran) ? 1:0);

sample=Iref*scale/256;
I(iout) <+ transition(sample,tdel,trise,tfall);
end

endmodule
```

APPENDIX II

Verilog code for Ideal Current mode ADC and Sample and Hold

```
// Ideal current mode algorithmic ADC using VerilogA
`include "constants.vams"
`include "disciplines.vams"

module ADC_IDEAL(in,clk,Dout,I_out,Ir);
  parameter integer bits=8;
  parameter dlay=20e-12;
  parameter ttime=20e-12;
  parameter rtime=20e-12;
  parameter logic_high=2.5;
  input in,clk,ir; //Input Current Signal,Clock and Ref. Current
  output [7:0] Dout; //Output bits of ADC
  output I_out; //Sample and Hold Output
  electrical in,clk,ir;
  electrical [7:0] Dout;
  electrical I_out;
  real sample, midpoint,I8,I7,I6,I5,I4,I3,I2,I1;
  real result[7:0];
  real Iref;

analog begin
  @(cross(V(clk)-logic_high/2.0,+1))
begin
        Iref=I(ir);
        sample = I(in);

if(2*I(in)<=Iref) begin
       result[7]=0.0;
       I8=2*I(in);
end
if(2*I(in)>Iref) begin
       result[7]=logic_high;
       I8=2*I(in)-Iref;
end
if (2*I8<=Iref) begin
        result[6]=0.0;
        I7=2*I8;
end
if (2*I8>Iref) begin
        result[6]=logic_high;
        I7=2*I8-Iref;
end
if (2*I7<=Iref) begin
        result[5]=0.0;
        I6=2*I7;
end
```

```
if (2*I7>Iref) begin
            result[5]=logic_high;
            I6=2*I7-Iref;
end
if (2*I6<=Iref) begin
            result[4]=0.0;
            I5=2*I6;
end
if (2*I6>Iref) begin
            result[4]=logic_high;
            I5=2*I6-Iref;
            end
if (2*I5<=Iref) begin
            result[3]=0.0;
            I4=2*I5;
end
if (2*I5>Iref) begin
            result[3]=logic_high;
            I4=2*I5-Iref;
end
if (2*I4<=Iref) begin
            result[2]=0.0;
            I3=2*I4;
end
if (2*I4>Iref) begin
            result[2]=logic_high;
            I3=2*I4-Iref;
end
if (2*I3<=Iref) begin
            result[1]=0.0;
            I2=2*I3;
end
if (2*I3>Iref) begin
            result[1]=logic_high;
            I2=2*I3-Iref;
end
if (2*I2<=Iref) begin
                  result[0]=0.0;
                  I1=2*I2;
end
if (2*I2>Iref) begin
                  result[0]=logic_high;
                  I1=2*I2-Iref;
end
            end
V(Dout[7]) <+ transition(result[7],dlay,ttime,rtime);
V(Dout[6]) <+ transition(result[6],dlay,ttime,rtime);
V(Dout[5]) <+ transition(result[5],dlay,ttime,rtime);
V(Dout[4]) <+ transition(result[4],dlay,ttime,rtime);
V(Dout[3]) <+ transition(result[3],dlay,ttime,rtime);
V(Dout[2]) <+ transition(result[2],dlay,ttime,rtime);
```

```
V(Dout[1]) <+ transition(result[1],dlay,ttime,rtime);
V(Dout[0]) <+ transition(result[0],dlay,ttime,rtime);
I(I_out) <+ transition(sample,dlay,ttime,rtime);
end
endmodule
```

TURNITIN SUBMMISSION REPORT