Design 3 bit Minnick Counter Using Different Kind of Threshold logic

Thesis Submitted

In Partial Fulfilment of The Requirements

For The Award of the Degree of

Master of Technology
in
VLSI Design and Microelectronics Technology

By

Suvojijt Shome

Examination Roll No.: **M6VLS19002**

University Registration No.: **137279 of 2016-2017**

University Class Roll No.: **001610703007**

Under the Supervision of

Prof. Subir Kumar Sarkar

Department of Electronics & Tele-Communication Engineering

Jadavpur University,

Kolkata-700032 West Bengal, India

May 2019

Faculty of Engineering & Technology

Jadavpur University

CERTIFICATE OF EXAMINATION

This is to certify that the thesis entitled "**Design 3 bit Minnick Counter Using Different Kind of Threshold logic**" has been carried out by **Suvojijt Shome (Roll No: 001610703007, Examination Roll No: M6VLS19002 and Registration No: 137279 of 2016-17)** under my guidance and supervision and can be accepted in partial fulfilment for the degree of Master of Technology in VLSI Design and Microelectronics Technology. In my opinion the work fulfils the requirement for which it is submitted. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other organization.

Prof. Subir Kumar Sarkar

Project Supervisor

Prof. Sheli Sinha Chaudhuri

Head of the Department,
Department of Electronics and
Tele-Communication Engineering,
Jadavpur University,
Kolkata-700032

Prof. Chiranjib Bhattacharjee

Dean,
Faculty Council of Engineering
and Technology (FET),
Jadavpur University,
Kolkata-700032

Faculty of Engineering & Technology

Jadavpur University

CERTIFICATE OF APPROVAL*

The foregoing thesis is hereby approved as a credible study of an engineering subject and presented in a manner satisfactory to authorization acceptance as pre-requisite to the degree for which it has been submitted.it is understood that by this approve any statement made, opinion or conclusion drawn there in but approve the thesis only for which it is submitted.

Committee on Final Examination

For the Evaluation of the Thesis

____

____

Examiner

***Only in the case the thesis is approved.**

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that the thesis contains literature survey and original research work done by the undersigned candidate, as a part of her Master of Technology in VLSI Design and Microelectronics Technology.

All information in this document has been obtained and presented in accordance with the academic rules and ethical conduct.

I also declare that as required by the code of conduct, I have fully cited and referenced all material and results that are not original to this work.

Suvojijt Shome
Examination Roll No. : **M6VLS19002**

University Registration No.: **137279 of 2016-2017**

Thesis Title: **"Design 3 bit Minnick Counter Using Different Kind of Threshold logic"**

____

Suvojijt Biswas
(Signature with date)

ACKNOWLEDGMENT

This thesis is the end of my journey in obtaining my master degree. This thesis has been kept on track and been through to completion with the support and encouragement of numerous people including my well-wishers, my friends, and teacher. At the end of my thesis, I would like to thank all those people who made this thesis possible and an unforgettable experience for me.

First and foremost, I express my sincerest gratitude to my supervisor **Prof. Subir Kumar Sarkar**, who has supported me throughout my thesis with patience and knowledge. Without him, this thesis would not has been a successful one.

I would also like to thank **Prof. Sheli Sinha Chaudhuri**, Head of the Department, Electronics and Telecommunication Engineering, Jadavpur University for her generous help and providing me the necessary infrastructure and labs for this project.

I wish to express my sincere thanks to all my teachers and my batch mates for providing necessary information whenever required. I am also thankful to all my friends **Mr. Ayan Bhattacharjee, Mr. Arindam Halder, Mr. Bikram Biswas** for their suggestion and encouragement. The words of thanks are only a token of my true appreciation for all they have done to make my project in the present shape.

Finally, I thank my family members, their support encourage me to reach up to this level. Their understanding and admiration is the key to my success. I pray for their happiness and good health, and I dedicate my work to them in the most sincere way I can think of.

Finally, I thank God who has been kind enough to me to perform this work.

Suvojijt Shome

M.Tech in VLSI Design and Microelectronics Technology

Department of ETCE, Jadavpur University
Kolkata-700032, West Bengal, India

CONTENTS

ABSTRACT

The demands of upcoming computing, as well as the challenges of nanometer-era of VLSI design necessitate new digital logic techniques and styles which are at the same time high performance, energy efficient and robust to noise and variation.

Threshold logic gates are mainly used to design high performance circuits because of their high speed. Conversely, the vital demerit of threshold logic style is its high power consumption. With continuous technology scaling, this problem is getting more and more severe. For this reason a new circuit has been proposed which can implement various linearly separable logic functions in one circuit by varying circuit control voltage. A multi valued circuit also designed to generate discrete voltage levels. A detailed study of AND, OR, XOR and Majority Gate and Full Adder circuit has been implemented by using various circuit design techniques such as: capacitive voltage divider threshold logic (CVDTL), CMOS Output Wired logic circuit. A minnick counter also has been designed here.

Minnick counter is a special kind of electronic circuit that is employed to determine the number of ones in a binary input sequence. Here I have proposed a circuit which can determine the number of ones in a binary sequence. If we look close to the circuit then we may find that this circuit can also be used as a Full Adder circuit with a little modification. The three bit Minnick Counter Circuit can act like an one bit Full Adder .The circuits were designed and then simulated using simulation software Tanner EDA for validation.

CHAPTER: 1

1.1. INTRODUCTION

Very Large Scale Integration (VLSI) is the process of creating an Integrated Circuit (IC) by combining thousands of transistors into a single chip. VLSI began in 1970s when complex semiconductors technologies were being developed. The microprocessor is a VLSI device. Before the introduction of VLSI technology, most ICs had a limited date of functions they could perform. An electronic circuit might consist of CPU, ROM, RAM and other logic elements. VLSI lets IC designers to add allow these into a single chip.

Today the entire domain of electronics – from designing to fabrication and testing – everything falls under the domain of VLSI technology. Though engineers and scientists have geared up to implement ULSI (Ultra Large Scale Integration), VLSI still remains as a touted choice among designers since VLSI is based on the micron technology and designing circuits in micron dimension is comparatively easier than designing the same in nano or pico dimensions.

Keeping this in mind, current project is based on culturing a part of this VLSI technology, design aspect where I have not only designed the circuits in concerned but also I have simulated them using VLSI simulation tools.

We know that the cell level of every circuit is the logic gates. Conventional logic gates include AND, NAND, OR, NOR, XOR, XNOR & NOT gates namely. Here, I have designed a different logic element called the threshold gate or T-gate. Using the T-gate, I have shown that the number of transistors required for designing the various circuits has been reduced.

Here the Threshold gate has been designed using Output Wired CMOS Inverter Circuit, Capacitive voltage divider Threshold Logic (CVDTL).Minnick Counter has been designed by using capacitive voltage divider Threshold Logic and Output Wired logic.

All the circuits included thereafter have been designed and simulated using Tanner EDA. Tanner is useful software for electronic circuits. It is used to design and simulate electronic circuits so that we can see the responses both DC and transient responses and measurement of various circuit parameters.

1.2. THRESHOLD LOGIC

Threshold logic (TL) itself was first introduced by Warren McCulloch and Walter Pitts in 1943 in the early model of an artificial neuron based on the basic property of firing of the biological neuron. The logic computed the sign of the weighted sum of its inputs. Modeling a neuron as a threshold logic gate (TLG) that fires when input reaches a threshold has been the basis of the research on neural networks (NNs) and their hardware implementation in standard CMOS logic.

A linear threshold gate (LTG) is an n binary input and one binary output function. Threshold Logic Gates are able to compute any linearly separable Boolean function given by -

$$Y = sgn\big(f(x)\big) = \begin{cases} 0, iff(x) < 0 \\ 1, iff(x) \geq 0 \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(1)}$$

$$f(x) = \sum w_i x_i - \psi \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2)}$$

Where x_i is the n Boolean inputs and w_i is the corresponding n integer weights. The LTG compares the weighted sum of inputs and the threshold value ψ. If the weighted sum of inputs is greater than or equal to the threshold, the gate produces logic 1. Otherwise, the output is logic.

The basic Boolean logic functions AND, OR, NAND, NOR and NOT can be represented in the form of (1) and (2) as follows:

AND(x1, x2) = sgn{x1+ x2-1.5}

OR(x1, x2) = sgn{x1+ x2-0.5}

NAND(x1, x2) = sgn {- x1- x2+1.5}

NOR(x1, x2) = sgn {- x1- x2+0.5}

NOT (x1) = sgn {- x1+0.5}

These basic logic functions can be designed with threshold gate as shown in figure 1.In this diagram AND gate is designed. Other gates can be also designed by changing the weights and threshold values.

Threshold gates can be implemented by analog as well as digital circuits. Here we have implemented the majority gate which is a threshold gate using output wired CMOS inverter circuit, Capacitive voltage divider threshold logic and β driven threshold logic. Here the threshold gate has been implemented using β driven transistors where the current amplification factor is varied in proportion to the weight to be implemented by the following 2 methods:

- By varying the device geometry i.e. by varying (W/L) of the MOS.
- By varying a control voltage which in turn controls the effective β of the MOSFET .

1.3. CAPACITIVE VOLTAGE DIVIDER THRESHOLD LOGIC (CVDTL)

The CVDTL gate derives its name from the capacitors which are used to implement the weights. There are several advantages of using capacitors to implement the weights:

- When the process which is used to integrate the capacitors supports two poly silicon layers which is not uncommon capacitors take up very little area very high integration densities can therefore be achieved.

- Capacitors of identical value can be easily reproduced. Note that it is difficult to produce a capacitor of a given value. With CVDTL gates only the relative values of capacitors are important. Although it is difficult to make a capacitor which exactly twice (or any number of times) as large as another capacitor, it is no problem whatsoever to make many capacitors which are all equal in value. This is called a “unit capacitor”. Any desired ratio between capacitors can be created by putting the required number of unit capacitors in parallel.

- Because of the way capacitors are used here the gate inputs have very high impedance. The sources driving the inputs have to deliver only very limited power, which should benefit the speed of circuits, made up the CVTL gates.

These qualities make capacitors pre-eminently suited to be used as weights in a TL gate implementation. Resistors which are about the only alternatives available score very weak on the qualities mentioned above. They consume quite a lot of idea to be integrated, and a difficult to manufacture to a specified accuracy. For an area efficient integration of resistors and for the integration of registers with precise values additional processing steps are needed which makes the use of registers and weights far less generally applicable.

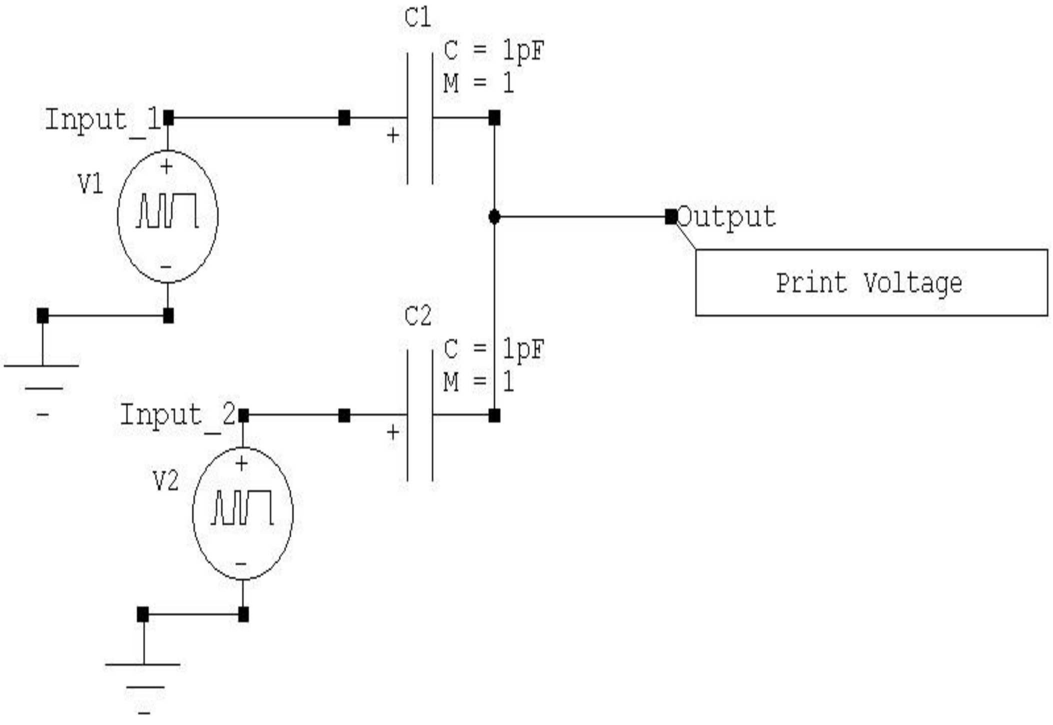

Fig. 1.1: A typical two input capacitive voltage divider circuit

This figure shows a capacitive voltage divider with two inputs. One terminal of each capacitor is connected to a common node and the other terminal of each capacitor is connected to the input.

Note that there are as many capacitors as there is input, and that each capacitor can be connected to exactly one input. The (relative) size of capacitors determinants the weight which is associated to the input.

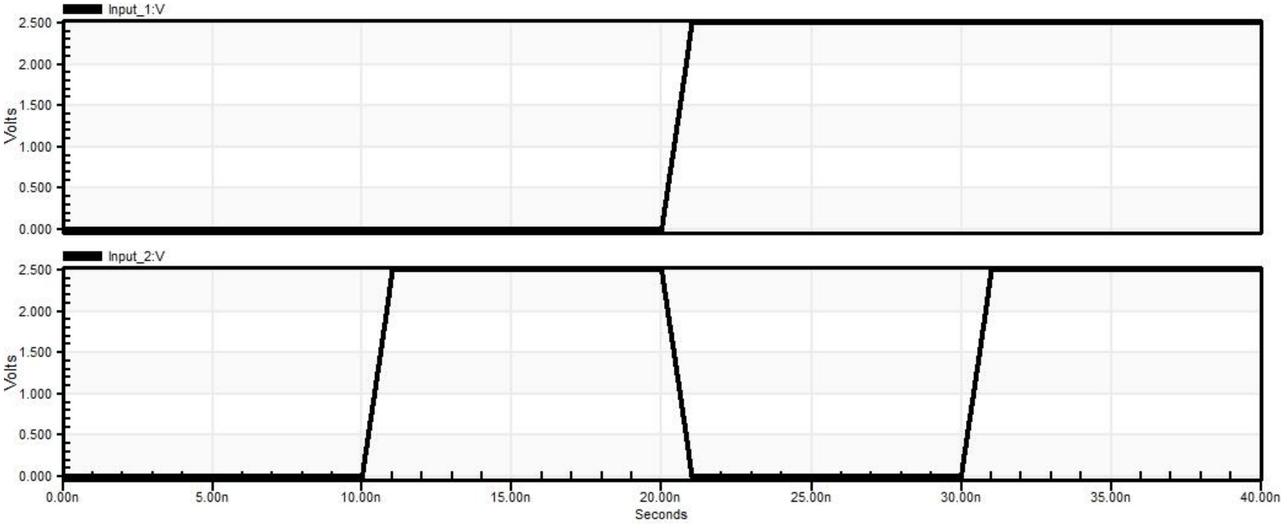

Fig. 1.1: Input sequences for previous circuit

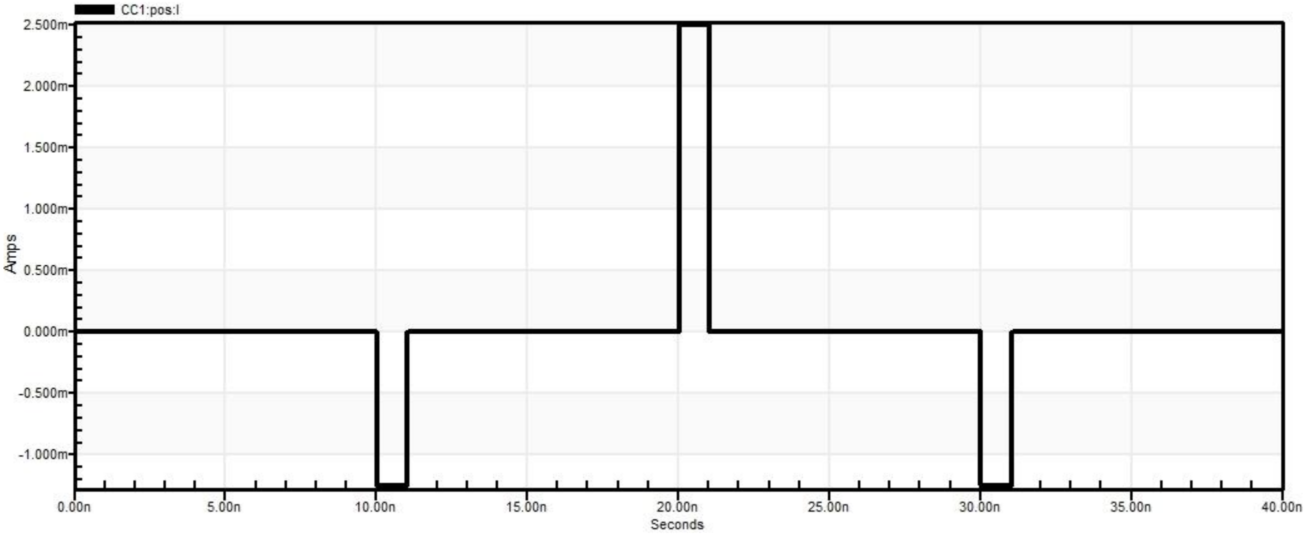

Fig. 1.2: Current waveform of - C1

If we carefully observe the current graph of the capacitive voltage divider, then we can see that the current is only flow when the inputs change their states.

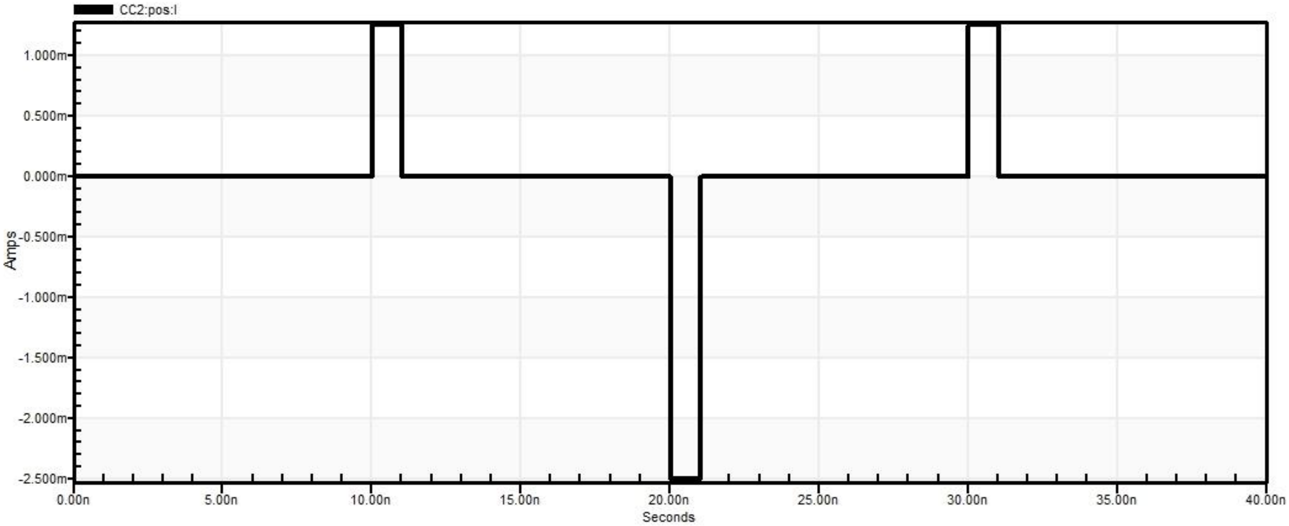

Fig. 1.3: Current waveform of – C2

In the circuit implementation the common node to which all the capacitors are connected is also connected to the input of an inverter. The output was decided by the threshold value of the inverter and the voltage level at the common node.

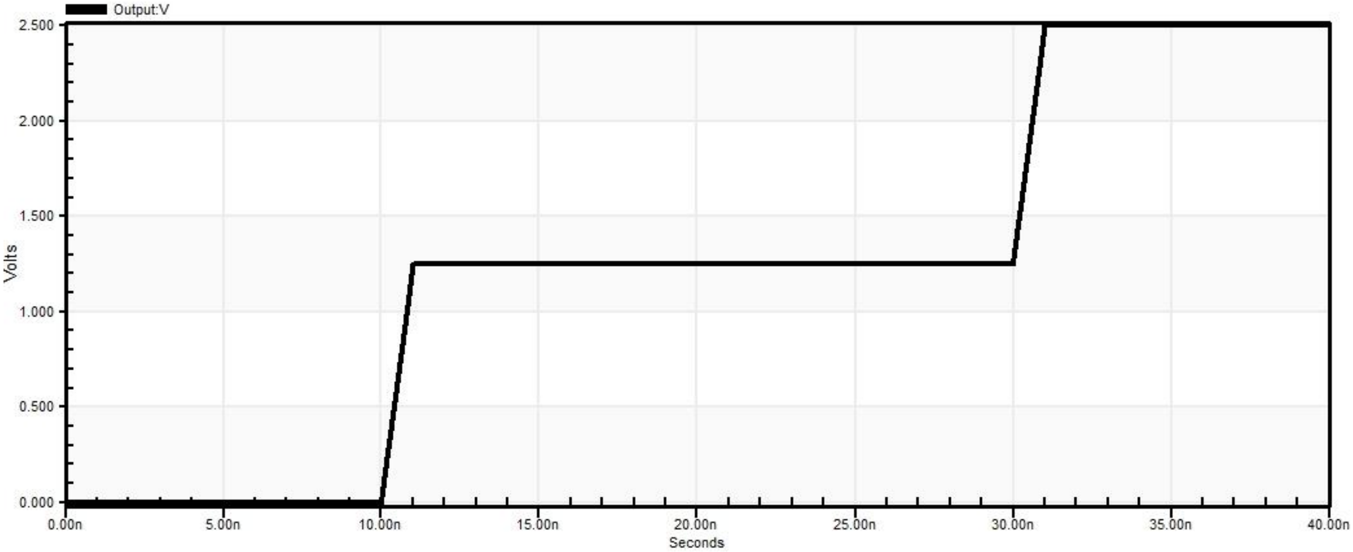

Fig. 1.4: Voltage level at the common node for different input sequence

In theories regarding threshold logic the threshold value is usually specified as an integer number. This is convenient as the weights are also specified as integers. The use of the Sgn function, which is a step function allows for convenient modeling of the threshold behavior.

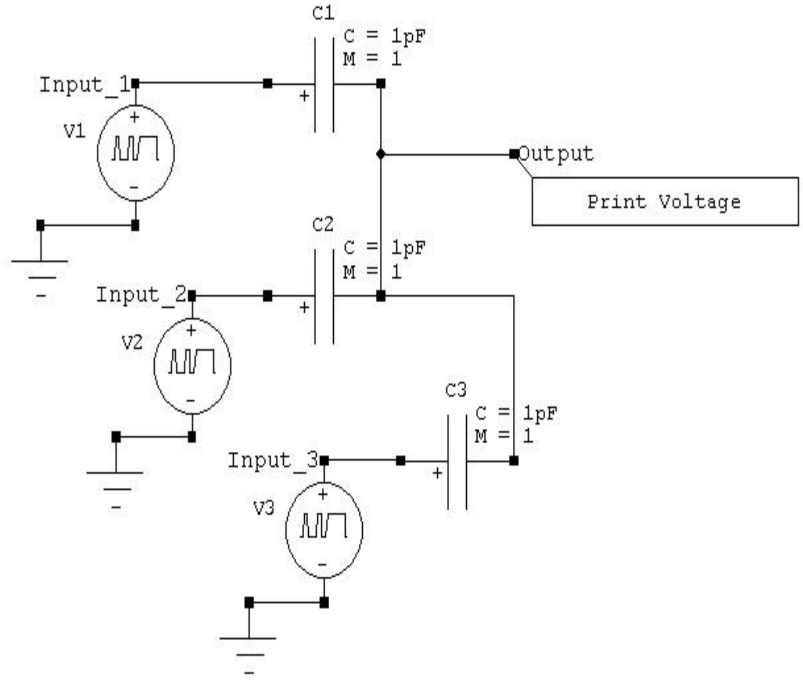

Fig. 1.5: A typical three input capacitive voltage divider circuit

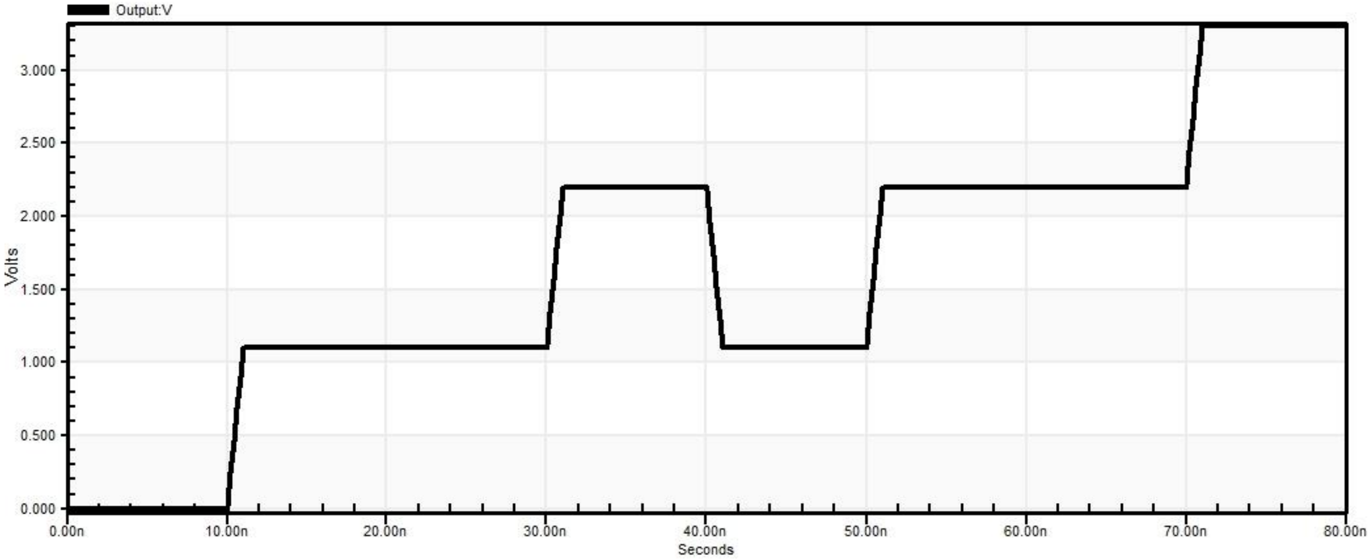

Fig. 1.6: Voltage level at the common node for three inputs

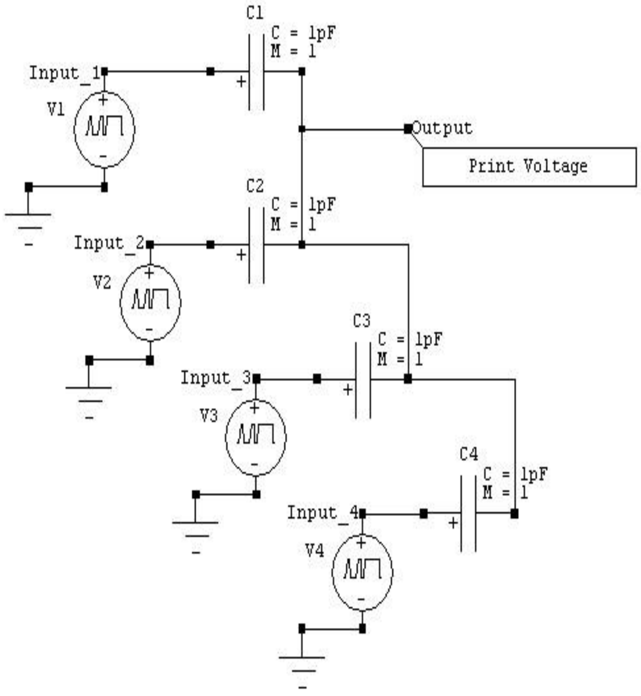

Fig. 1.7: A typical three input capacitive voltage divider circuit

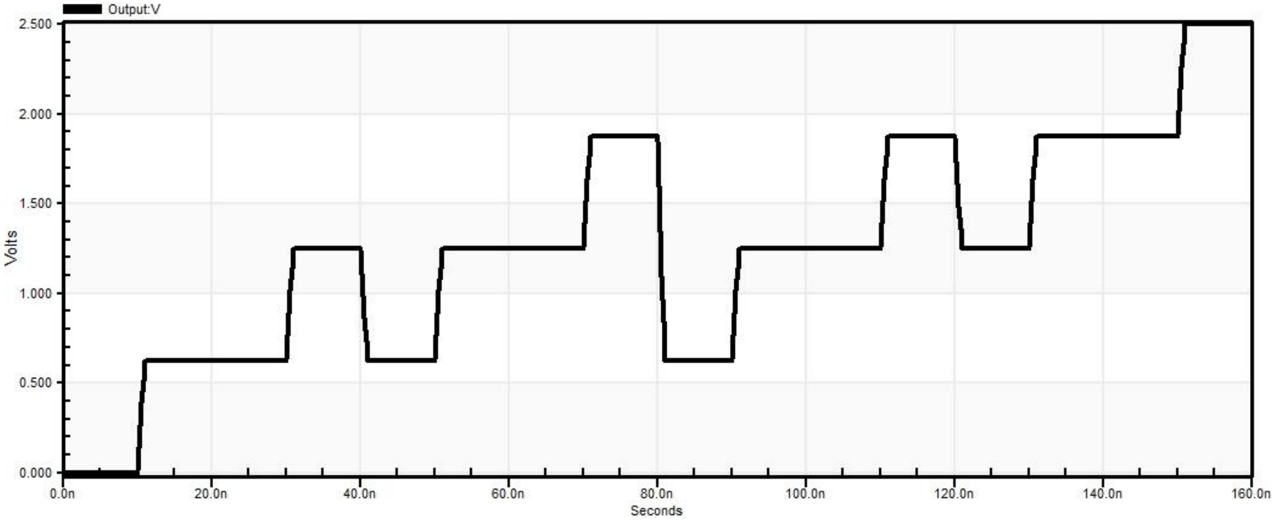

Fig. 1.8: Voltage level at the common node for four inputs

In this chapter the concept of capacitive voltage divider threshold logic was explored as a result several observations can made regarding the capacitive voltage divider logic Gate:

The exact value of threshold voltage of the inverter is not important for the functioning of CVDTL.

The exact value of the capacitors is not important, only their value relative to each other matters, and should be sufficiently accurate. This is especially true when the weights vary over wide range.

Operation of CVDTL gate is centered around the voltage at the common node. The voltage assumes discrete values, with the size of the steps being determinant by the number of inputs. This limits the maximum number of weights which can be used in a single CVDTL gate.

1.4. CMOS Output Wired logic circuit

The terms CMOS output wired logic circuit refers to a CMOS circuit where the outputs of several inverters are wired together. Instead of acting as a switches (standard digital CMOS) the transistors act as a variable resistors controlled by their gate voltages. The circuit may thus be represented as a resistor network where conducting transistor are represented by small resistors and non conducting transistor by large resistor.
It is possible to exploit the concept of a resistor network for generating various logic functions.

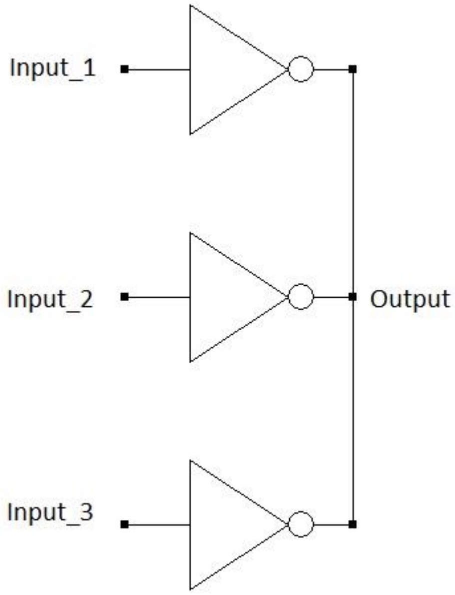

Fig. 1.9: Typical CMOS output wired logic circuit for three inputs

In the previous circuit (Fig.1.9) if the below input sequence was fed then, a multi valued signal was generated.

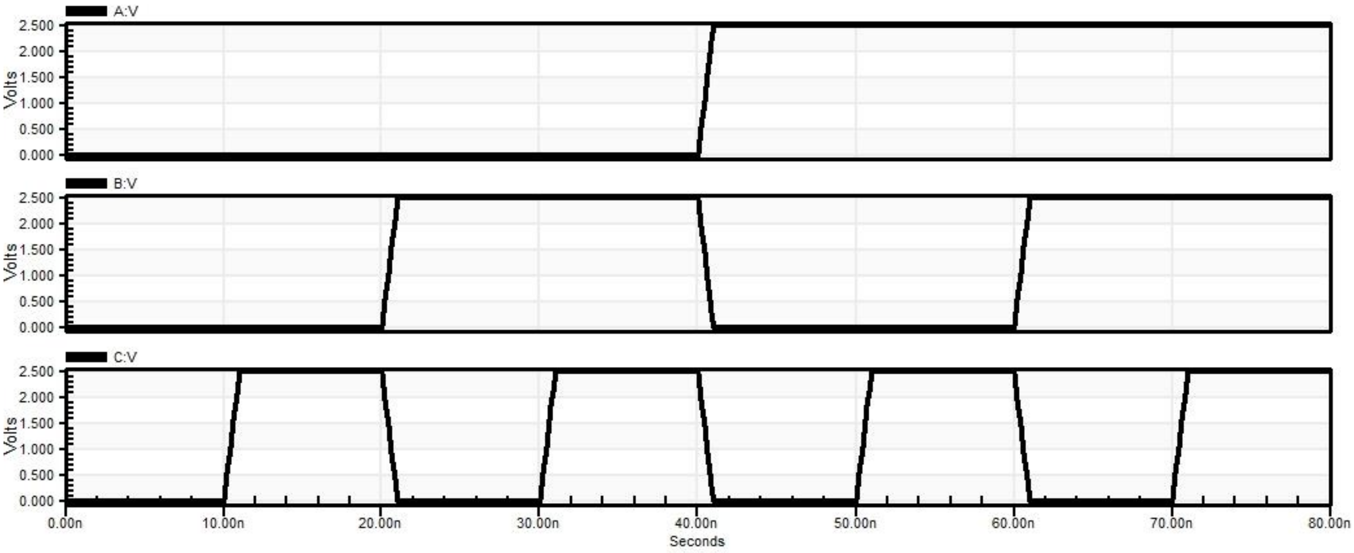

Fig. 1.10: A typical inputs sequence

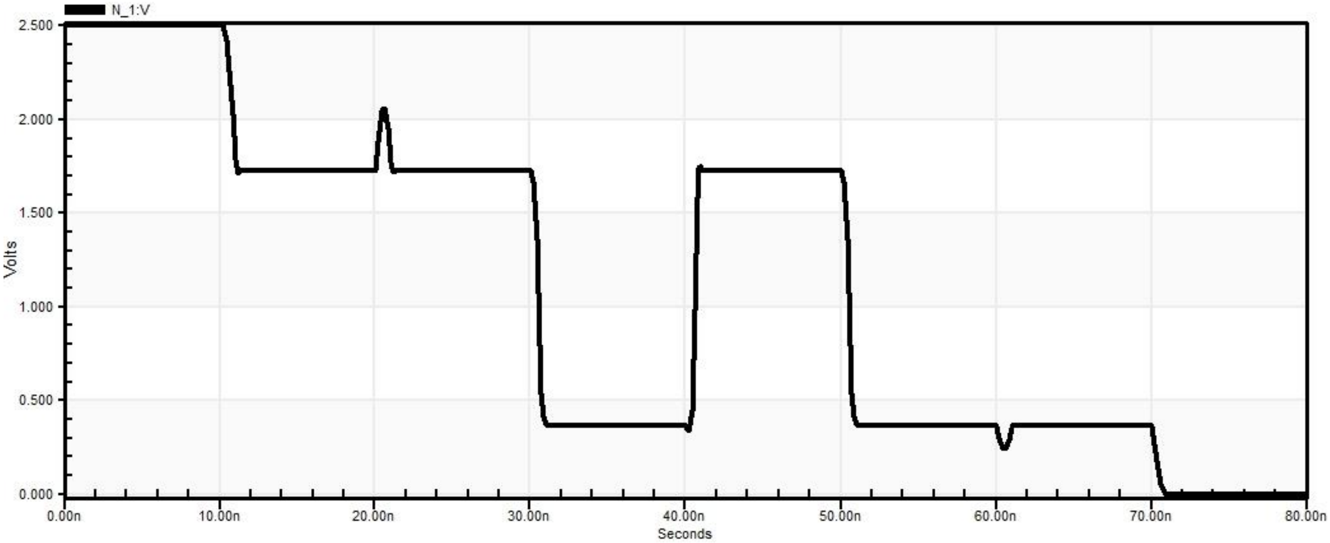

Fig. 1.11: A multi valued signal

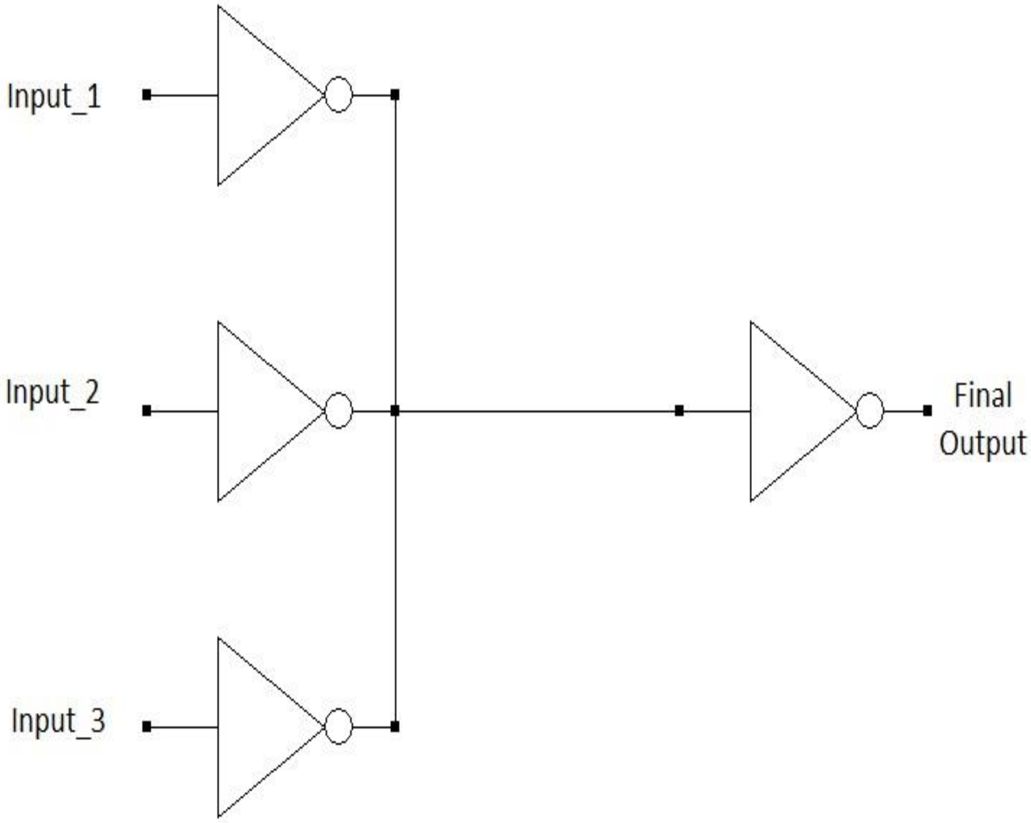

Fig. 1.11: Typical CMOS output wired logic circuit with a threshold element

By changing the aspect ratio (W/L) of threshold element we can generate various logic functions, without increasing the circuit complexity.

By buffering the ganged the node with a simple CMOS inverter, a number of advantages are obtained. First of all the ganged node is effectively isolated from external circuitry, its value is neither transmitted on long interconnect wires and corrupted by noise, nor does it drive complex gates, where any voltage exceeding a transistor threshold can cause a logic error. Essentially, one can tolerate much lower noise margin on a local node than on a global node. This benefit is enhanced by the inverter's inherent encoding action, it is high gain results in a sharp distinction between low and high input.

CMOS output wired logic circuit results in lower transistor count for more Complex functions. Although it is true that the Exclusive use of inverters limit the area saving parallel and series layout of transistors, the lower transistor count overcome this area deficiency. For Further area saving, it is possible to group n- channel and p-channel transistors, such that the inverter's two transistors are not constrained to be physically adjacent. It is also possible to alter p and n area requirement by enhancing the encoding inverter threshold.

1.5 β- DRIVEN THRESHOLD LOGIC

Hardware implementation of threshold logic elements is intriguing. Since threshold functions are a subclass of Boolean functions, any threshold function can be naturally represented as a superposition of operations of any functionally full basis. The question arises whether this design is a traditional one or a more simplified implementation of any arbitrary logic functions.

A threshold function is defined as

$$y = sign(\sum_{i=0}^{n-1} w_i x_i - T) = \begin{cases} 1 & \text{if } \sum_{i=0}^{n-1} w_i x_i - T) \geq 0 \\ 0 & \text{if } \sum_{i=0}^{n-1} w_i x_i - T) < 0 \end{cases}$$

Where w_i is the weight of the i^{th} input and T is the threshold. Thus a threshold should consist of an adder and a threshold comparator. Hence, a threshold element is an analog-discrete or at least, multi-valued binary element. So, all the problems we have in multi-valued logic can be implemented using threshold logic.

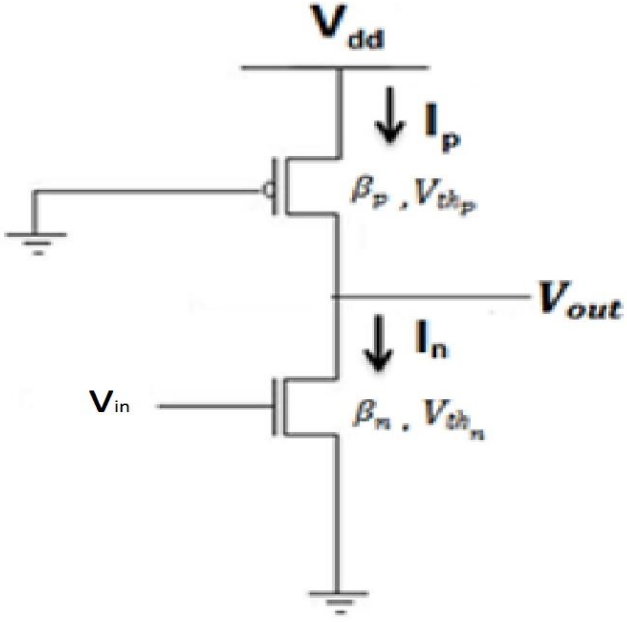

Fig. 1.12**:** Basic β- driven structure

In the static mode, when both transistors are completely conducting the output voltage V_{out} is determined by $\alpha = \beta_n/\beta_p$ and by $V_{th,p}$ and $V_{th,n}$ .V_{out} drops when α increases and increases when α decreases. This trivial thing is the base for the idea of a threshold element implemented by output-wired CMOS inverters. The class of threshold circuits produced by the CMOS couple in Fig. 1 will be referred to as the β-driven threshold elements (βDTE).

Effect of α on the output voltage of the basic CMOS structure

The following 3 possibilities of $\alpha = \beta_n/\beta_p$have been simulated.

Case 1:$\alpha = \frac{\beta_n}{\beta_p} = 1$

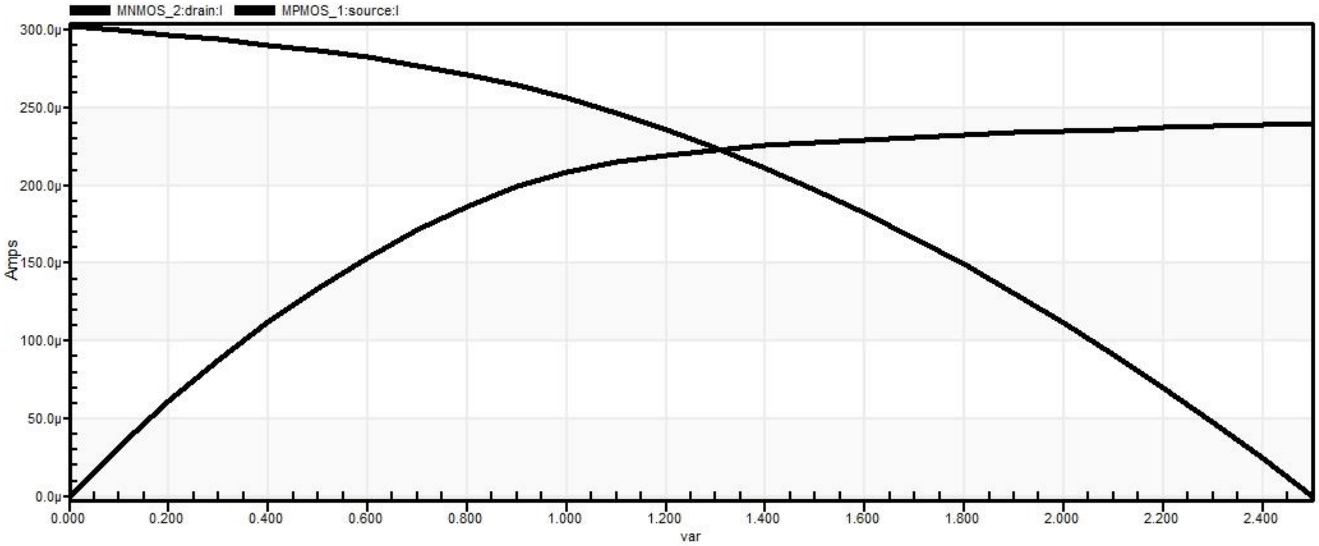

Fig. 1.13**:** I-V Characteristics for α = 1

Case 2:$\alpha = \frac{\beta_n}{\beta_p} > 1$

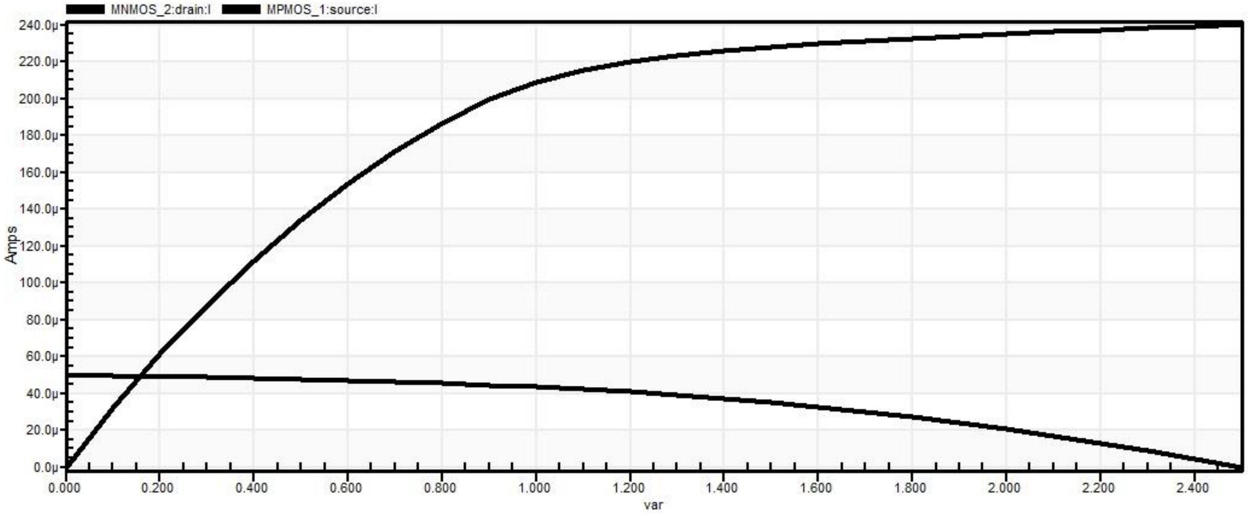

Fig. .1**:14** I-V Characteristics for $\alpha > 1$

Case 3:$\alpha = \frac{\beta_n}{\beta_p} < 1$

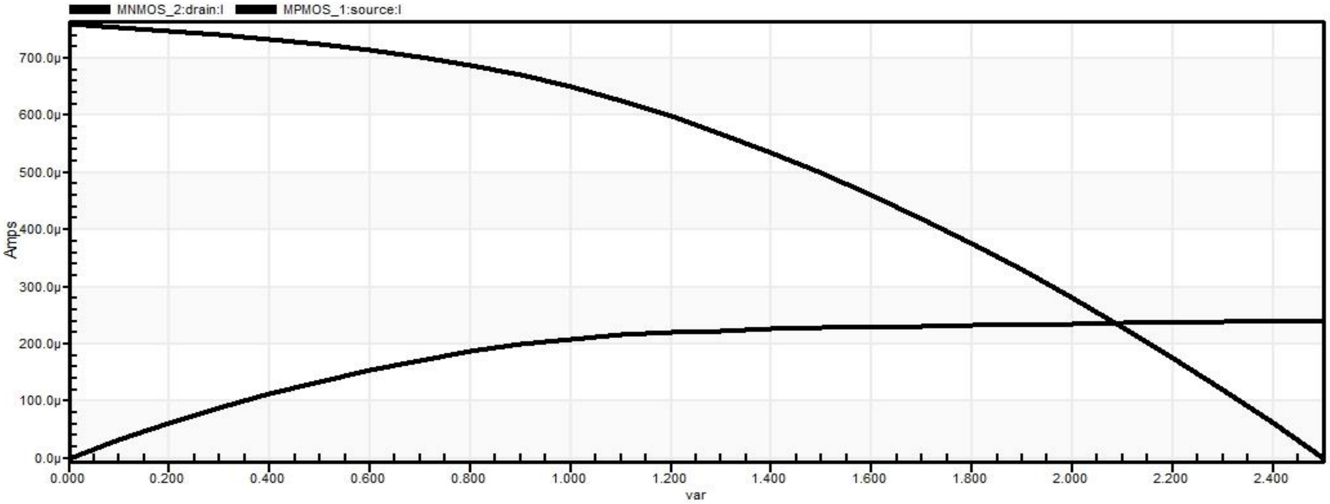

Fig. 1.15.**:** I-V Characteristics for $\alpha < 1$

It is evident from the above simulation results that the output voltage V_{out} and the amount of Drain Current (I_D) flowing through the CMOS circuit depends on the value of β.The above observation can be summarized in the following table (assuming that $|Vth,p| = |Vth,n|$)

$\boldsymbol{\beta_n, \beta_p}$	$\boldsymbol{\alpha = \beta_n/\beta_p}$	**V_{out}**
$\beta_n > \beta_p$	$\alpha > 1$	$Vout < \frac{Vdd}{2}$
$\beta_n = \beta_p$	$\alpha = 1$	$Vout = \frac{Vdd}{2}$
$\beta_n < \beta_p$	$\alpha < 1$	$Vout > \frac{Vdd}{2}$

Now we consider 2 NMOS, M1 and M2 with respective β-values being β_1 and β_2 connected in parallel.X_1 and X_2 are their respective inputs.

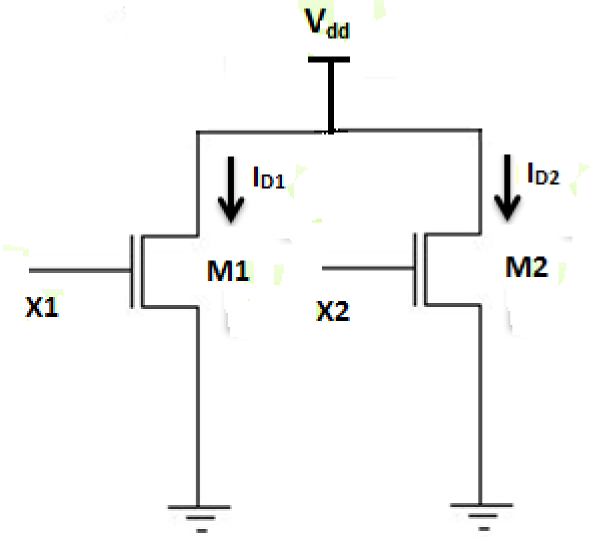

Fig. 1.16**:** Two NMOS connected in parallel

	$\underline{X_1}$	$\underline{X_2}$	$\underline{I_{D1}}$	$\underline{I_{D2}}$	$\underline{I_{total}}$
<u>Case 1:</u>	1	0	β_1 f($V_{DS,}$$V_{GS}$)	0	β_1 f($V_{DS,}$$V_{GS}$)

Case 2:	0	1	0	β_2 f($V_{DS,}V_{GS}$)	β_2 f($V_{DS,}V_{GS}$)
Case 3:	1	1	β_1 f($V_{DS,}V_{GS}$)	β_2 f($V_{DS,}V_{GS}$)	($\beta_1 + \beta_2$) f($V_{DS,}V_{GS}$)

The above table shows that the total current through the circuit (I_{total}) depends on the sum total of the β-values of the NMOS. So, $\beta_{eff} = \beta_1 + \beta_2$To be more precise $\beta_{eff} = X_1\beta_1 + X_2\ \beta_{2.}$ Thus, if we consider 'n' number of NMOS M_1, $M_2 \ldots M_n$ with respective β-values being β_1, $\beta_2 \ldots\ \beta_n$ connected in parallel as shown in Fig. 4.6, then effective β-value for NMOS is

$$\beta_{n,eff} = \sum_{i=1}^{n} X_i\,\beta_i$$

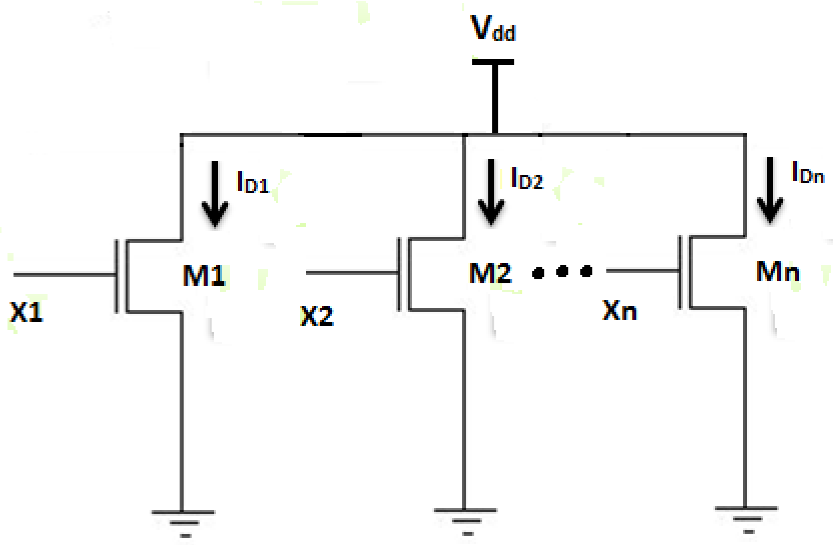

Fig. 1.17**:** 'n' NMOS connected in parallel

Similarly, if we have 'p' number of PMOS connected in parallel as shown in Fig. 1.17, then effective β-value is

$$\beta_{p,eff} = \sum_{j=1}^{p} \bar{X}_j \beta_j$$

In this case, $\bar{X}_j$ is considered because a pMOS is turned on only when V_{SG}= V $_{dd}$.

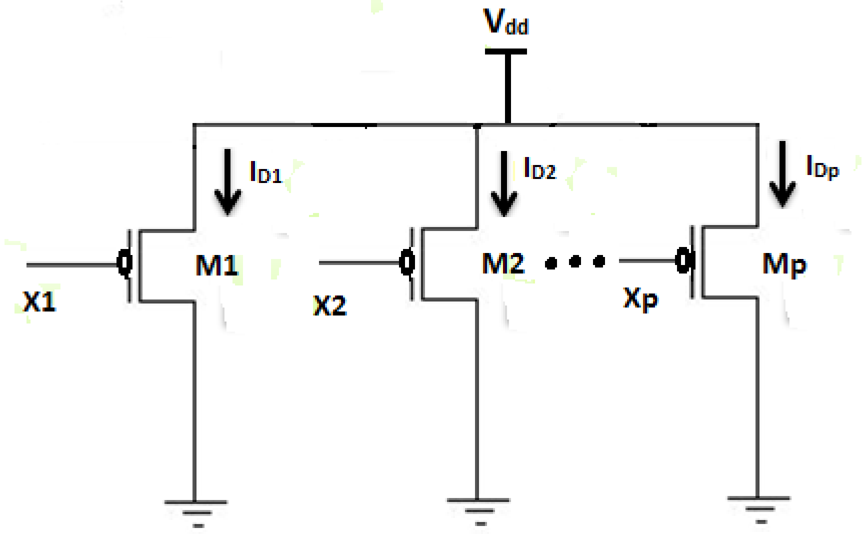

Fig. 1.18**:** ‘p’ PMOS connected in parallel

Hence, for a CMOS circuit having ‘n’ number of NMOS and ‘p’ number of PMOS shown in Fig. 3.d, the NMOS and PMOS blocks can be separately considered as a single combination of one NMOS and one PMOS with respective β-values being $\beta_{n,eff}$ and $\beta_{p,eff}$ as shown in Fig. 4.8

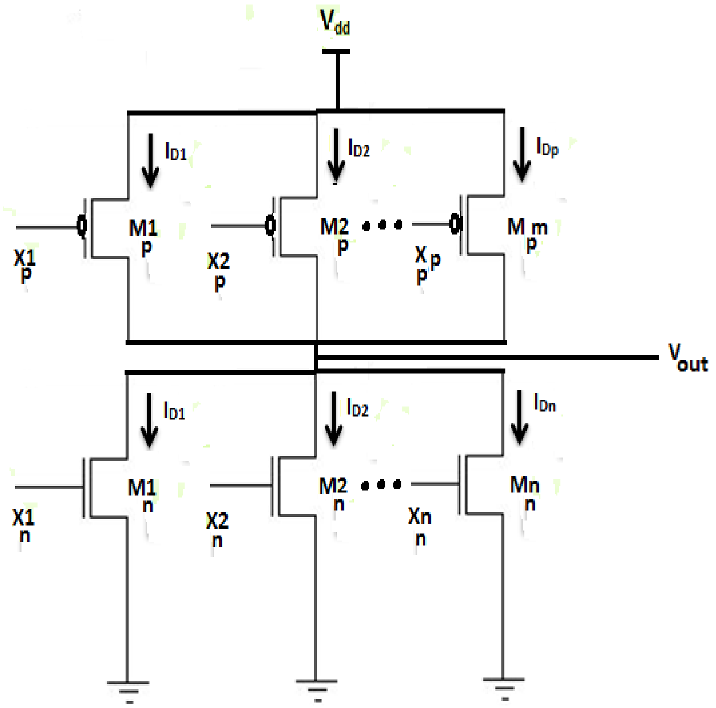

Fig. 1.19**:** Multiple p-MOS and n-MOS arrangement

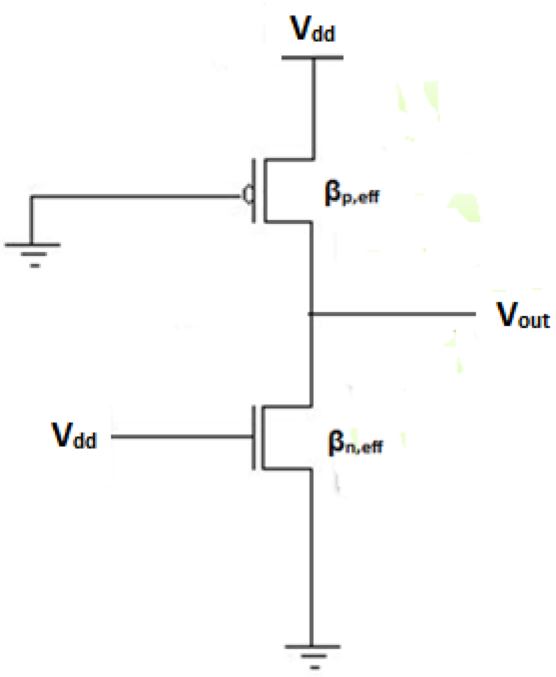

Fig. 1.20**:** Effective CMOS circuit

Hence, it can be said that $\alpha = \frac{\beta_{n,eff}}{\beta_{p,eff}} = \frac{\sum_{i=1}^{n} X_i}{\sum_{i=1}^{n} \bar{X}_i}$

Synthesis. If the output inverter threshold is properly selected, the circuit implements a majority function of n variables$Sign(\alpha - 1)$.

$$Sign(\alpha - 1) = Sign\left(\sum_{i=1}^{n} X_i - \sum_{i=1}^{n} \bar{X}_i\right) = Sign\left(\sum_{i=1}^{n} X_i - \sum_{i=1}^{n} (1 - X_i)\right)$$

$$= Sign\left(\sum_{i=1}^{n} X_i - \frac{n}{2}\right)$$

The input weights for an arbitrary threshold function can be introduced by multiple magnifications of the transistors' widths in the respective invertors. The threshold can be changed with respect to average level by incorporating permanently open transistors of appropriate width. This seemingly exhausts the task of βDTE logical.

CHAPTER: 2

2.1. IMPLEMENTATION OF CAPACITIVE VOLTAGE DIVIDER THRESHOLD LOGIC (CVDTL)

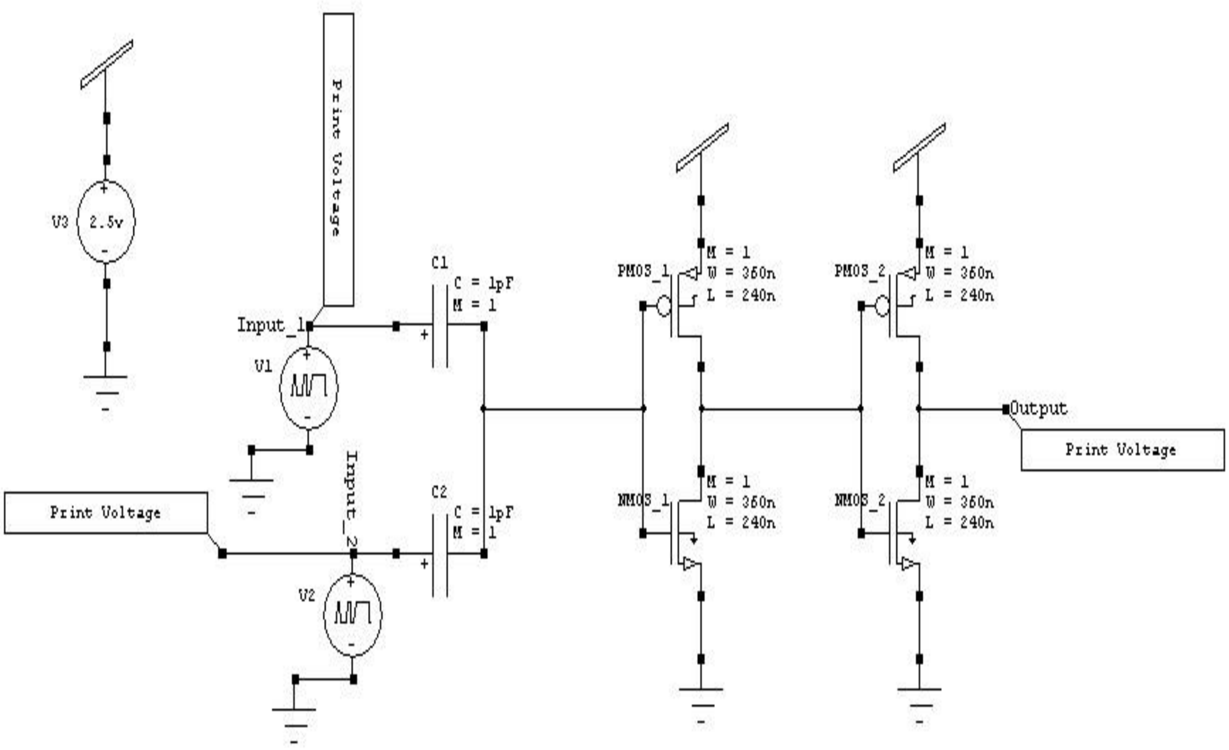

Fig. 2.1**:** OR gate circuit diagram using CVDTL

In the above picture capacitor divider circuit use as a driver circuit (two same value capacitors have been used here).First CMOS inverter circuit has been used as a sensor. It detects two values .If input is less then threshold value then it shows logic one. If input is greater then threshold value, then it shows logic zero. If we shift the threshold value of this cmos by the varying of width of mos (by different nano meter technology), then we design various logic gate. here we design or gate by using CVDTL.

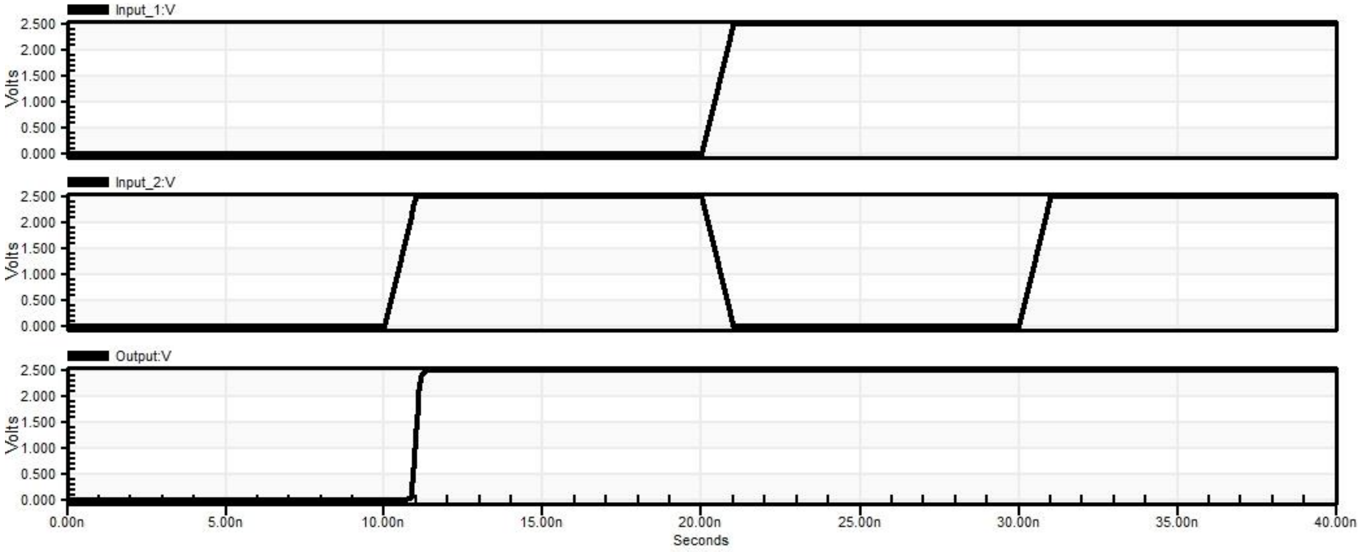

Fig. 2.2**:** OR gate output waveform using CVDTL

The above wave form shows, if any input is one then output will be one.

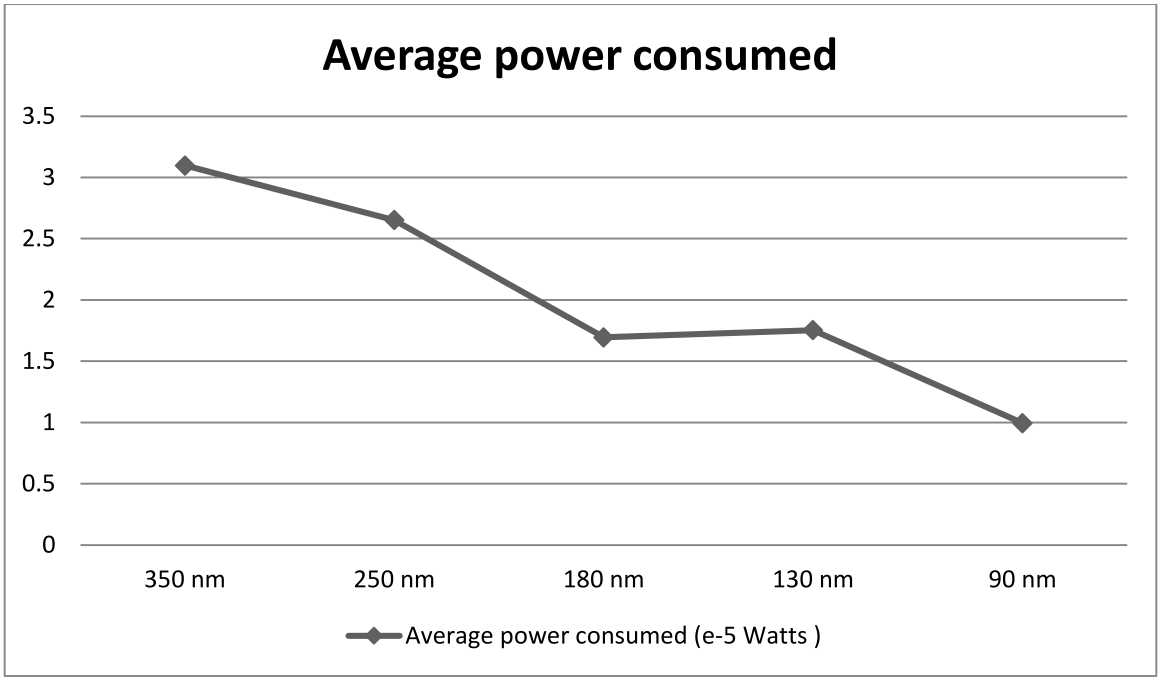

Fig. 2.3**:** Average power consumed by OR Gate using CVDTL in different NM

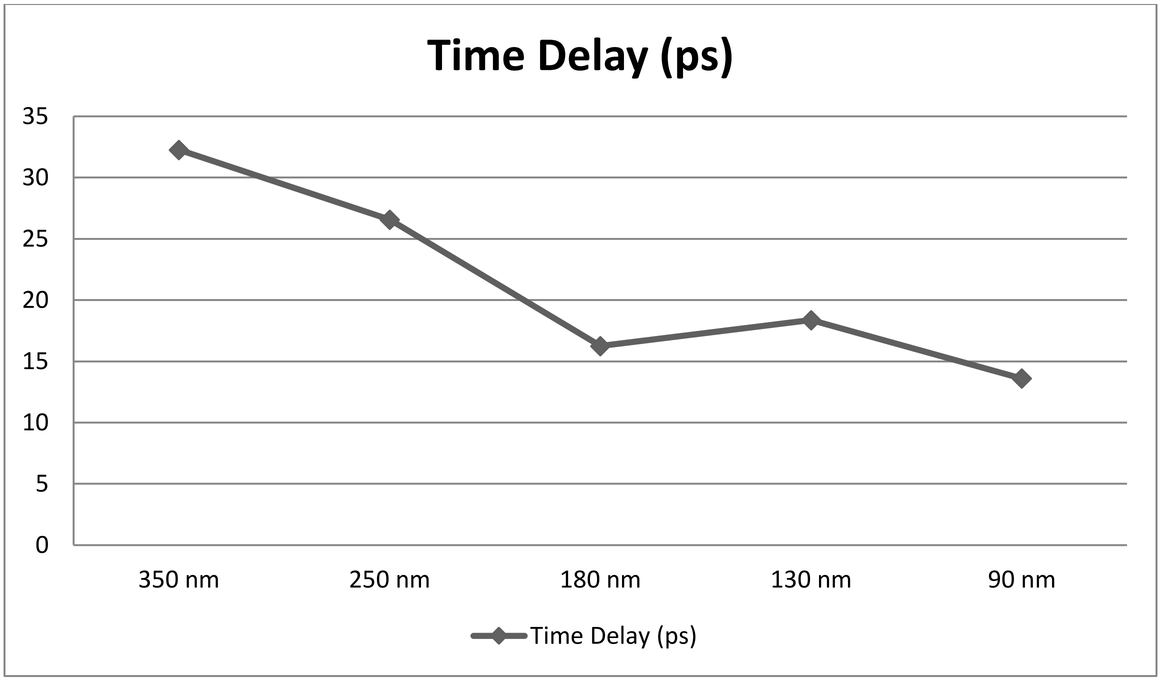

Fig. 2.4**:** Time delay of OR Gate using CVDTL in different NM

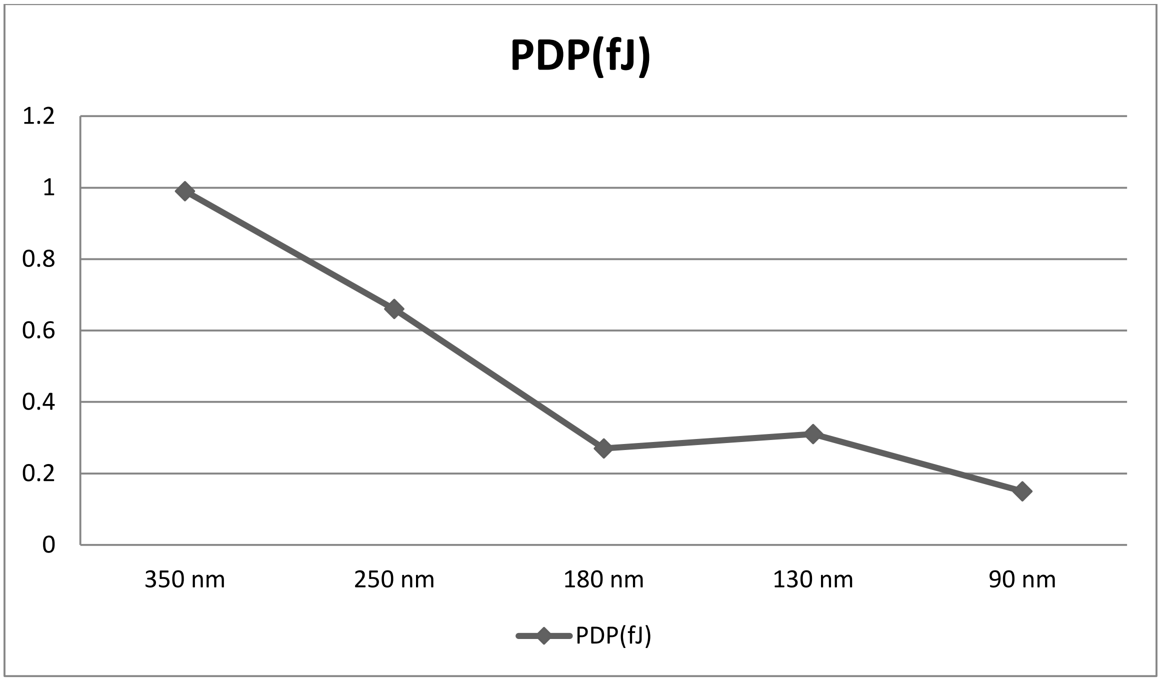

Fig. 2.5**:** Power Delay Product of OR Gate using CVDTL in different NM

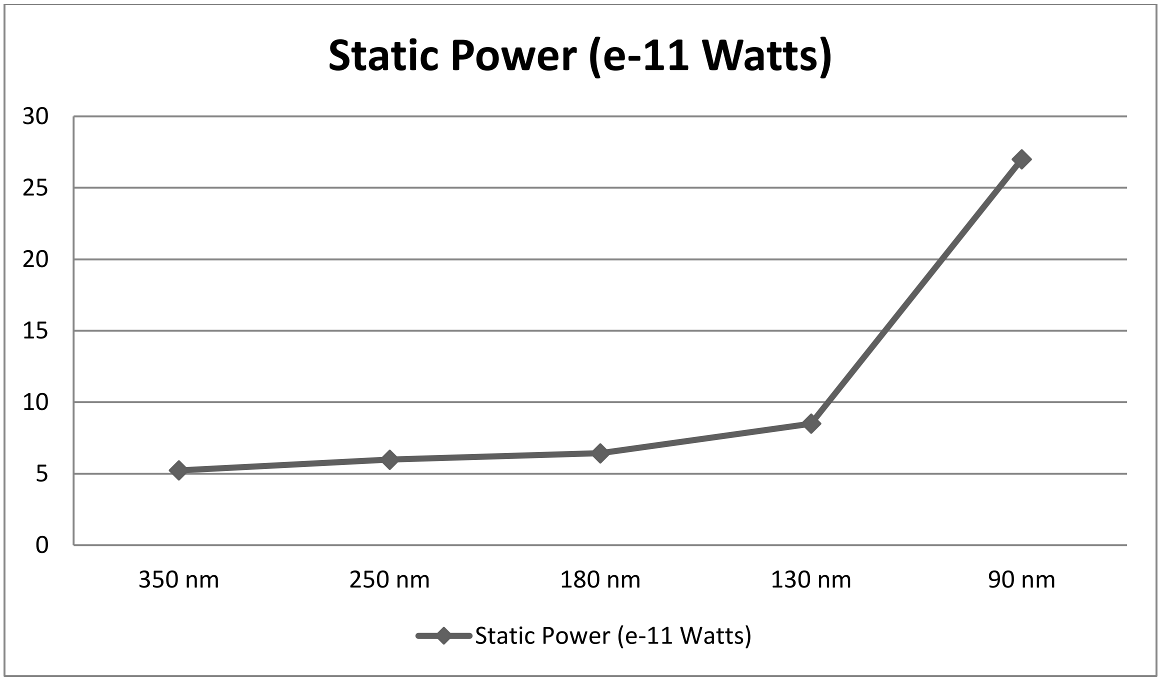

Fig. 2.6**:** Static Power of OR Gate using CVDTL in different NM

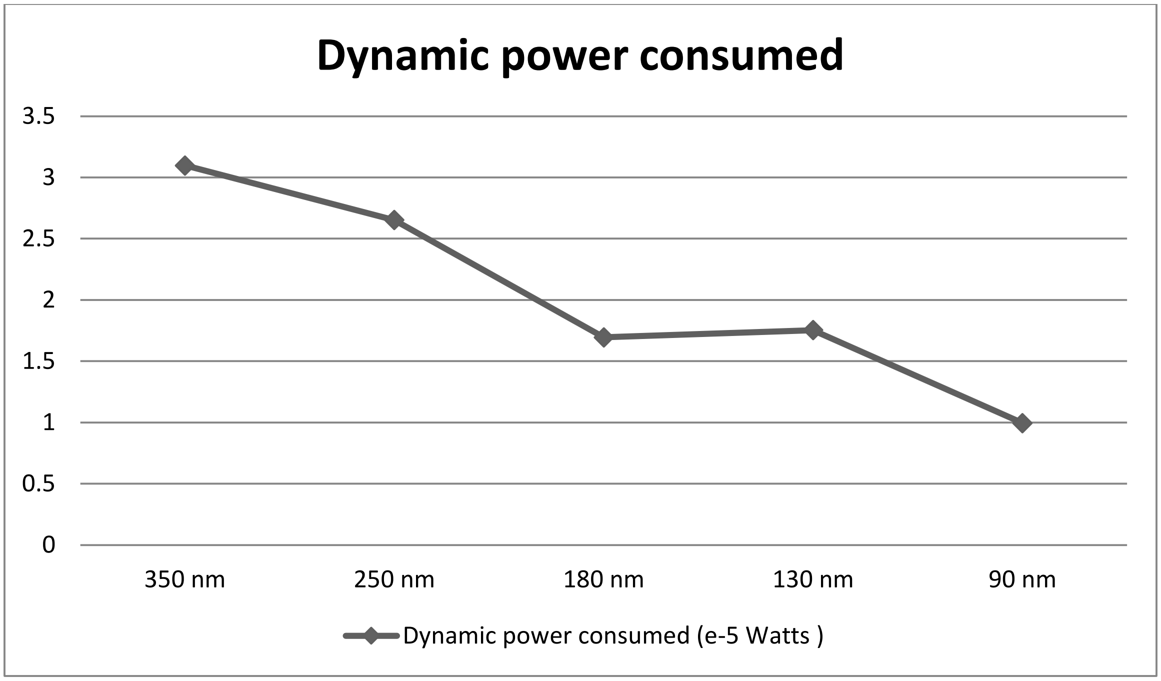

Fig. 2.7**:** Dynamic Power of OR Gate using CVDTL in different NM

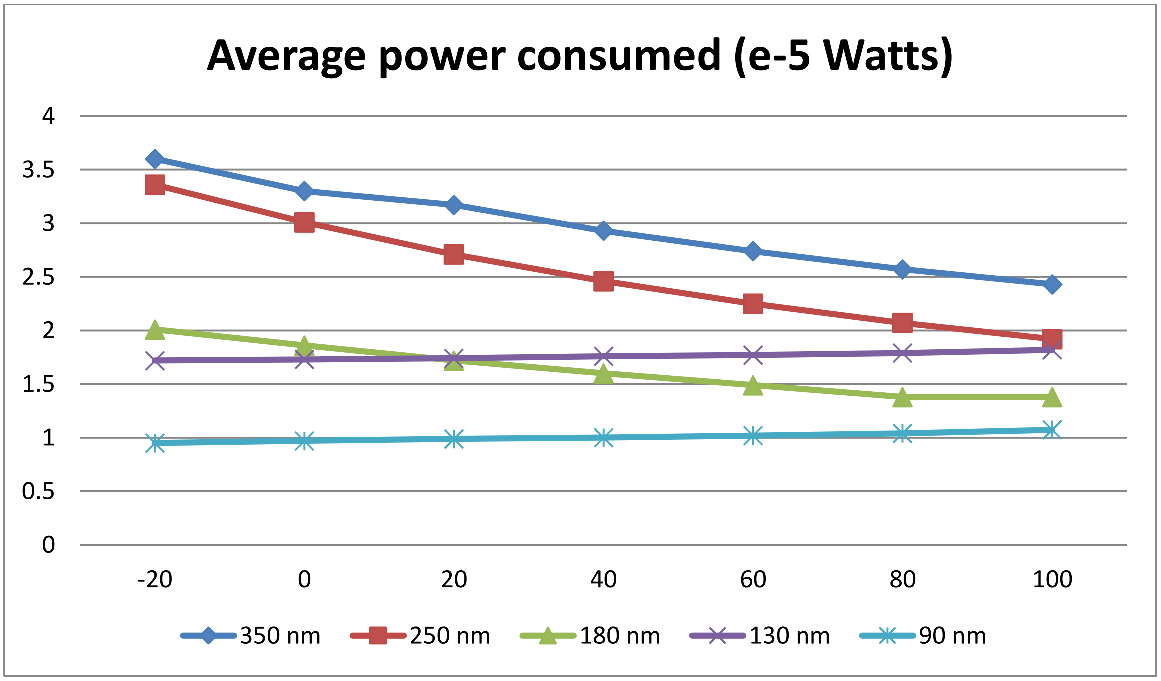

Fig. 2.8**:** Average power consumed by of OR Gate using CVDTL in different NM and in different Temperature

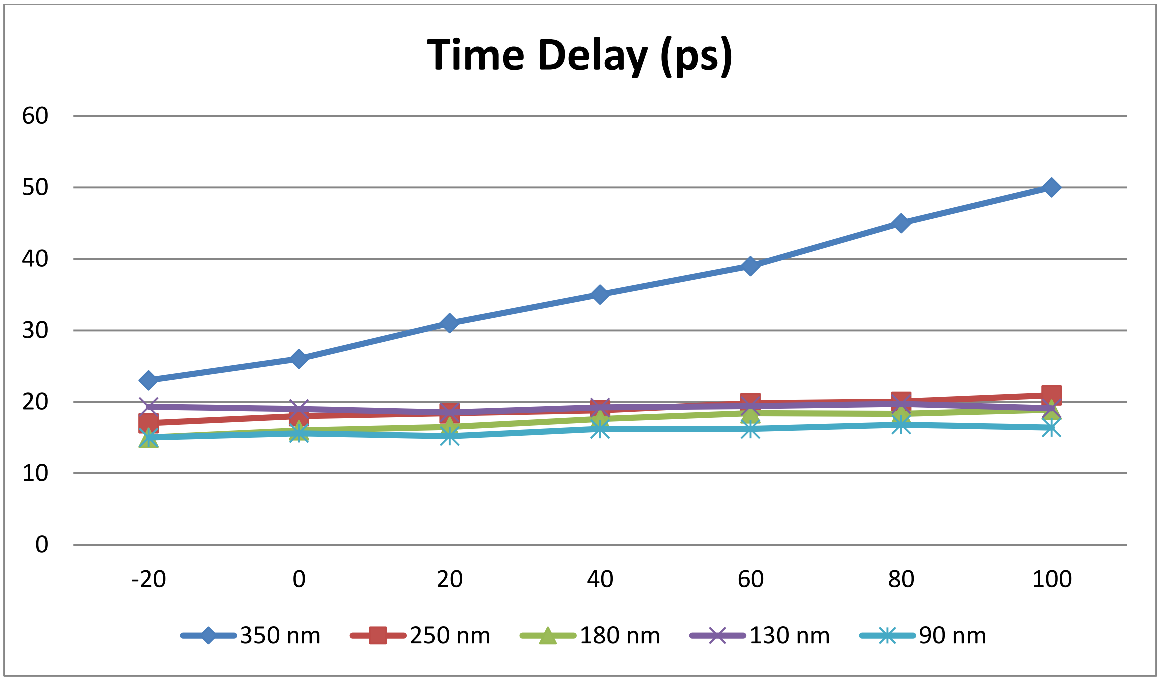

Fig. 2.9**:** Time Delay of OR Gate using CVDTL in different NM and in different Temperature

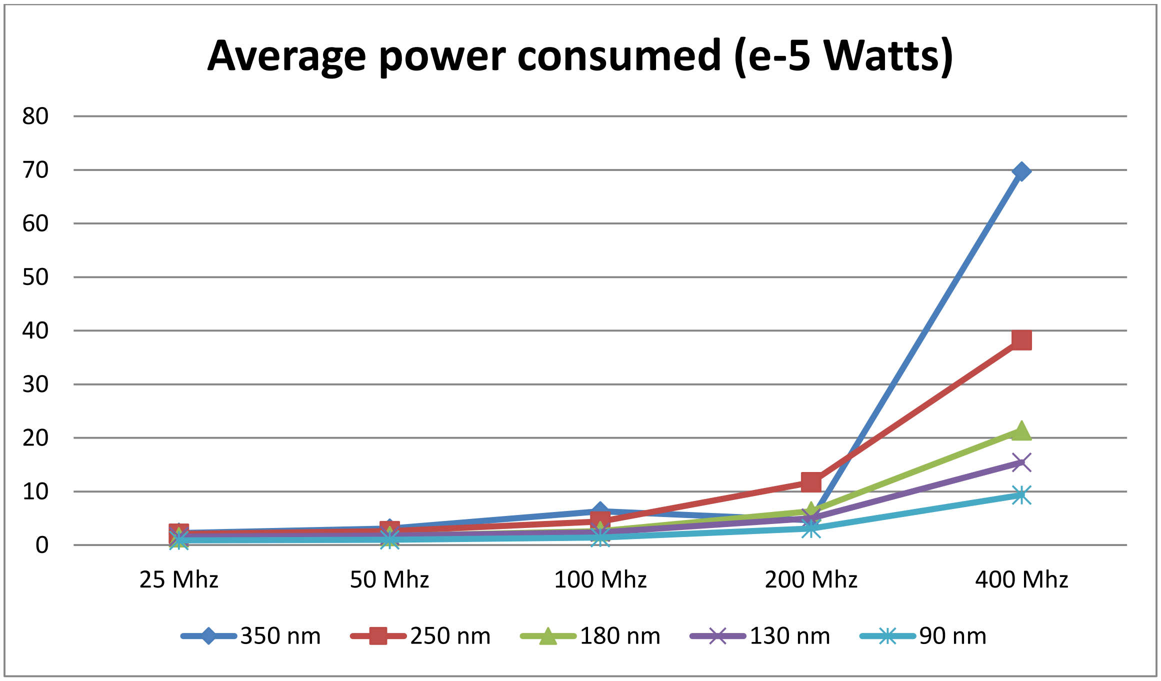

Fig. 2.10**:** Average power consumed by of OR Gate using CVDTL in different NM and in different Frequency

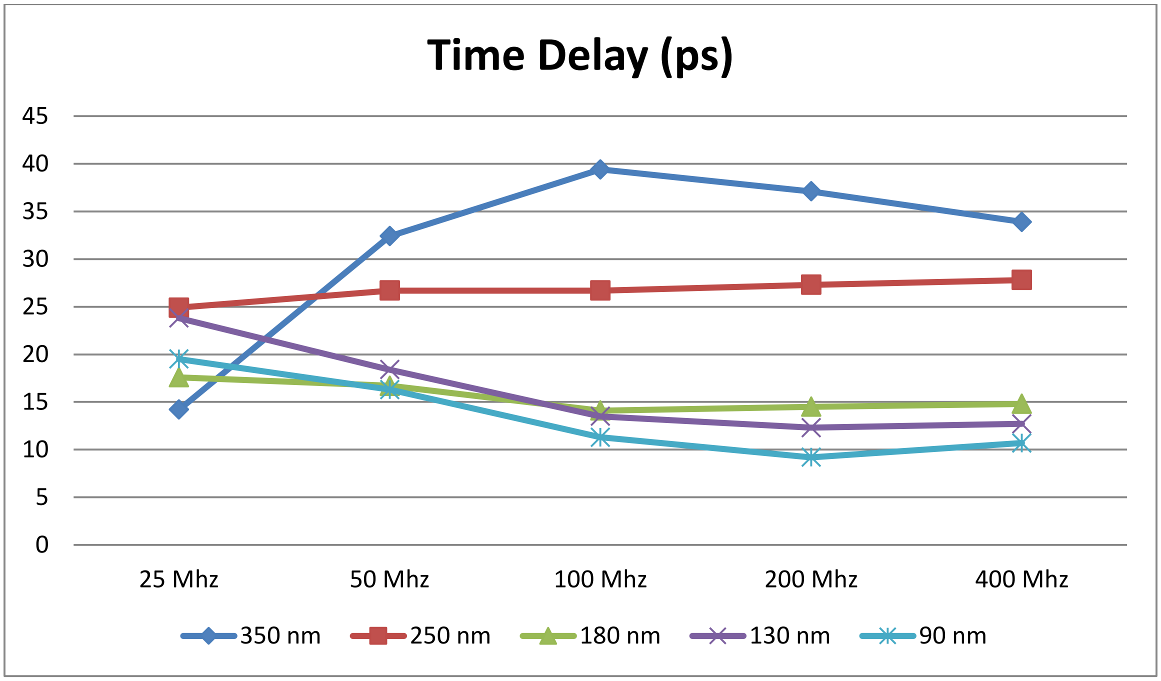

Fig. 2.11**:** Time Delay of OR Gate using CVDTL in different NM and in different Frequency

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

***** Top Level *****
CC1 Input_1 N_1  1p $ $x=-200 $y=800 $w=600 $h=400
CC2 Input_2 N_1  1p $ $x=-200 $y=-300 $w=600 $h=400
MNMOS_1 N_2 N_1 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1400 $y=-200
$w=400 $h=600
MNMOS_2 Output N_2 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=2900 $y=-
200 $w=400 $h=600
MPMOS_1 N_2 N_1 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1400
$y=900 $w=400 $h=600
MPMOS_2 Output N_2 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=2900
$y=900 $w=400 $h=600
VV3 Vdd Gnd  DC 2.5 $ $x=-2500 $y=1300 $w=400 $h=600
VV1 Input_1 Gnd  BIT({0011} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-1100 $y=500 $w=400
$h=600
```

```
VV2 Input_2 Gnd  BIT({0101} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-800 $y=-600 $w=400
$h=600
.PRINT   V(Input_1) $ $x=-950 $y=1550 $w=300 $h=1500 $r=270
.PRINT   V(Input_2) $ $x=-2450 $y=-150 $w=1500 $h=300 $r=180
.PRINT   V(Output) $ $x=4450 $y=250 $w=1500 $h=300

********* Simulation Settings - Analysis Section *********
.tran 0.1n 40n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

2.2. ANALYSIS OF AND GATE USING CVDT

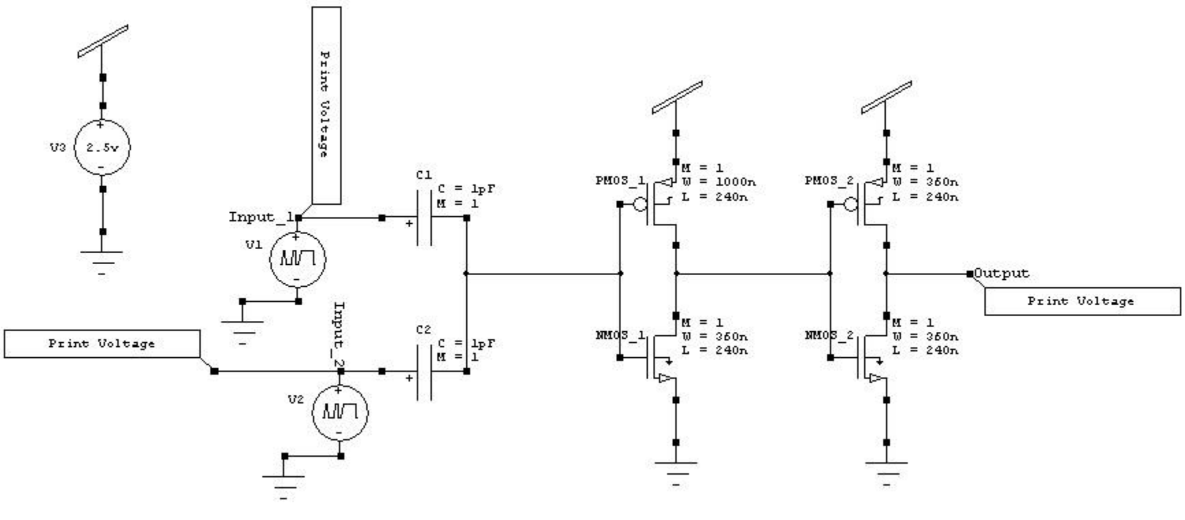

Fig. 2.11**:** AND gate circuit diagram using CVDTL

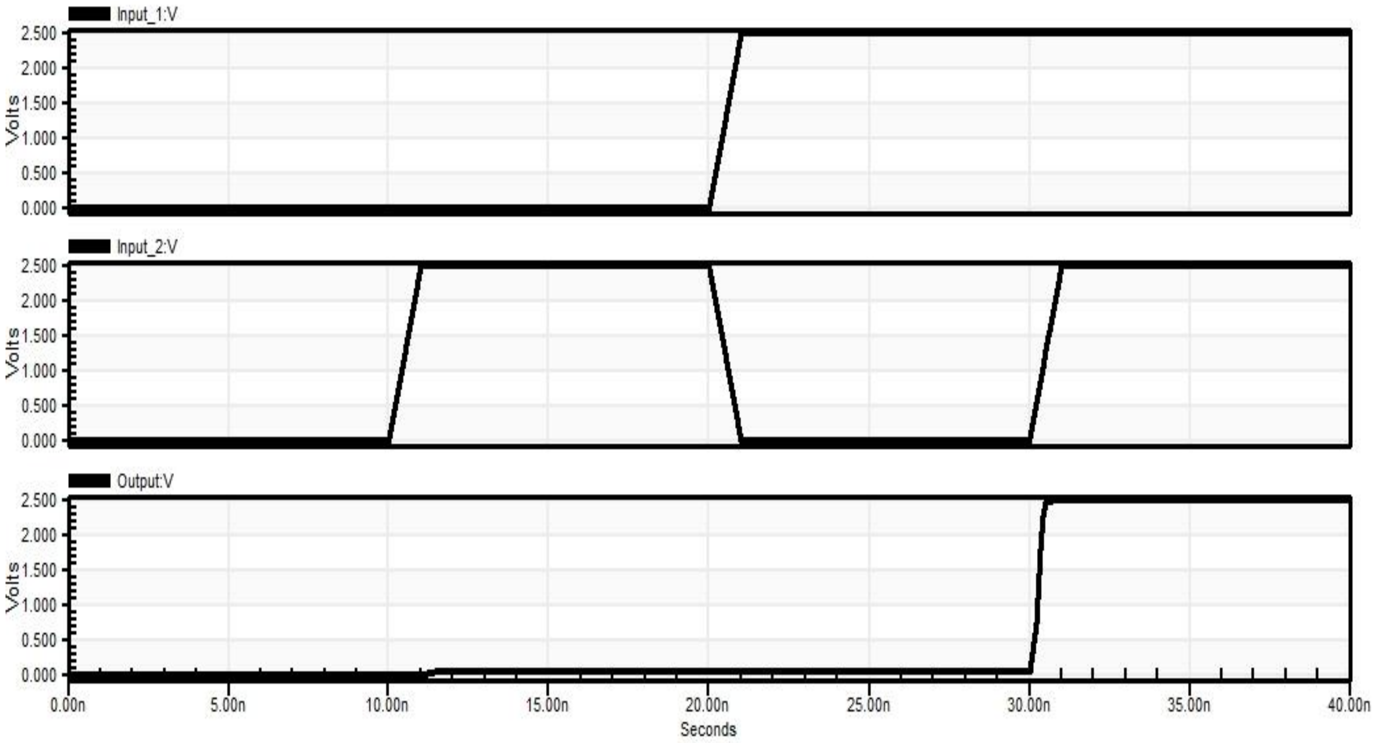

Fig. 2.12**:** AND gate output waveform using CVDTL

The above wave form shows, if any input is zero then output will be zero.

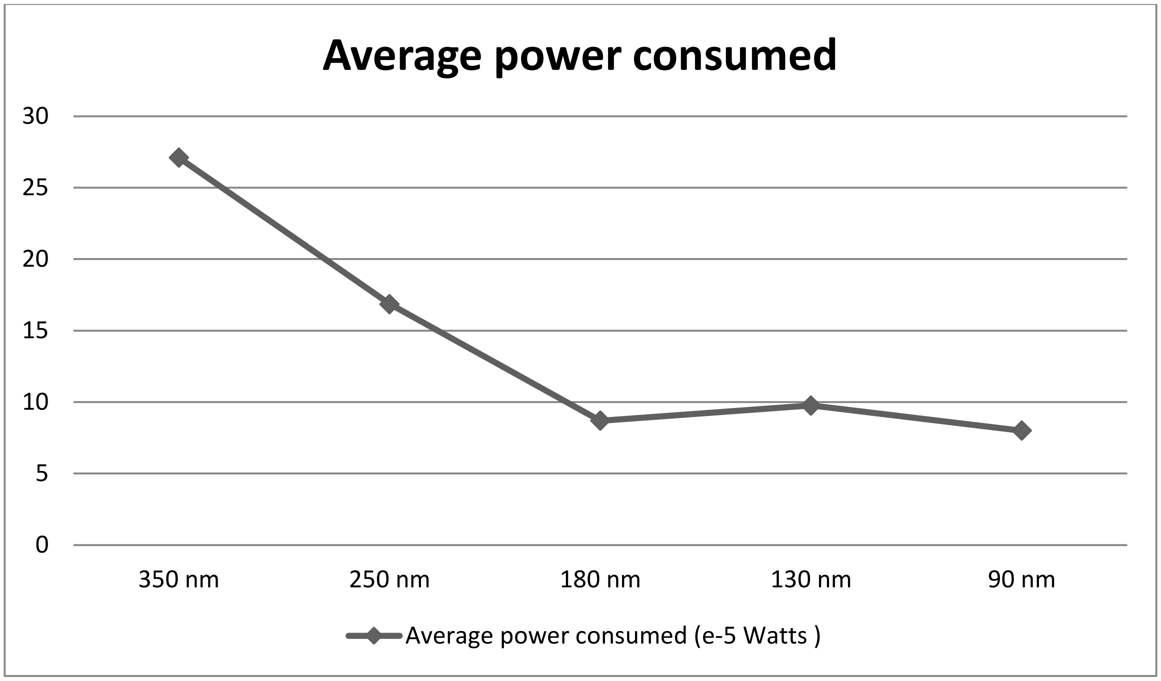

Fig. 2.13**:** Average power consumed by AND Gate using CVDTL in different NM

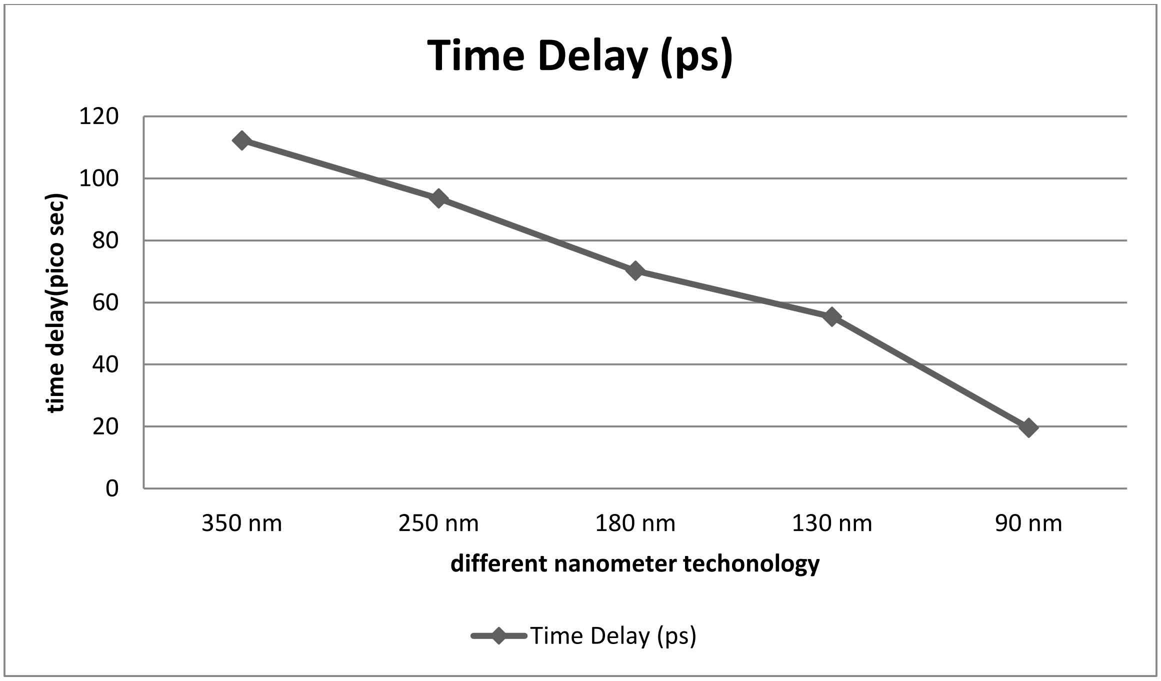

Fig. 2.14**:** Time delay of AND Gate using CVDTL in different NM

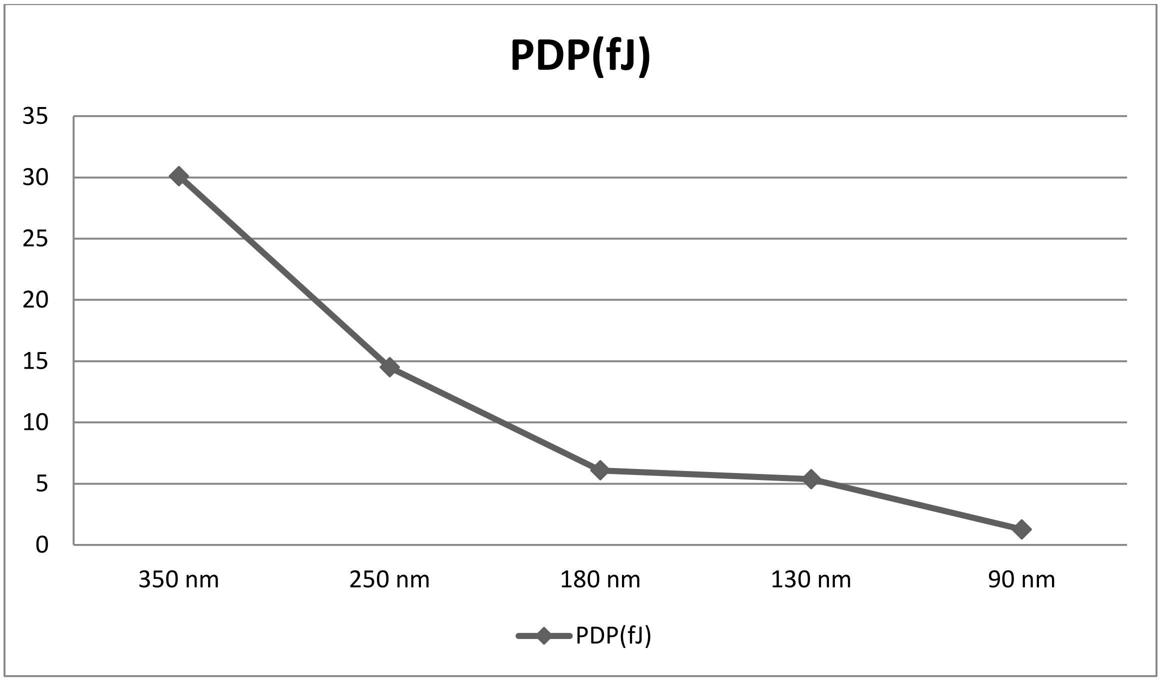

Fig. 2.15**:** Power Delay Product of AND Gate using CVDTL in different NM

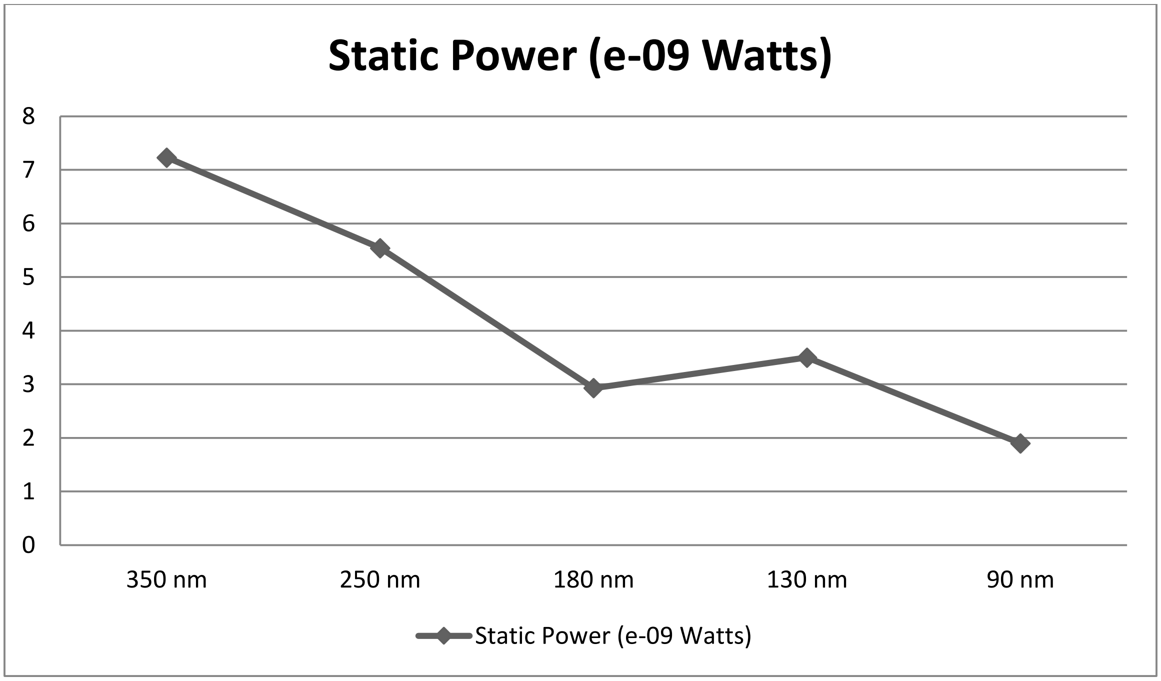

Fig. 2.16**:** Static Power of AND Gate using CVDTL in different NM

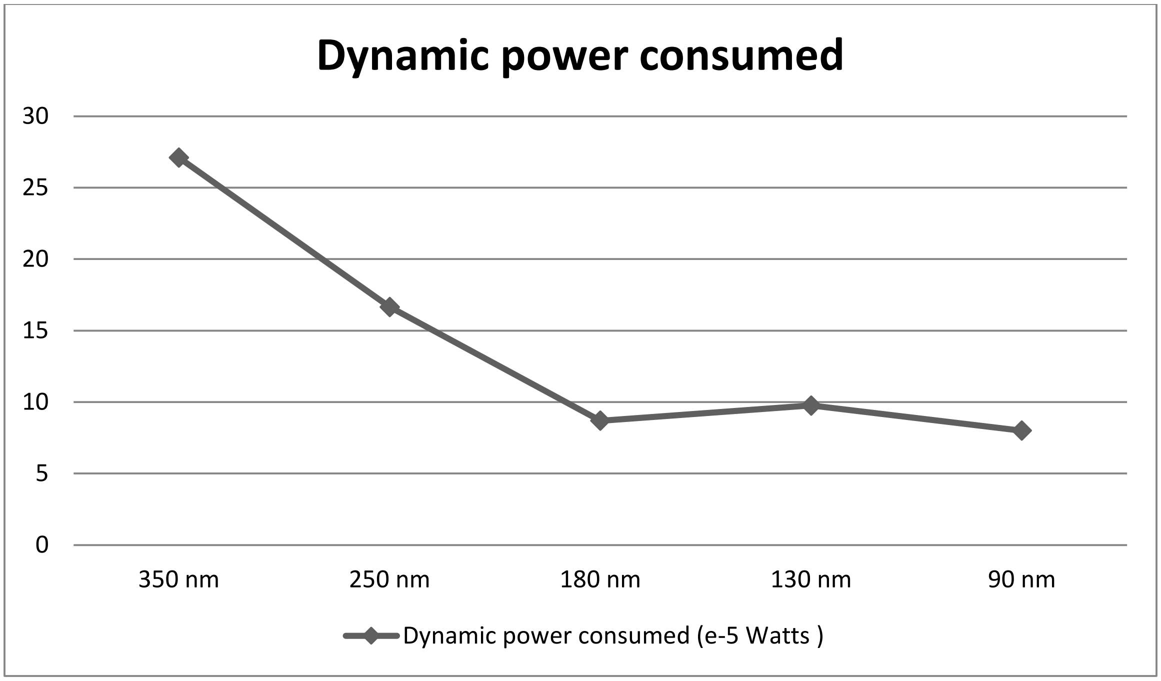

Fig. 2.17**:** Dynamic Power of AND Gate using CVDTL in different NM

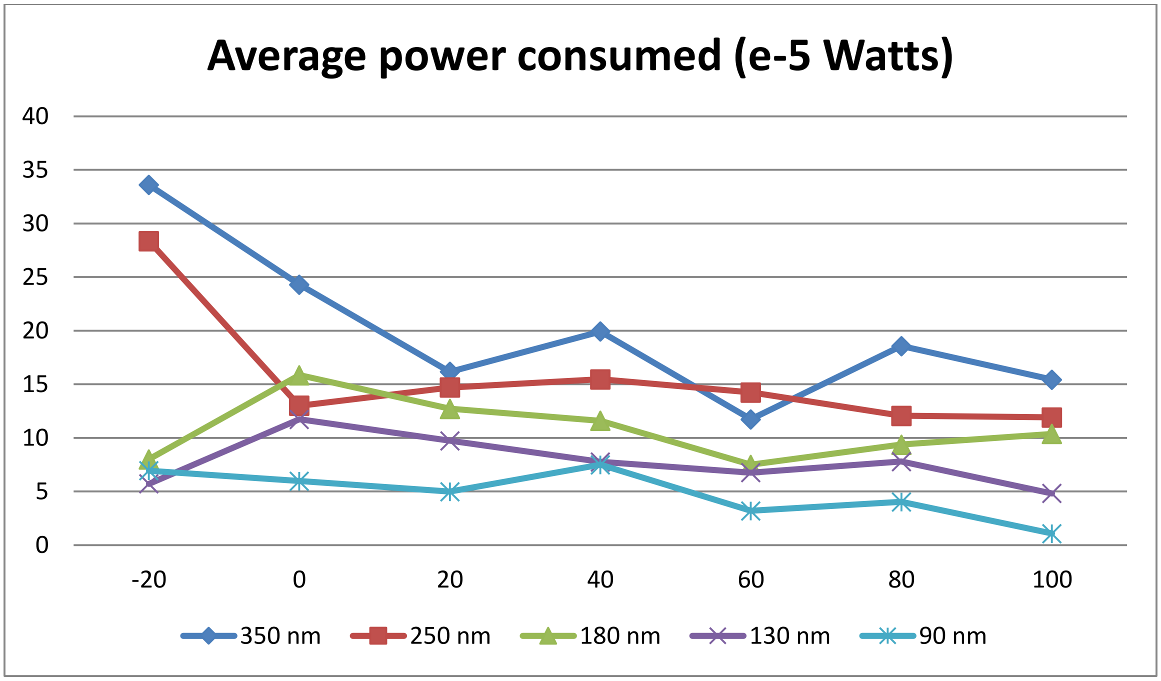

Fig. 2.18**:** Average power consumed by of AND Gate using CVDTL in different NM and in different Temperature

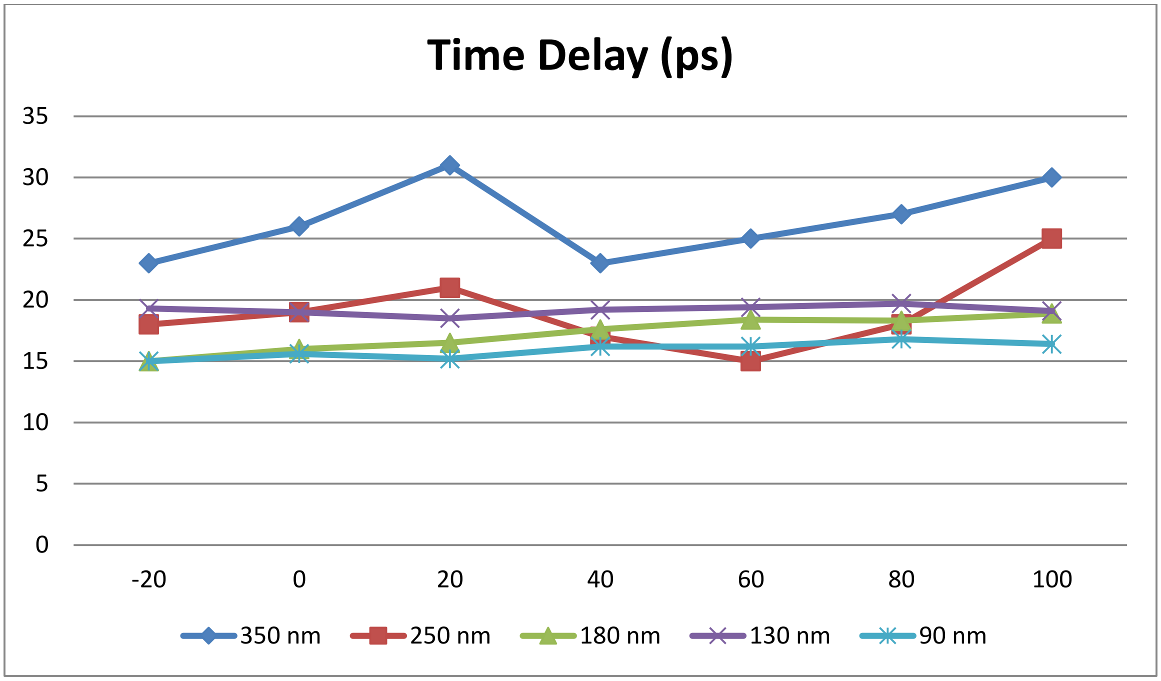

Fig. 2.19**:** Time Delay of AND Gate using CVDTL in different NM and in different Temperature

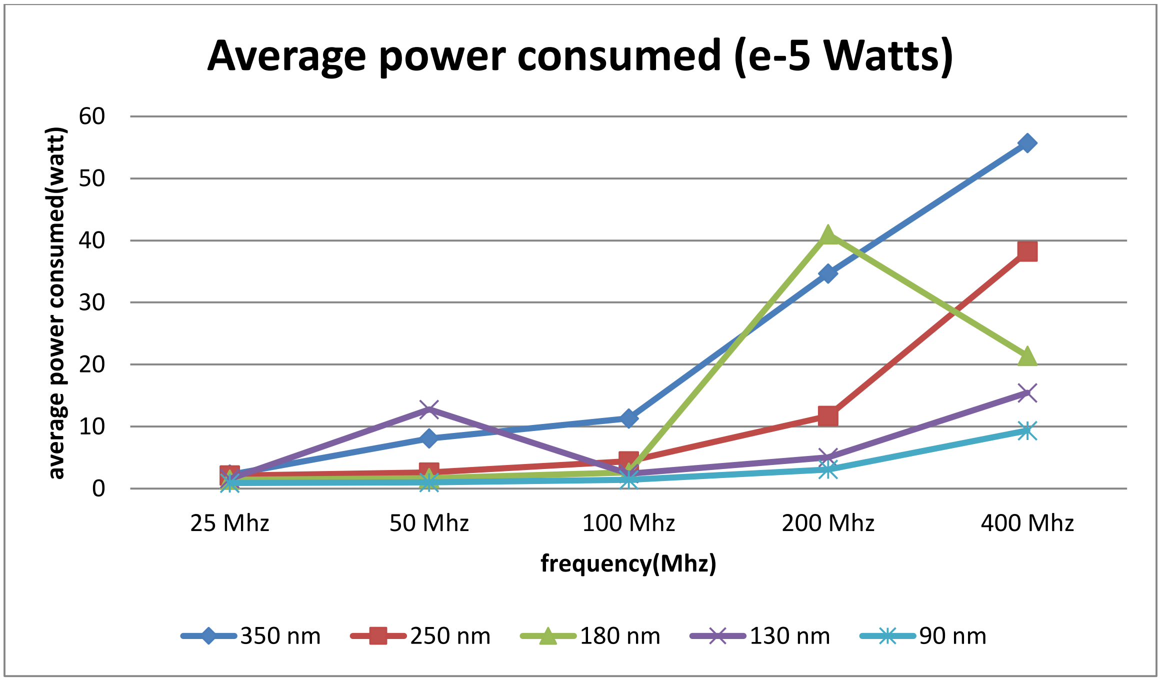

Fig. 2.20**:** Average power consumed by of AND Gate using CVDTL in different NM and in different Frequency

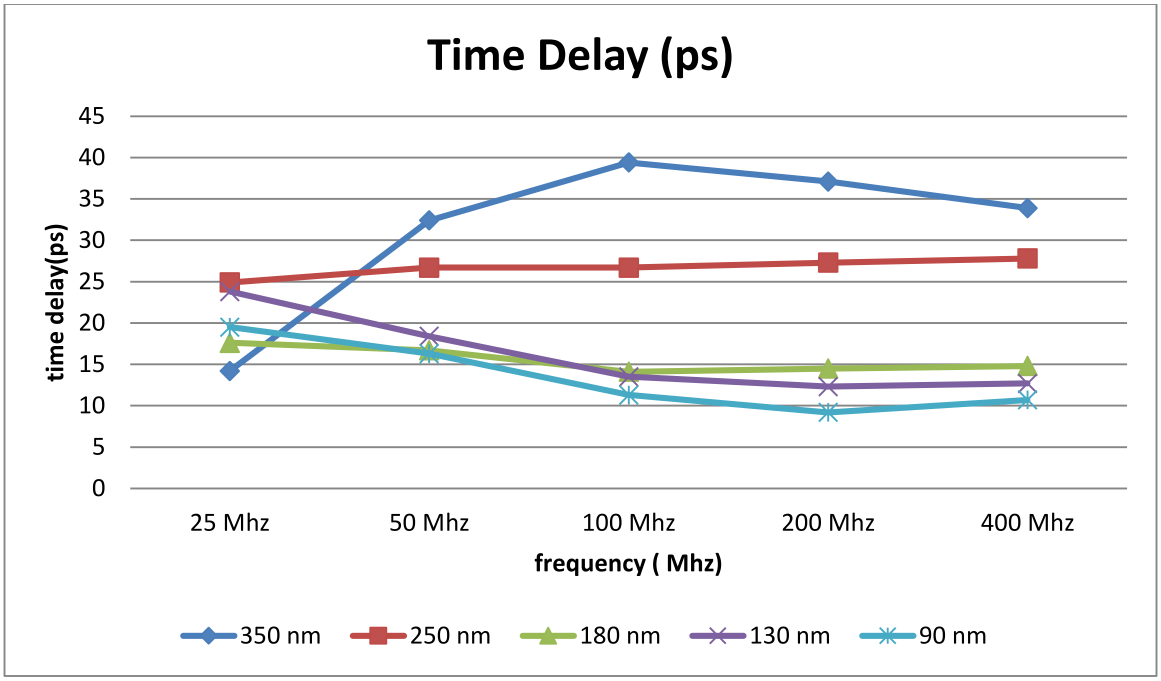

Fig. 2.21**:** Time Delay of AND Gate using CVDTL in different NM and in different Frequency

T-Spice:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra Krishnapura\TSMC 250nm CMOS.md"

***** Top Level *****
CC1 Input_1 N_1  1p $ $x=300 $y=1000 $w=600 $h=400
CC2 Input_2 N_1  1p $ $x=300 $y=-100 $w=600 $h=400
MNMOS_1 N_2 N_1 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1900 $y=0 $w=400 $h=600
MNMOS_2 Output N_2 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3400 $y=0 $w=400 $h=600
MPMOS_1 N_2 N_1 Vdd Vdd PMOS W=1u L=240n AS=900f PS=3.8u AD=900f PD=3.8u $ $x=1900 $y=1100 $w=400 $h=600
MPMOS_2 Output N_2 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3400 $y=1100 $w=400 $h=600
VV3 Vdd Gnd  DC 2.5 $ $x=-2000 $y=1500 $w=400 $h=600
VV1 Input_1 Gnd  BIT({0011} PW=1.25n ON=2.5 RT=100p FT=100p LT=1.25n HT=1.25n) $ $x=-600 $y=700 $w=400 $h=600
VV2 Input_2 Gnd  BIT({0101} PW=1.25n ON=2.5 RT=100p FT=100p LT=1.25n HT=1.25n) $ $x=-300 $y=-400 $w=400 $h=600
.PRINT   V(Input_1) $ $x=-450 $y=1750 $w=300 $h=1500 $r=270
.PRINT   V(Input_2) $ $x=-1950 $y=50 $w=1500 $h=300 $r=180
```

```
.PRINT   V(Output) $ $x=4950 $y=450 $w=1500 $h=300

********* Simulation Settings - Analysis Section *********
.tran 0.1n 5n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

2.3 ANALYSIS OF XOR GATE USING CVDTL

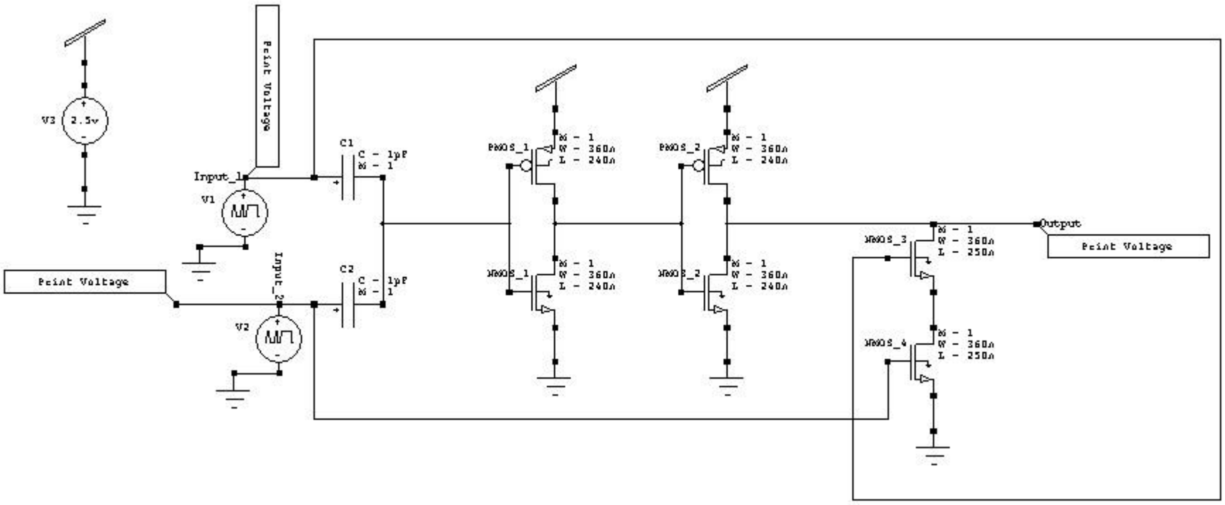

Fig. 2.22**:** XOR gate circuit diagram using CVDTL

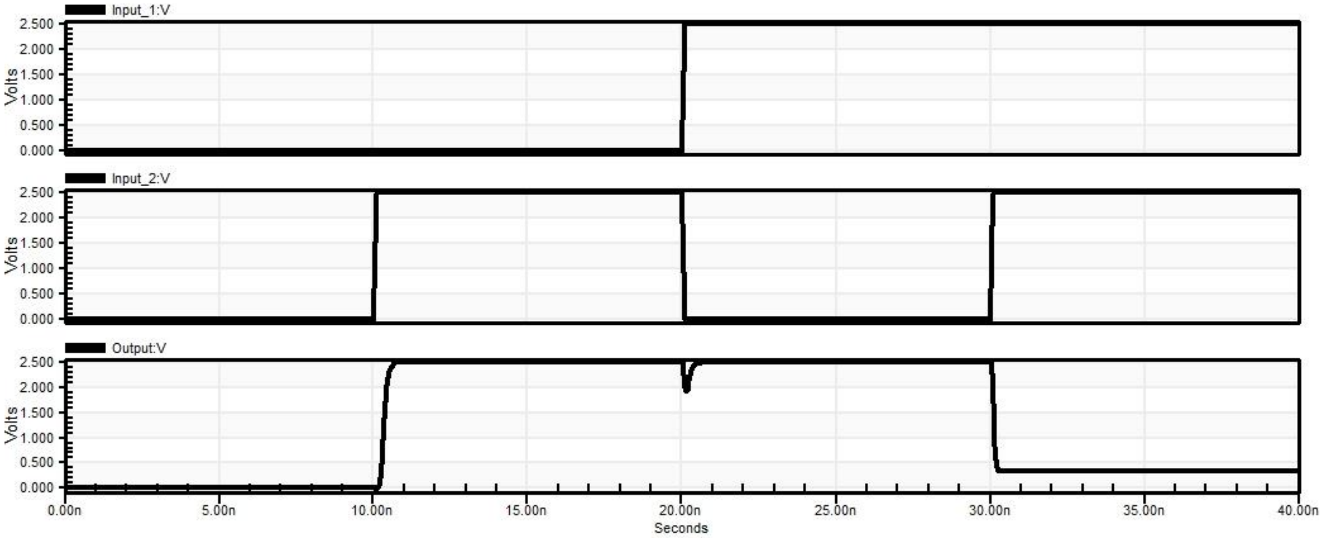

Fig. 2.23**:** XOR gate output waveform using CVDTL

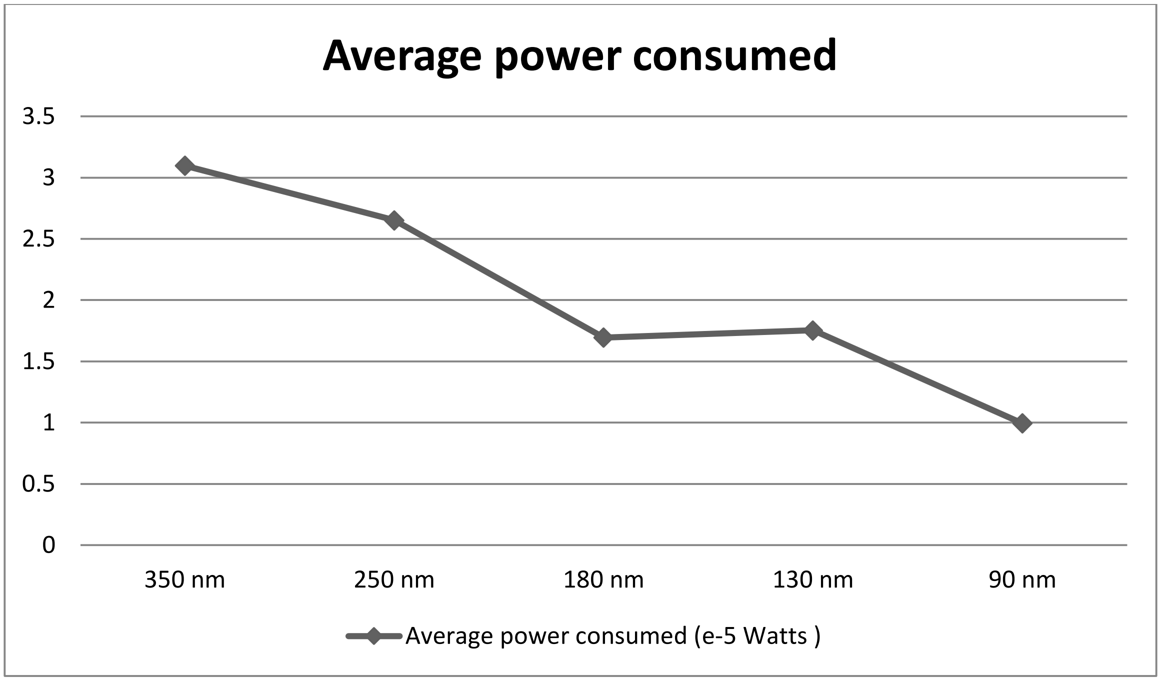

Fig. 2.24**:** Average power consumed by XOR Gate using CVDTL in different NM

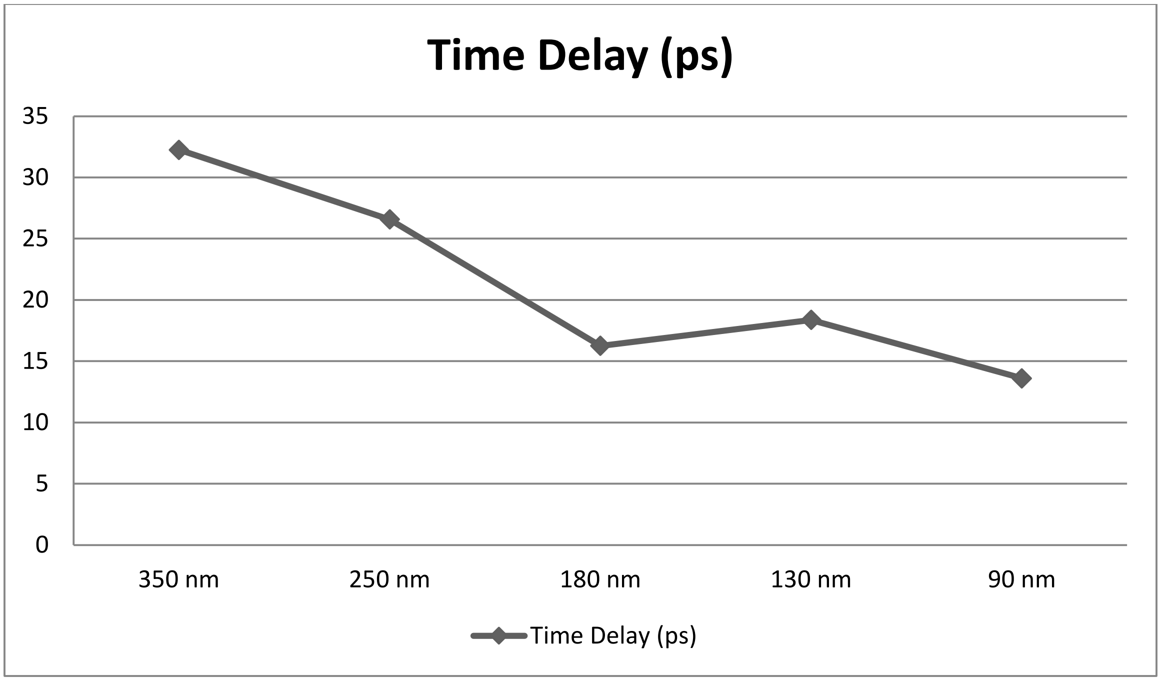

Fig. 2.25**:** Time delay of XOR Gate using CVDTL in different NM

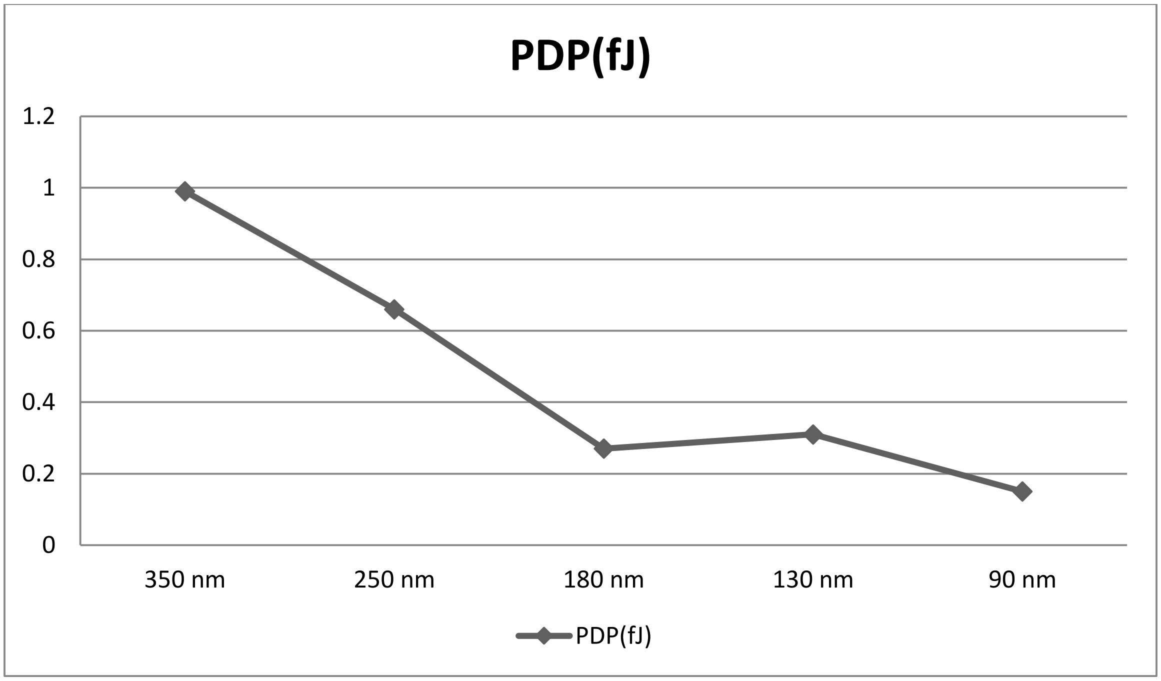

Fig. 2.26**:** Power Delay Product of XOR Gate using CVDTL in different NM

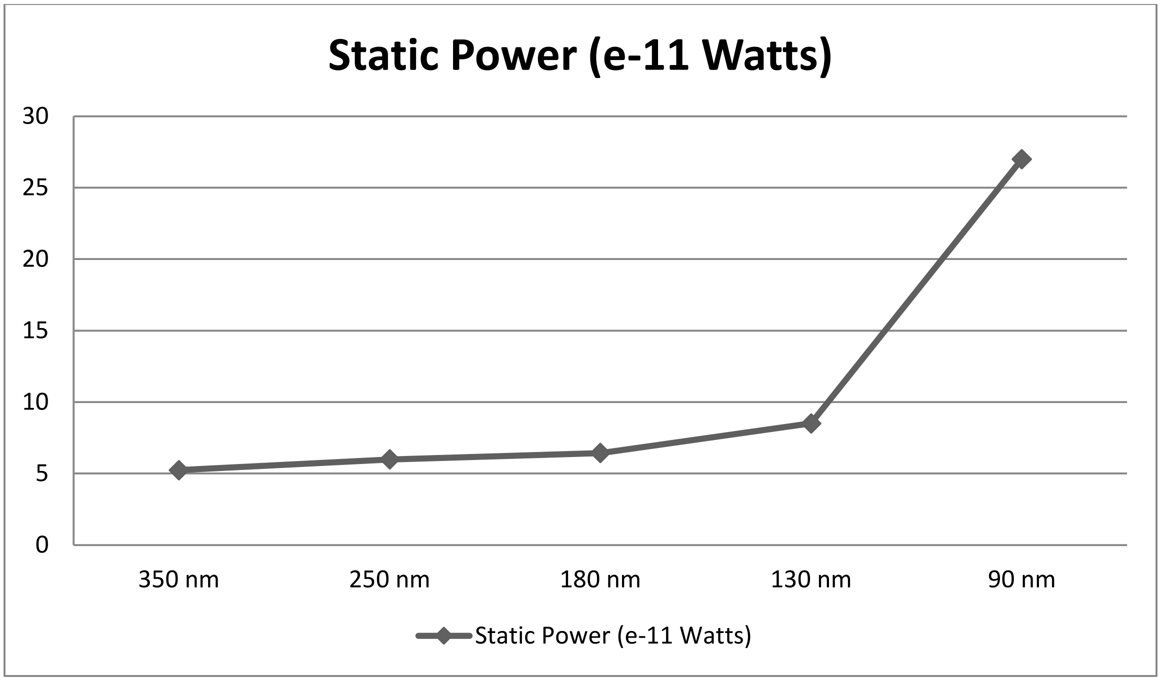

Fig. 2.27**:** Static Power of XOR Gate using CVDTL in different NM

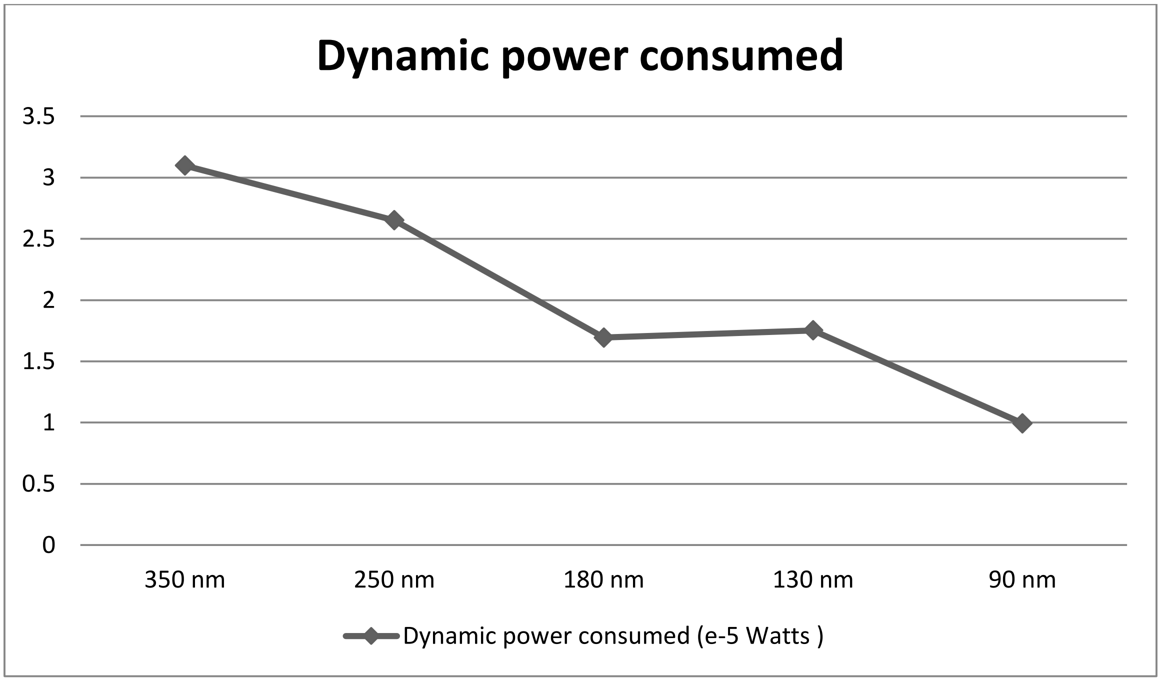

Fig. 2.28**:** Dynamic Power of XOR Gate using CVDTL in different NM

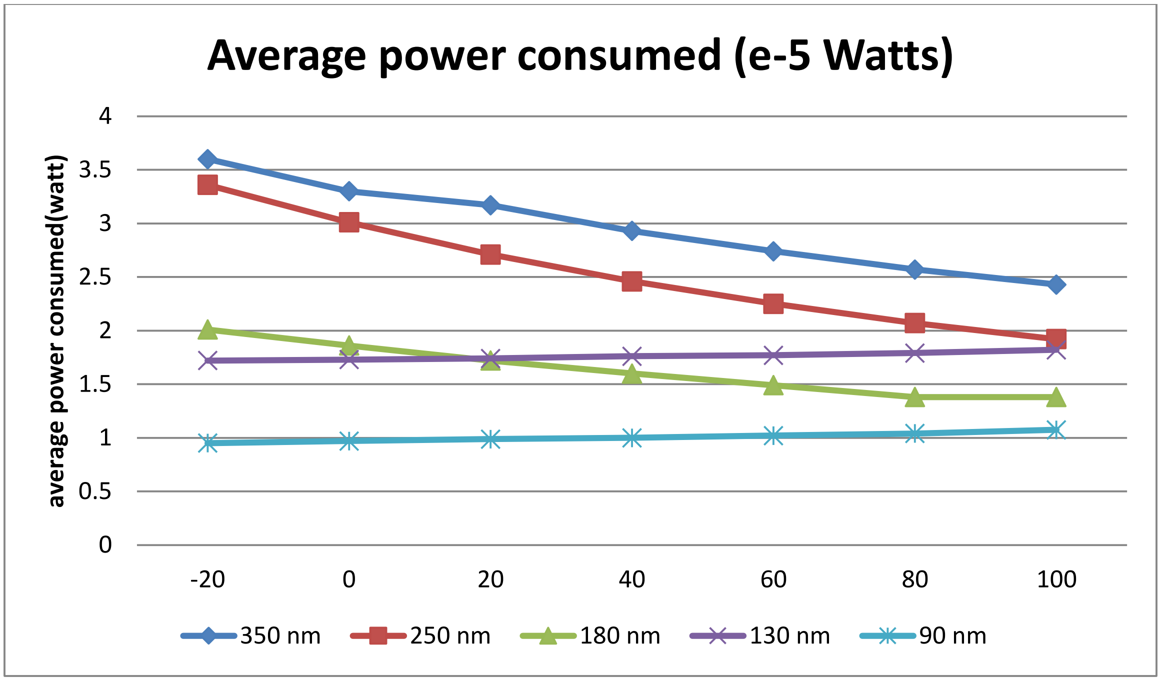

Fig. 2.29: Average power consumed by of XOR Gate using CVDTL in different NM and in different Temperature

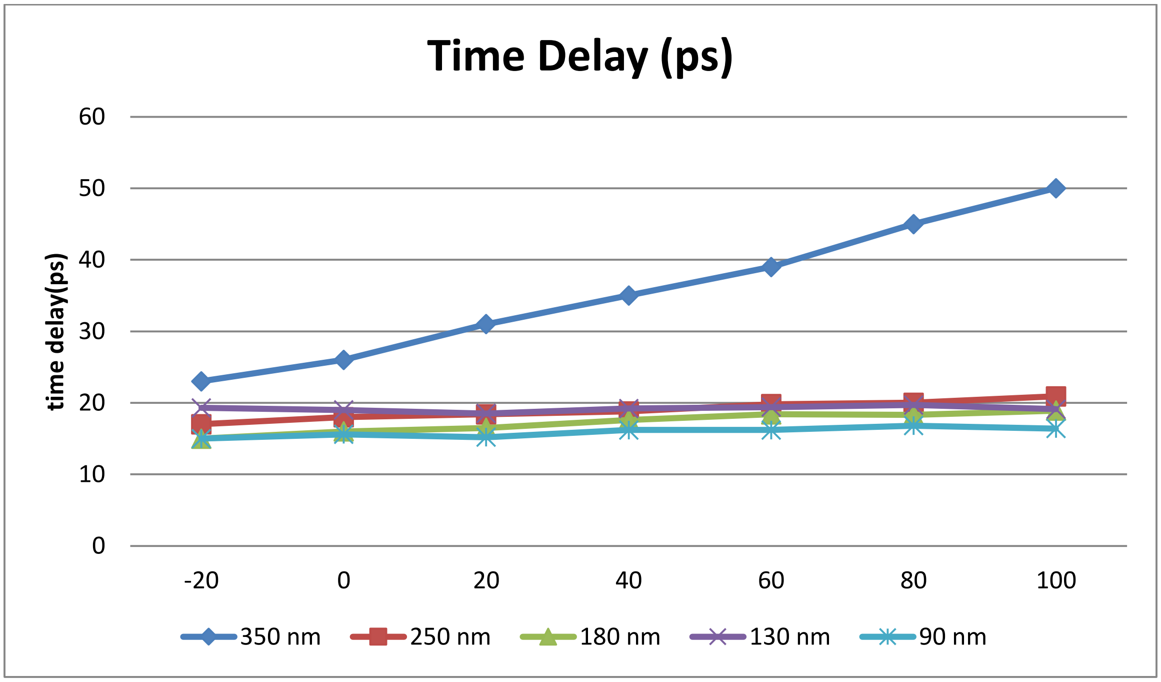

Fig. 2.30: Time Delay of XOR Gate using CVDTL in different NM and in different Temperature

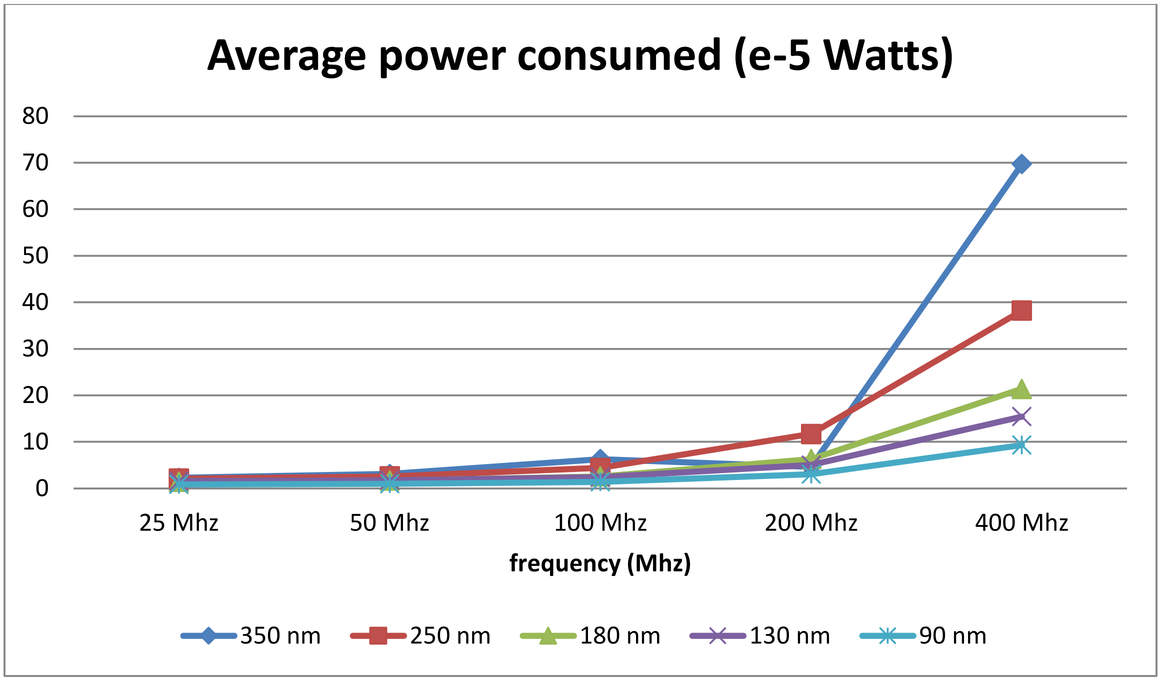

Fig. 2.31**:** Average power consumed by of XOR Gate using CVDTL in different NM and in different Frequency

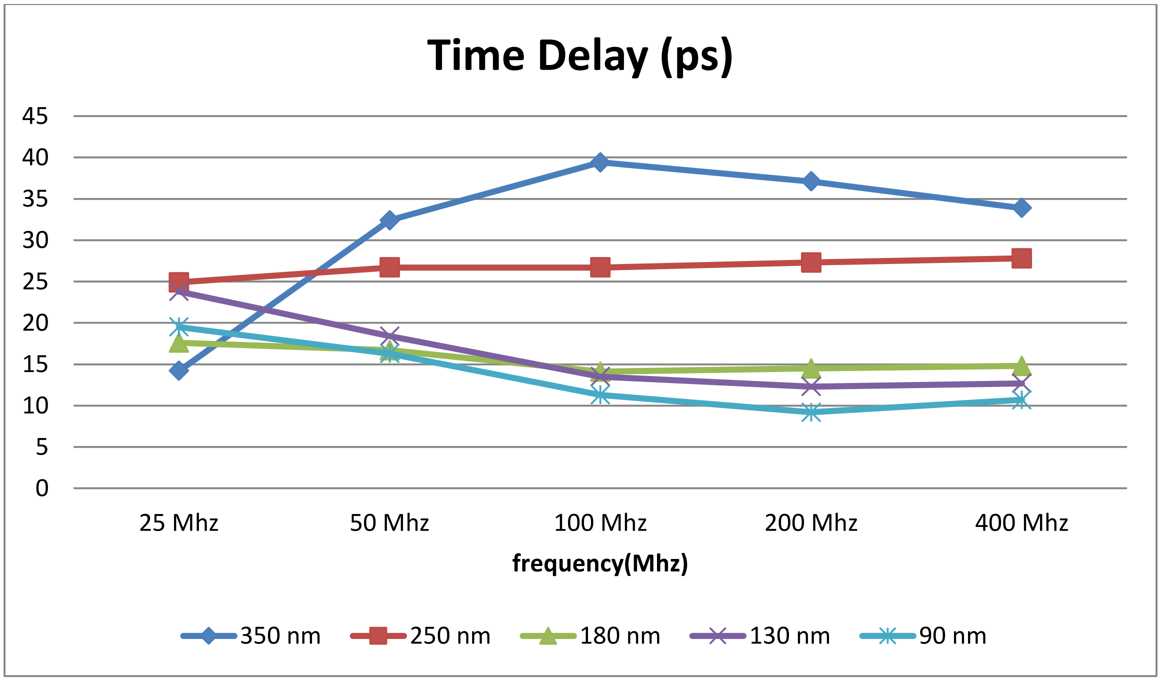

Fig. 2.32**:** Time Delay of XOR Gate using CVDTL in different NM and in different Frequency

T-Spice Code:

```
********* Simulation Settings - General Section *********
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

***** Top Level *****
CC1 Input_1 N_2  1p $ $x=800 $y=1400 $w=600 $h=400
CC2 Input_2 N_2  1p $ $x=800 $y=300 $w=600 $h=400
MNMOS_1 N_3 N_2 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=2400 $y=400
$w=400 $h=600
MNMOS_2 Output N_3 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3900
$y=400 $w=400 $h=600
MNMOS_3 Output Input_1 N_1 0 NMOS W=360n L=250n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5700
$y=700 $w=400 $h=600
MNMOS_4 N_1 Input_2 Gnd 0 NMOS W=360n L=250n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5700 $y=-
200 $w=400 $h=600
MPMOS_1 N_3 N_2 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=2400
$y=1500 $w=400 $h=600
MPMOS_2 Output N_3 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3900
$y=1500 $w=400 $h=600
VV3 Vdd Gnd  DC 2.5 $ $x=-1500 $y=1900 $w=400 $h=600
VV1 Input_1 Gnd  BIT({0011} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=-100 $y=1100
$w=400 $h=600
VV2 Input_2 Gnd  BIT({0101} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=200 $y=0 $w=400
$h=600
.PRINT   V(Input_1) $ $x=50 $y=2150 $w=300 $h=1500 $r=270
.PRINT   V(Input_2) $ $x=-1450 $y=450 $w=1500 $h=300 $r=180
.PRINT   V(Output) $ $x=7550 $y=850 $w=1500 $h=300

********* Simulation Settings - Analysis Section *********
.tran 0.1n 40n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

2.4. ANALYSIS OF MAJORITY GATE USING CVDTL

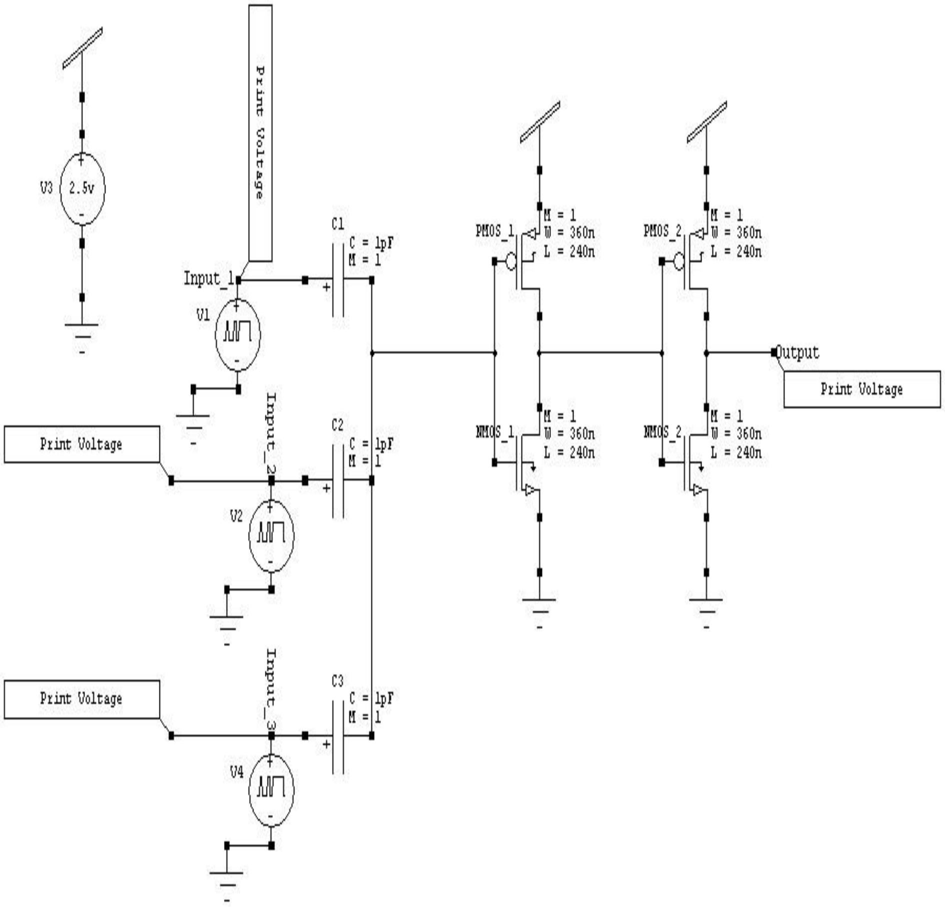

Fig. 2.34**:** Majority gate circuit diagram using CVDTL

It is a logic gate that implements the majority function. It is a device that out puts HIGH when the majority of its inputs are HIGH, otherwise it outputs a LOW.

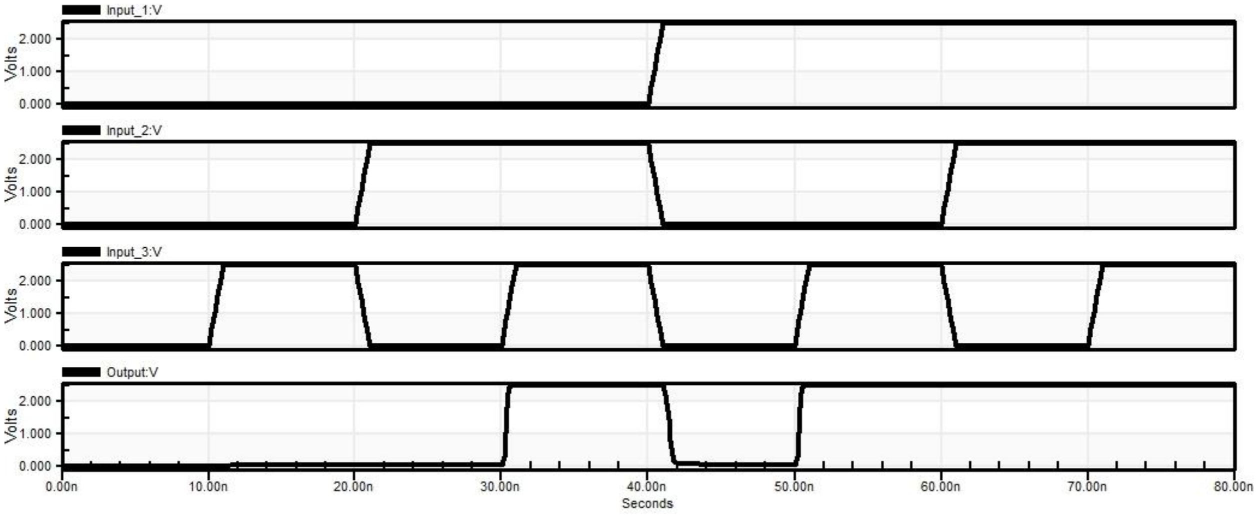

Fig. 2.35**:** Majority gate output waveform using CVDTL

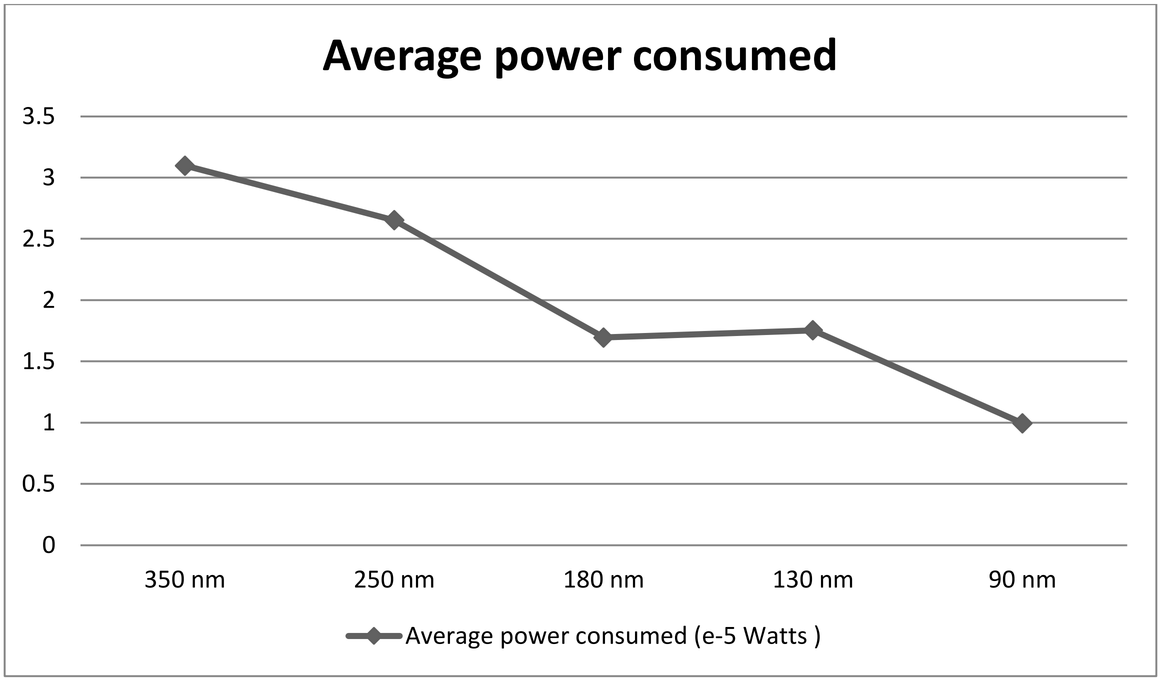

Fig. 2.36**:** Average power consumed by Majority Gate using CVDTL in different NM

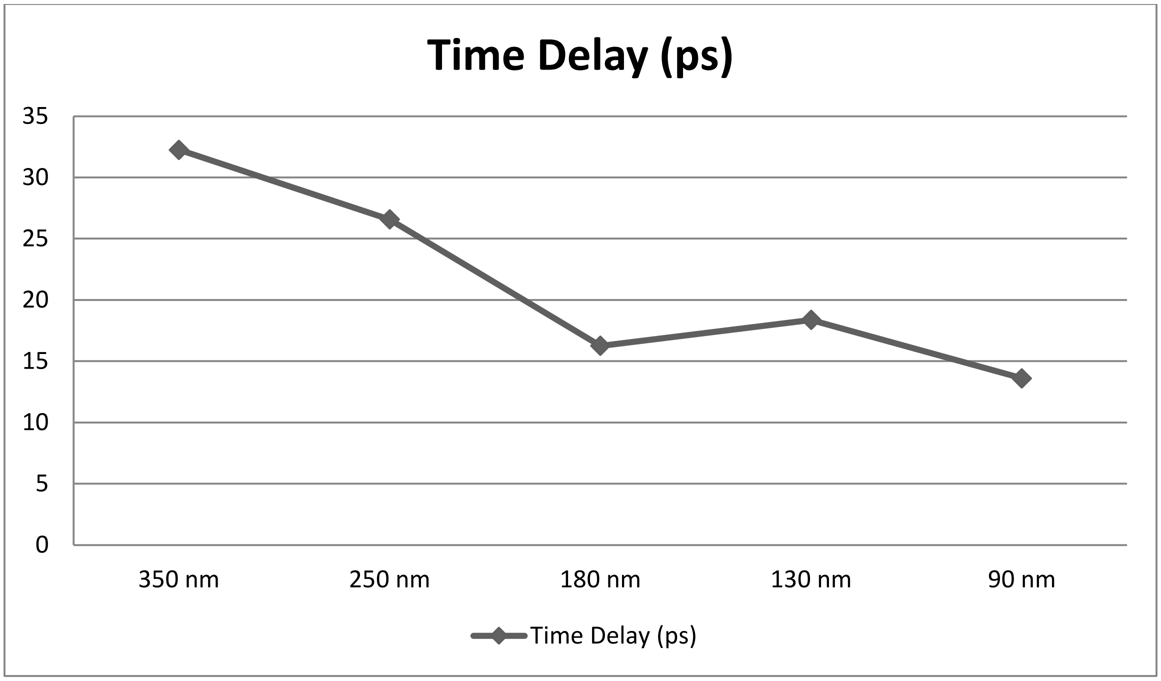

Fig. 2.37**:** Time delay of Majority Gate using CVDTL in different NM

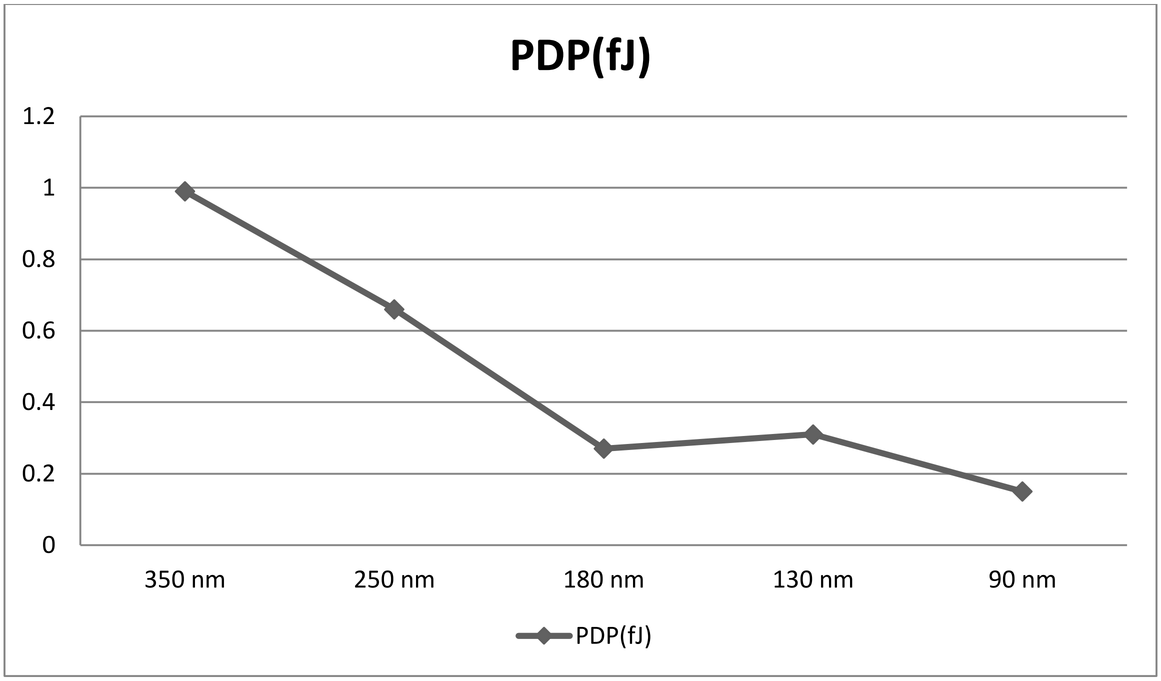

Fig. 2.38**:** Power Delay Product of Majority Gate using CVDTL in different NM

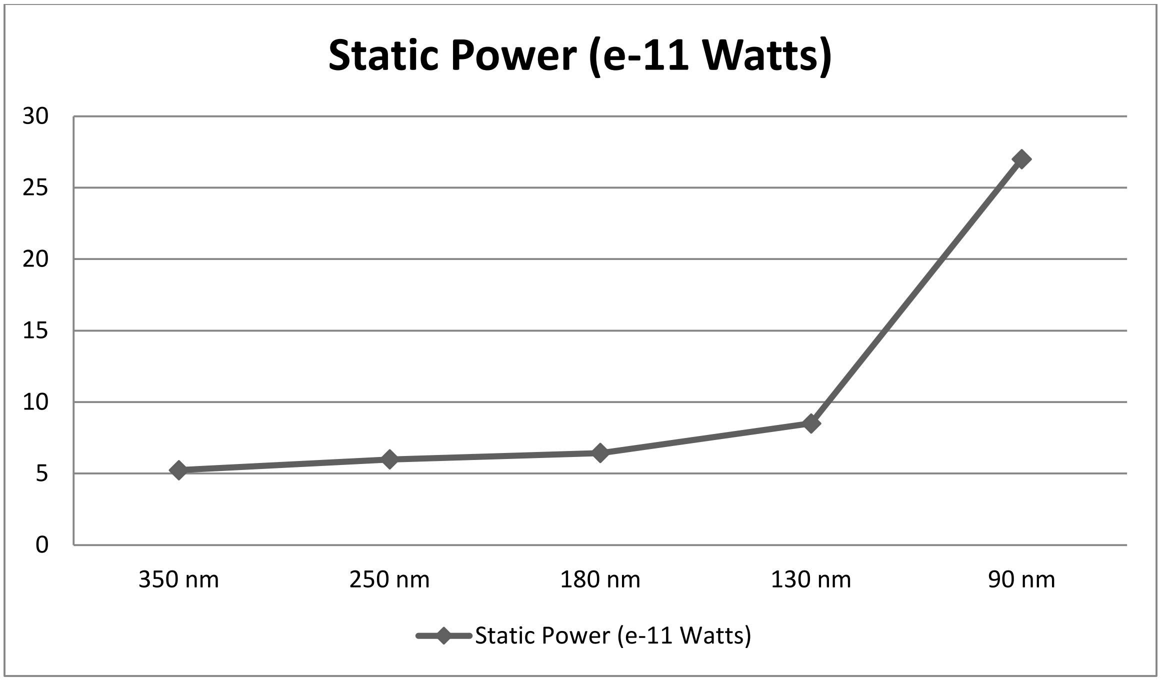

Fig. 2.39**:** Static Power of Majority Gate using CVDTL in different NM

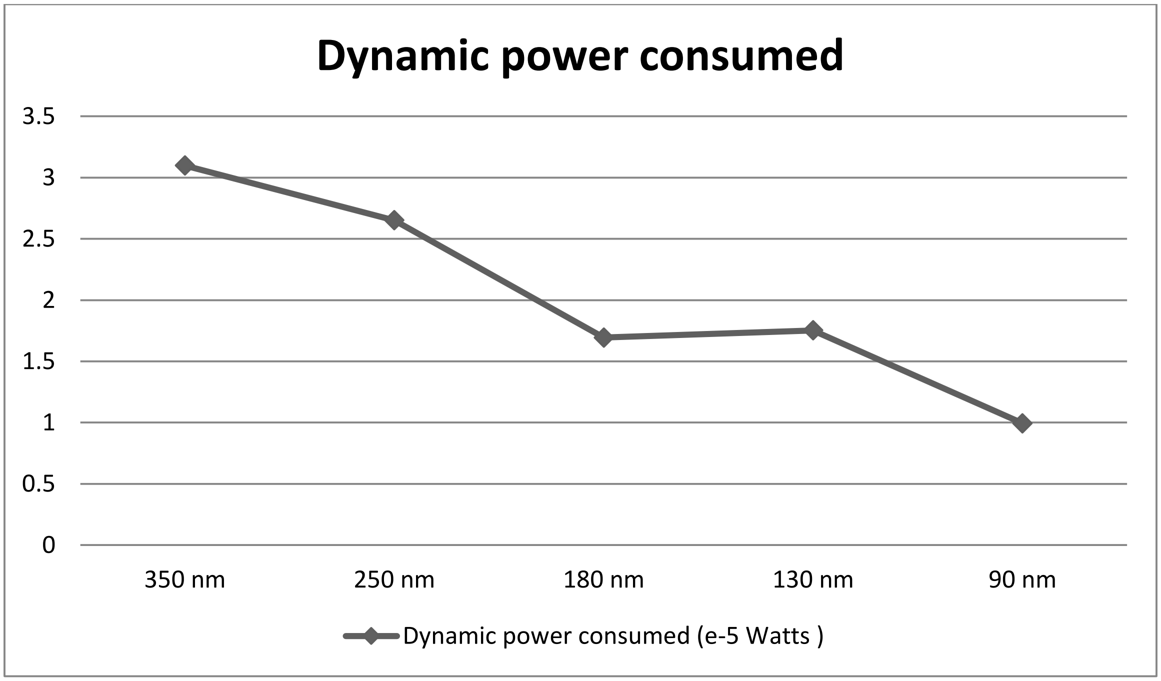

Fig. 2.40**:** Dynamic Power of Majority Gate using CVDTL in different NM

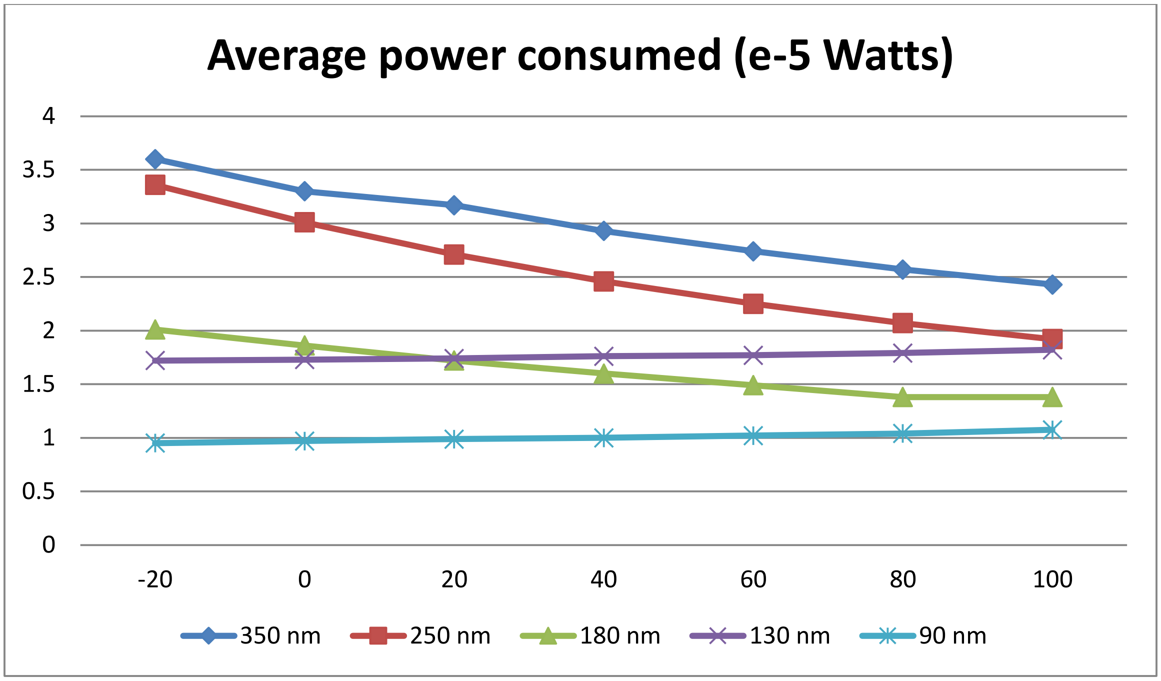

Fig. 2.41**:** Average power consumed by of Majority Gate using CVDTL in different NM and in different Temperature

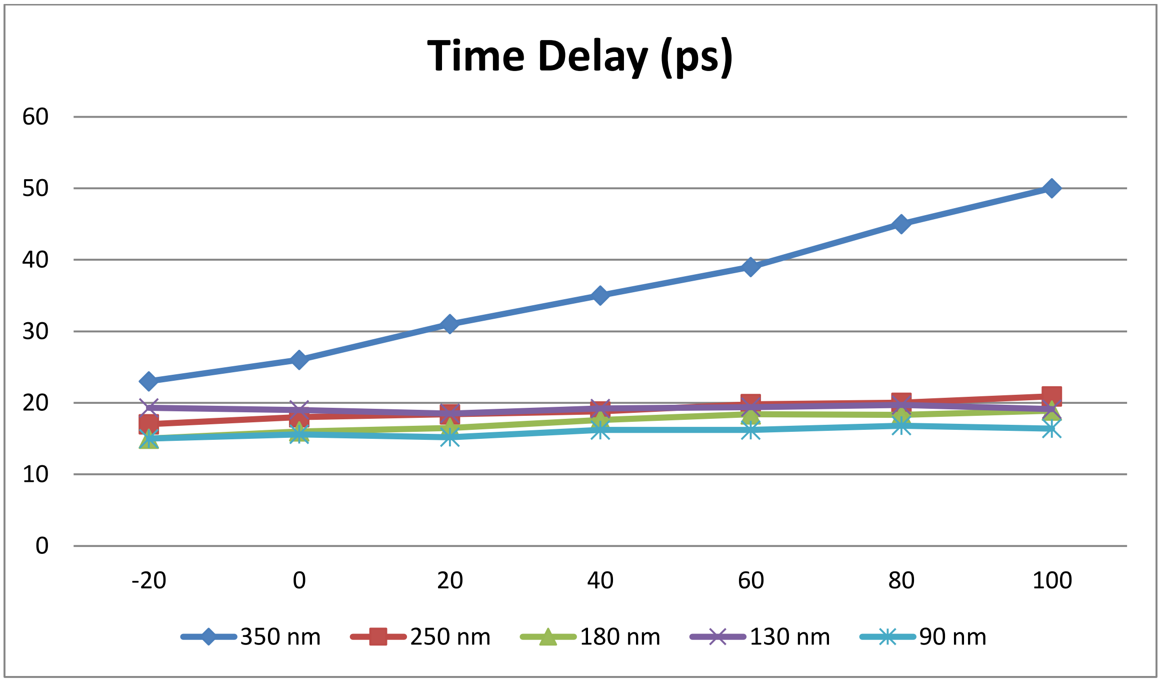

Fig. 2.42**:** Time Delay of Majority Gate using CVDTL in different NM and in different Temperature

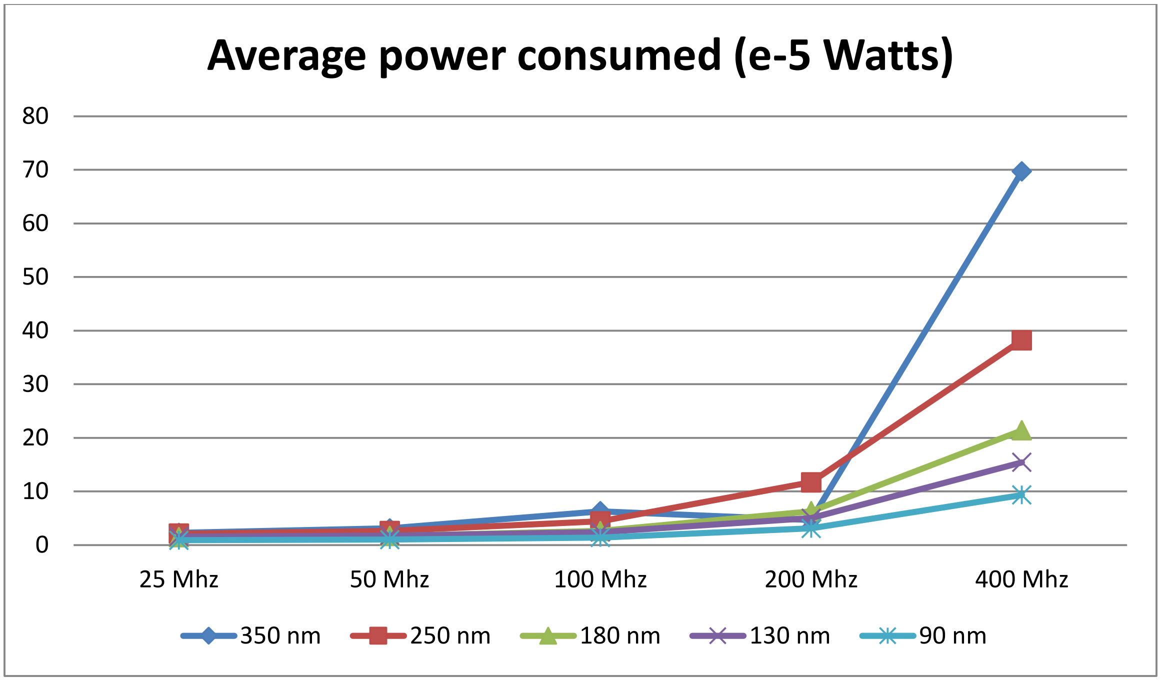

Fig. 2.43**:** Average power consumed by of Majority Gate using CVDTL in different NM and in different Frequency

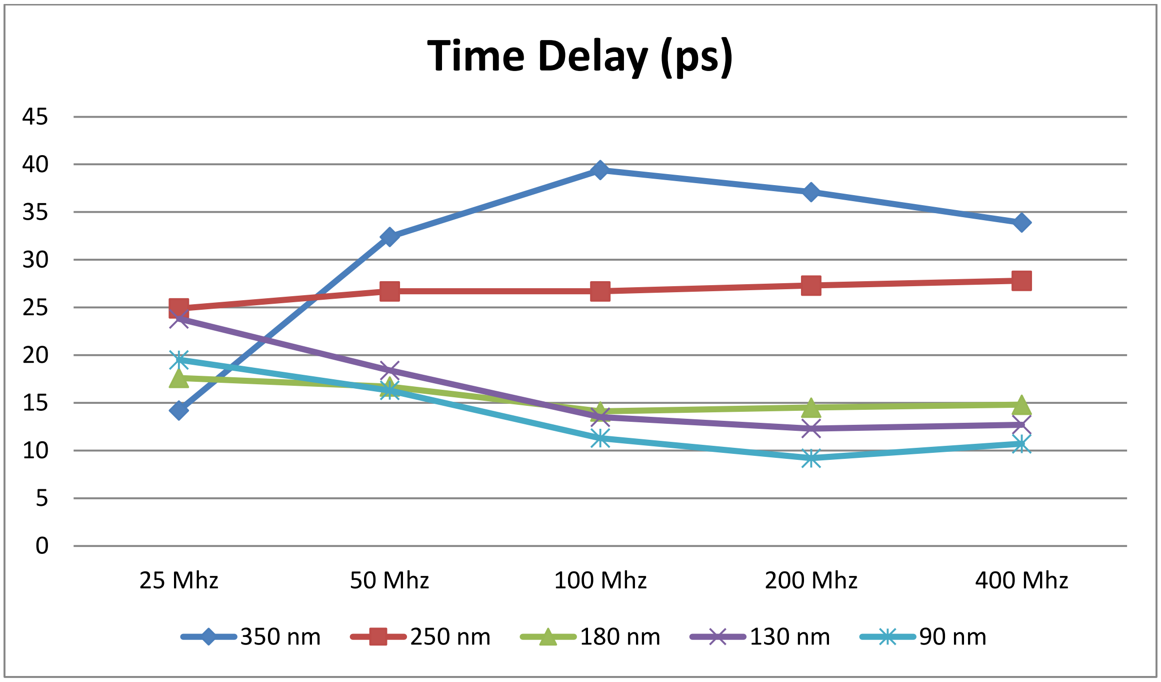

Fig. 2.44**:** Time Delay of Majority Gate using CVDTL in different NM and in different Frequency

T-Spice Code:

```
********* Simulation Settings - General Section *********
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra Krishnapura\TSMC 250nm CMOS.md"

***** Top Level *****
CC1 Input_1 N_1  1p $ $x=2300 $y=900 $w=600 $h=400
CC2 Input_2 N_1  1p $ $x=2300 $y=-200 $w=600 $h=400
CC3 Input_3 N_1  1p $ $x=2300 $y=-1600 $w=600 $h=400
MNMOS_1 N_2 N_1 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3900 $y=-100 $w=400 $h=600
MNMOS_2 Output N_2 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5400 $y=-100 $w=400 $h=600
MPMOS_1 N_2 N_1 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3900 $y=1000 $w=400 $h=600
MPMOS_2 Output N_2 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5400 $y=1000 $w=400 $h=600
VV3 Vdd Gnd  DC 2.5 $ $x=0 $y=1400 $w=400 $h=600
VV1 Input_1 Gnd  BIT({00001111} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=1400 $y=600 $w=400 $h=600
VV2 Input_2 Gnd  BIT({00110011} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=1700 $y=-500 $w=400 $h=600
VV4 Input_3 Gnd  BIT({01010101} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=1700 $y=-1900 $w=400 $h=600
.PRINT   V(Input_1) $ $x=1550 $y=1650 $w=300 $h=1500 $r=270
.PRINT   V(Input_2) $ $x=50 $y=-50 $w=1500 $h=300 $r=180
.PRINT   V(Input_3) $ $x=50 $y=-1450 $w=1500 $h=300 $r=180
.PRINT   V(Output) $ $x=6950 $y=350 $w=1500 $h=300

********* Simulation Settings - Analysis Section *********
.tran 0.1n 80n
.temp -20 0 20 40 60 80 100


********* Simulation Settings - Additional SPICE Commands *********

.end
```

2.5 ANALYSIS OF FULL ADDER GATE USING CVDTL

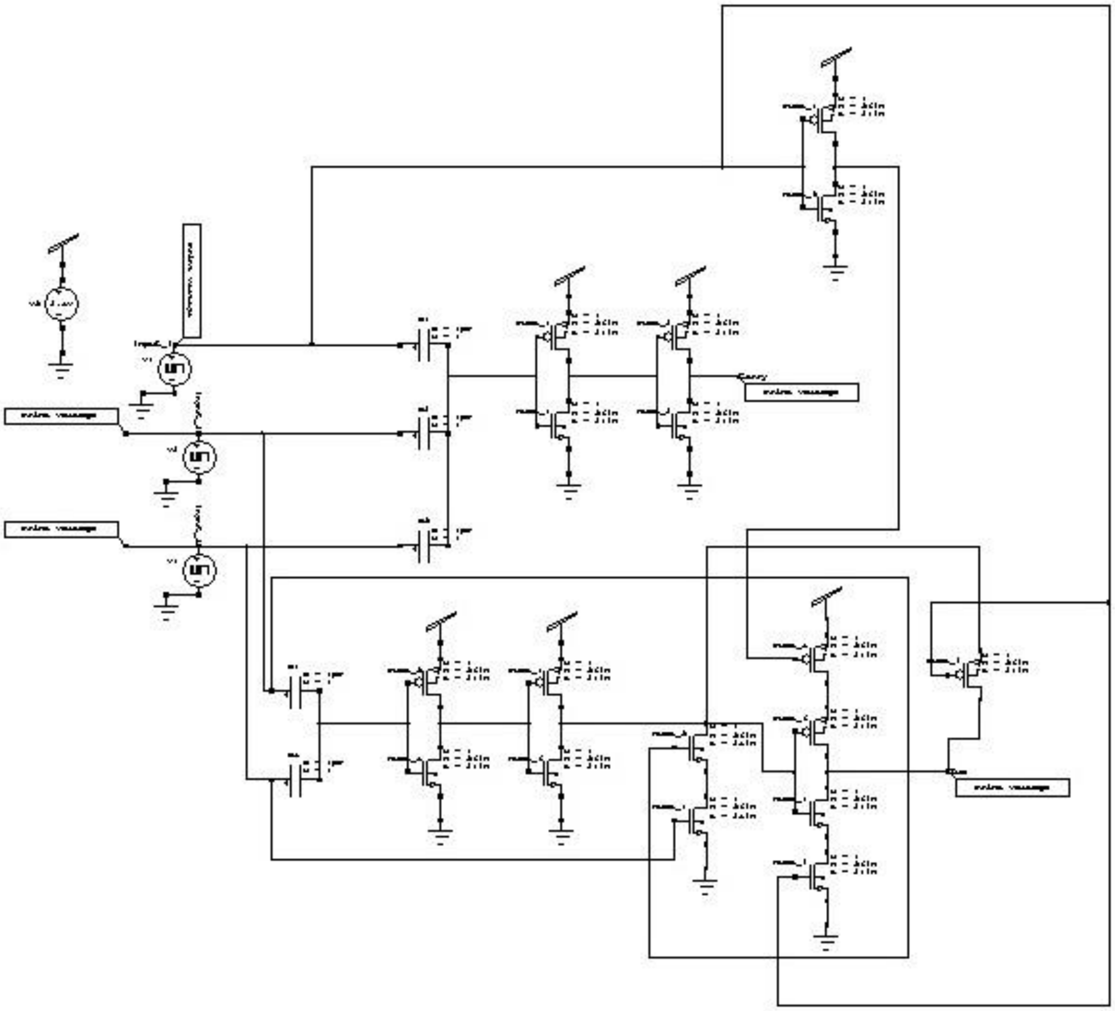

Fig. 2.45**:** Full Adder circuit diagram using CVDTL

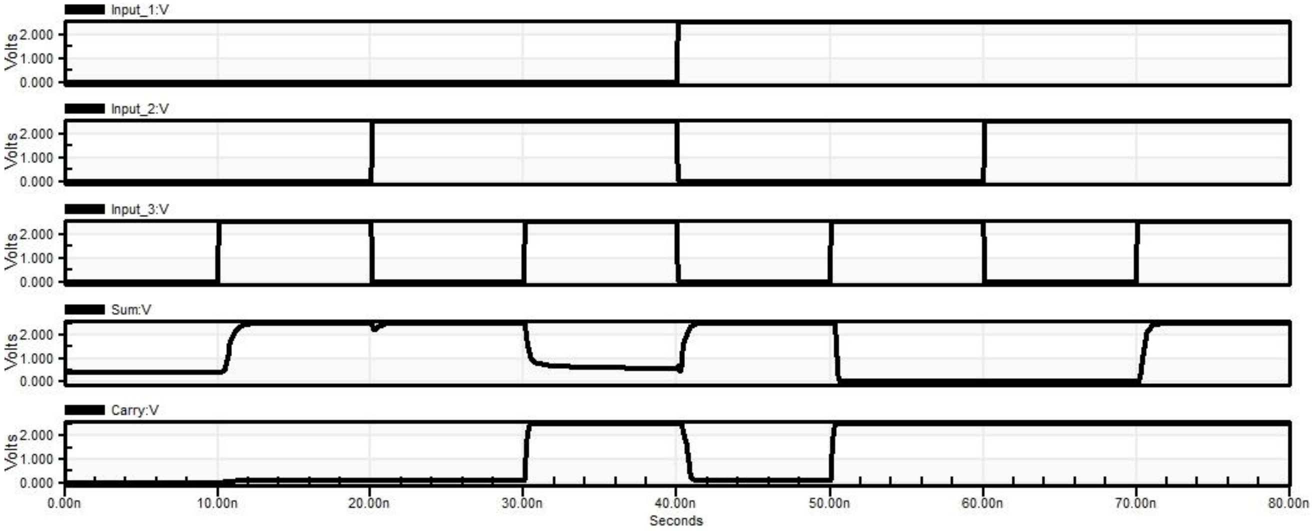

Fig. 2.46**:** Full Adder output waveform using CVDTL

Fig. 2.47**:** Average power consumed by Full Adder using CVDTL in different NM

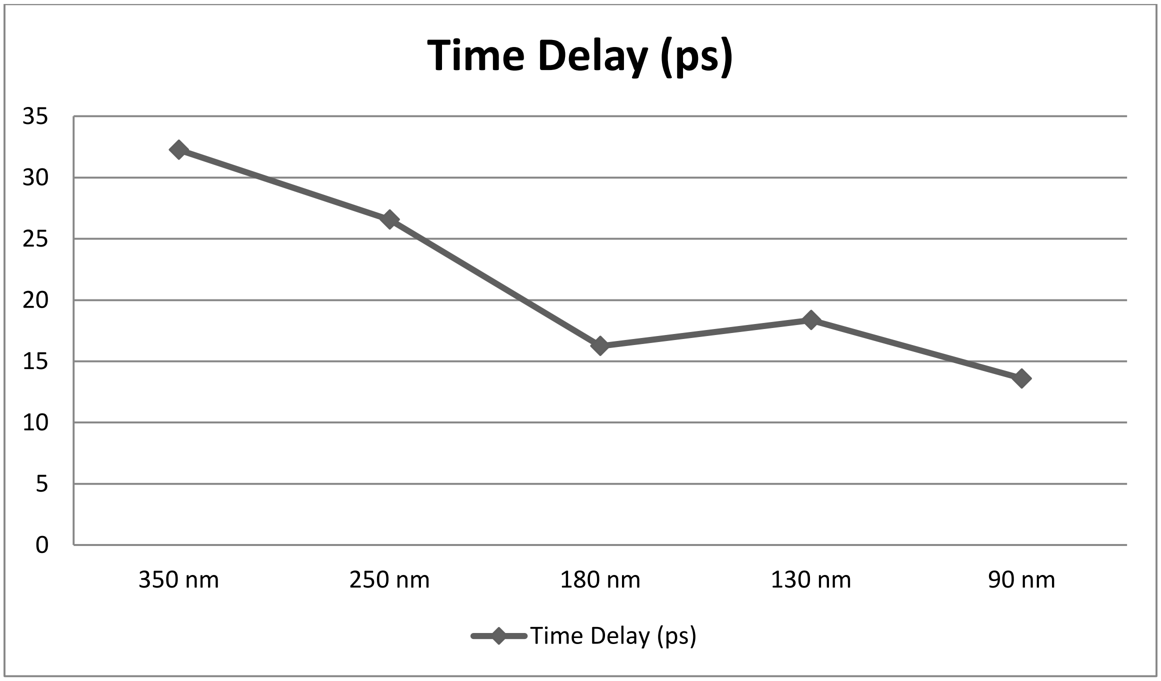

Fig. 2.48**:** Time delay of Full Adder using CVDTL in different NM

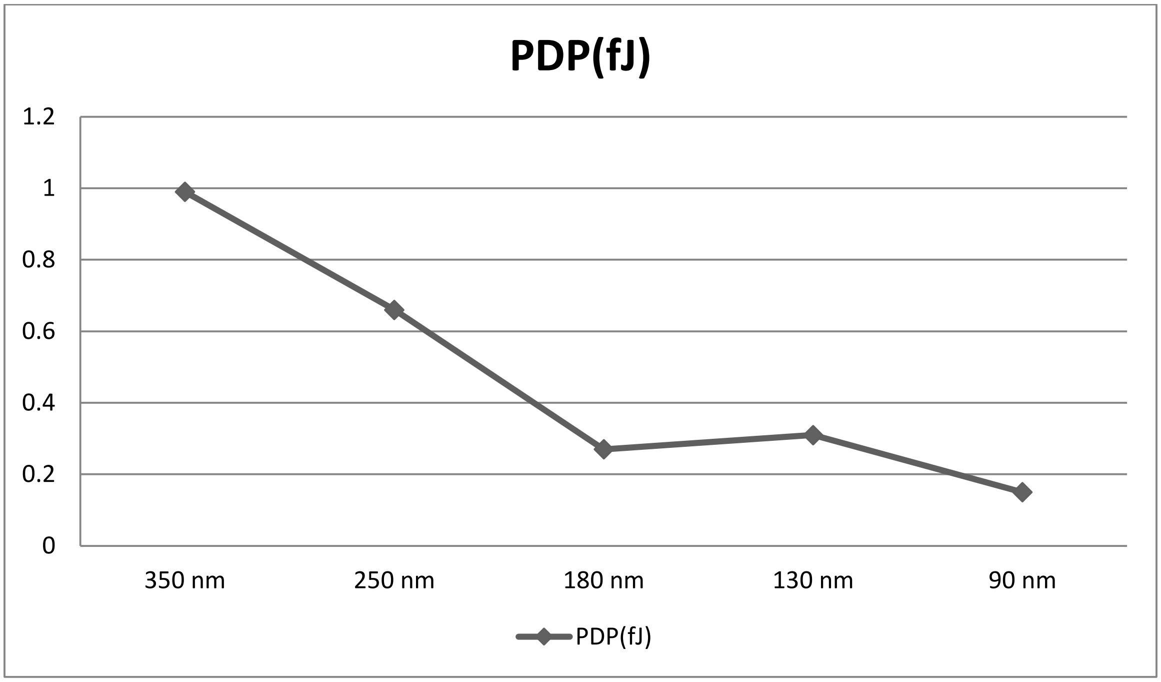

Fig. 2.49**:** Power Delay Product of Full Adder using CVDTL in different NM

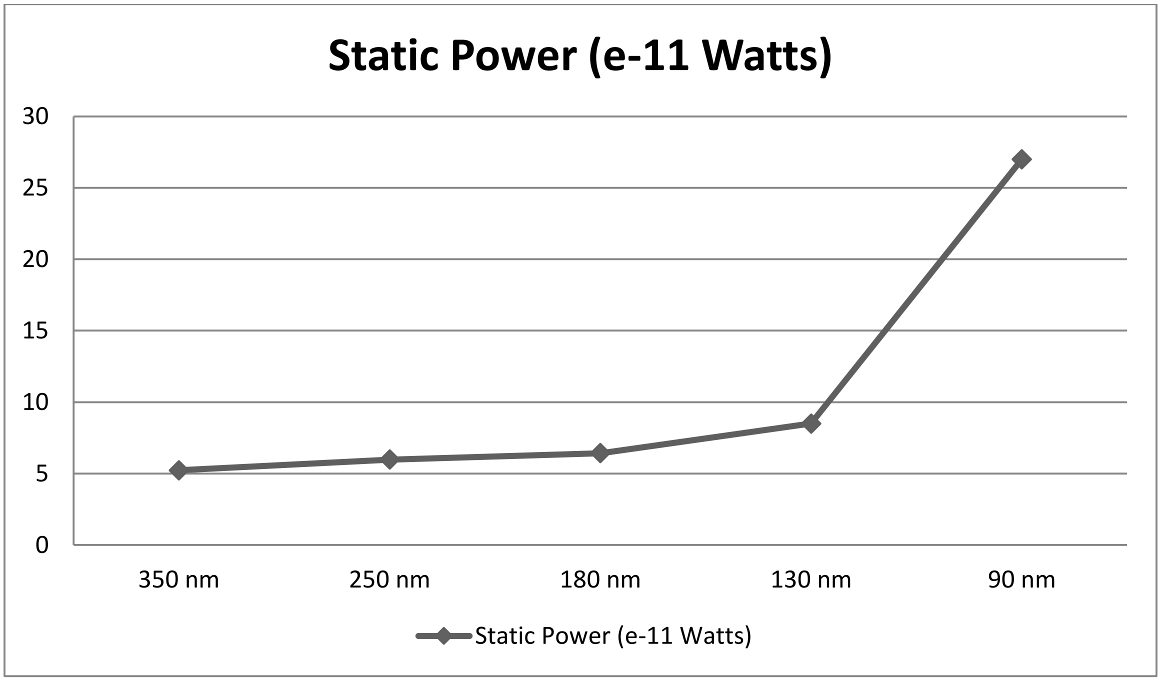

Fig. 2.50**:** Static Power of Full Adder using CVDTL in different NM

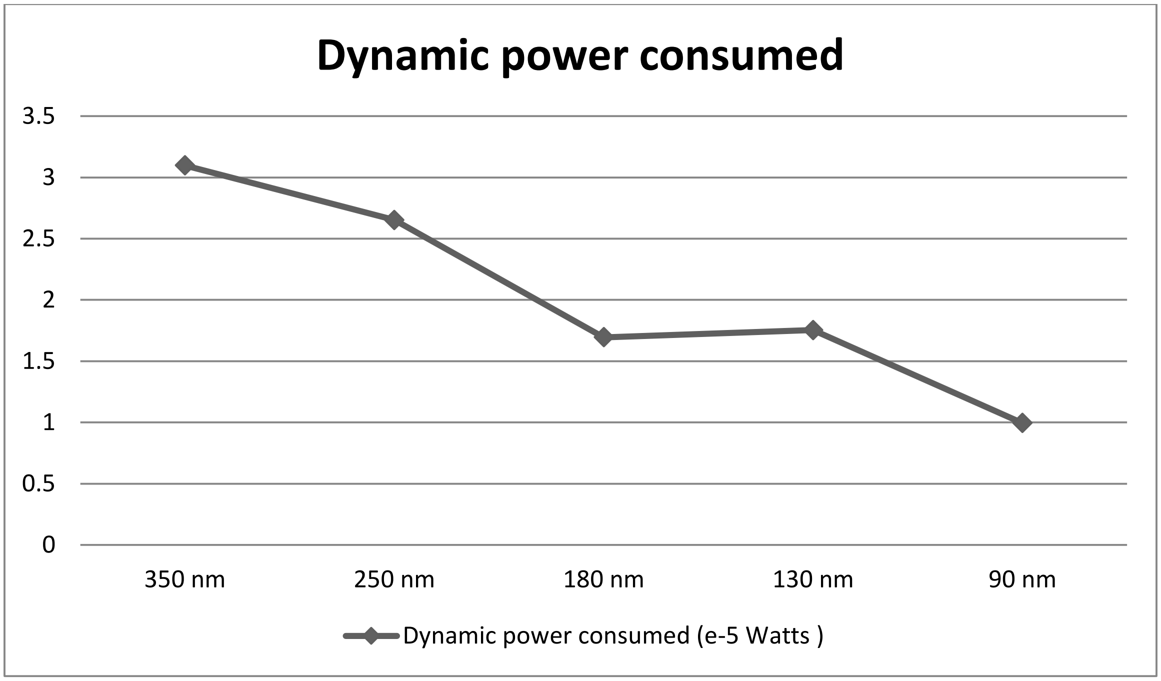

Fig. 2.51: Dynamic Power of Full Adder using CVDTL in different NM

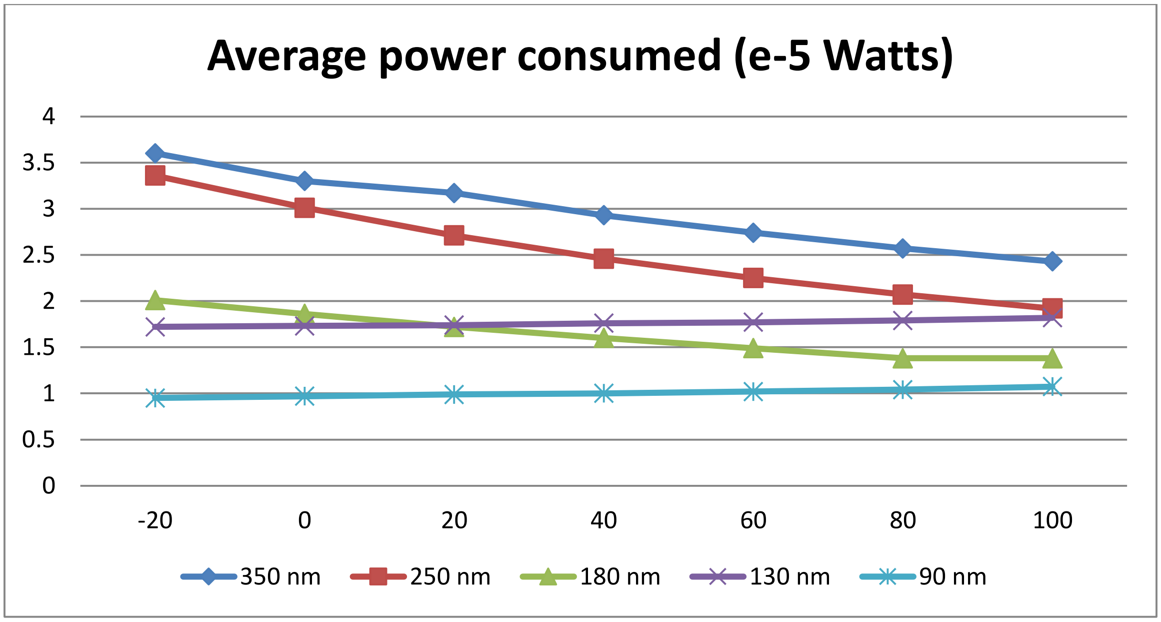

Fig. 2.52: Average power consumed by of Full Adder using CVDTL in different NM and in different Temperature

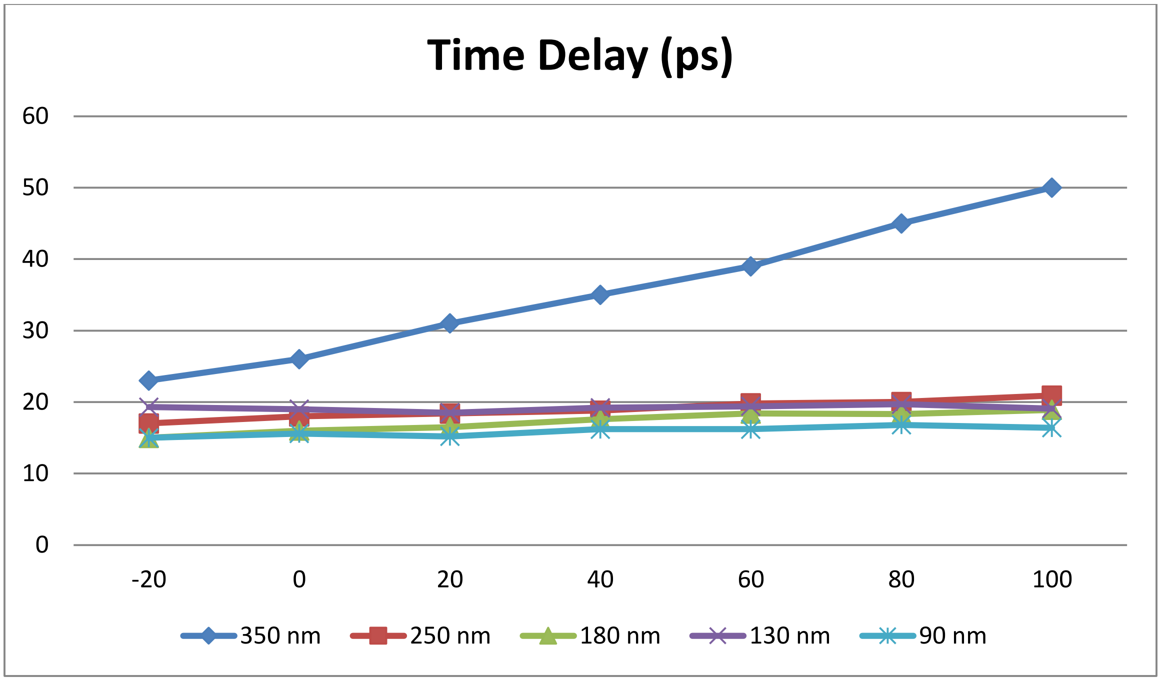

Fig. 2.53**:** Time Delay of Full Adder using CVDTL in different NM and in different Temperature

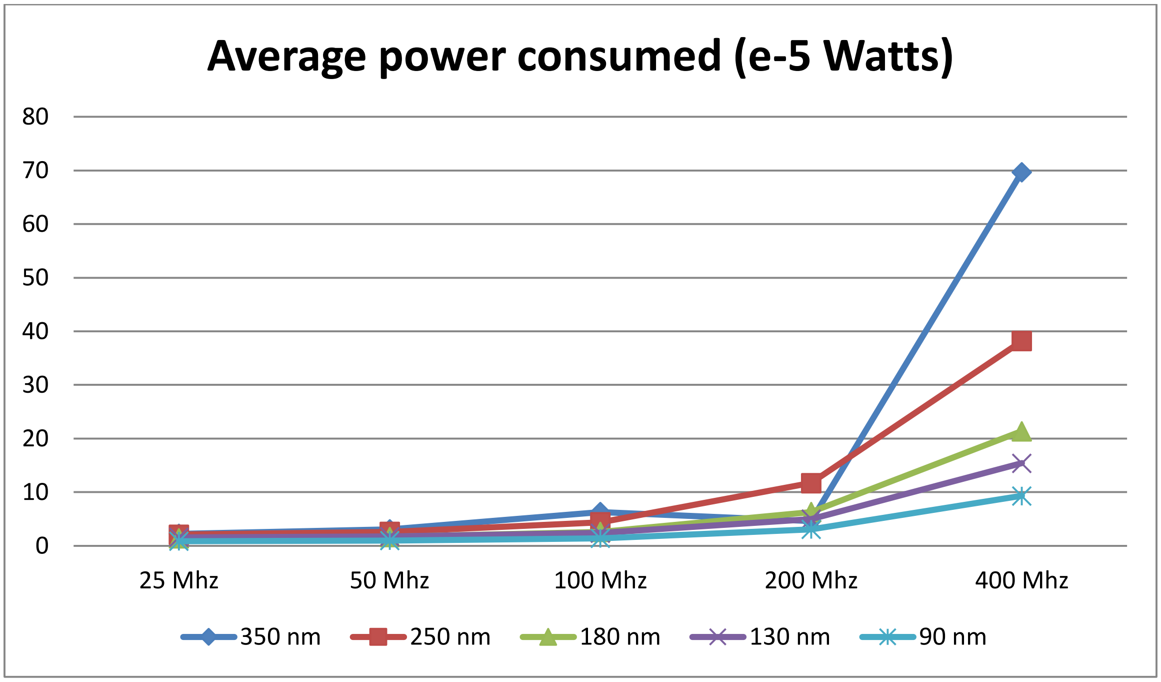

Fig. 2.54**:** Average power consumed by of Full Adder using CVDTL in different NM and in different Frequency

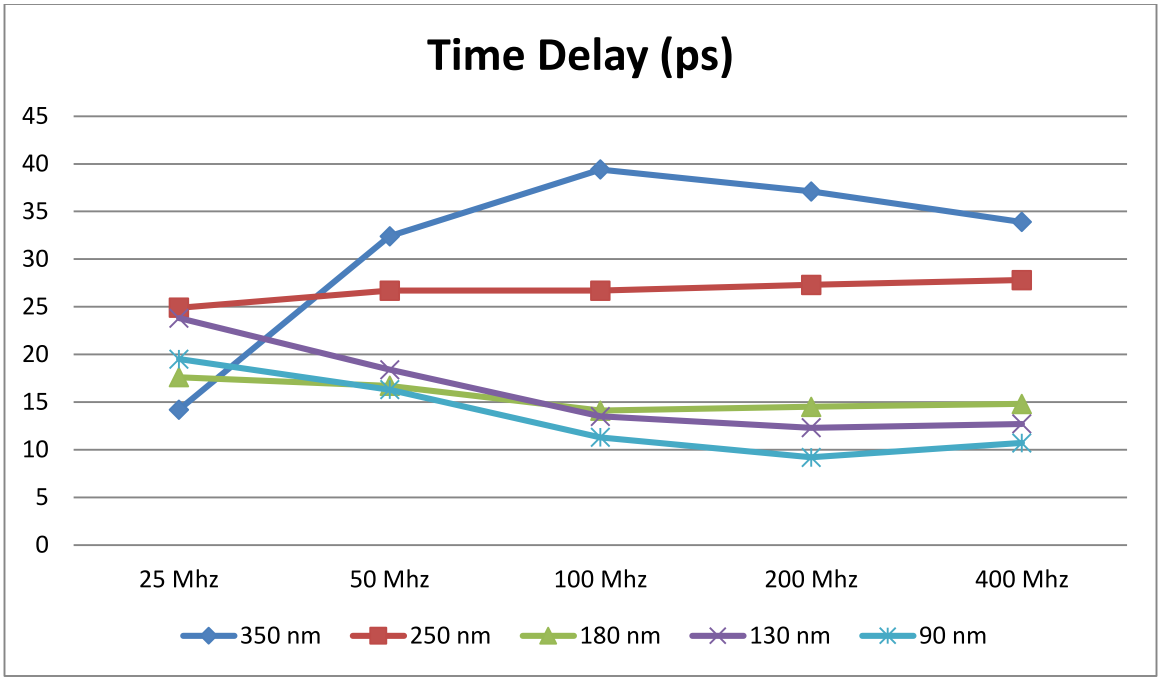

Fig. 2.55**:** Time Delay of Full Adder using CVDTL in different NM and in different Frequency

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

***** Top Level *****
CC1 Input_1 N_3  1p $ $x=-700 $y=2300 $w=600 $h=400
CC2 Input_2 N_3  1p $ $x=-700 $y=1200 $w=600 $h=400
CC3 Input_3 N_3  1p $ $x=-700 $y=-200 $w=600 $h=400
CC4 Input_2 N_5  1p $ $x=-1400 $y=-1900 $w=600 $h=400
CC5 Input_3 N_5  1p $ $x=-1400 $y=-3000 $w=600 $h=400
MNMOS_1 N_4 N_3 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=900 $y=1300
$w=400 $h=600
MNMOS_2 Carry N_4 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=2400
$y=1300 $w=400 $h=600
MNMOS_3 N_9 Input_2 N_1 0 NMOS W=360n L=250n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3500 $y=-
2600 $w=400 $h=600
MNMOS_4 N_1 Input_3 Gnd 0 NMOS W=360n L=250n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3500 $y=-
3500 $w=400 $h=600
MNMOS_5 N_6 N_5 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=200 $y=-2900
$w=400 $h=600
MNMOS_6 N_9 N_6 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1700 $y=-
2900 $w=400 $h=600
```

```
MNMOS_7 Sum N_9 N_2 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5000 $y=-
2900 $w=400 $h=600
MNMOS_8 N_2 Input_1 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5000 $y=-
3800 $w=400 $h=600
MNMOS_9 N_8 Input_1 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3100
$y=4200 $w=400 $h=600
MPMOS_1 N_4 N_3 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=900
$y=2400 $w=400 $h=600
MPMOS_2 Carry N_4 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=2400
$y=2400 $w=400 $h=600
MPMOS_3 N_6 N_5 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=200 $y=-
1800 $w=400 $h=600
MPMOS_4 N_9 N_6 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1700 $y=-
1800 $w=400 $h=600
MPMOS_5 Sum N_9 N_7 Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5000 $y=-
1800 $w=400 $h=600
MPMOS_6 N_7 N_8 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5000 $y=-
200 $w=400 $h=600
MPMOS_7 N_8 Input_1 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3100
$y=5300 $w=400 $h=600
MPMOS_8 Sum Input_1 N_9 Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=6600
$y=-2000 $w=400 $h=600
VV3 Vdd Gnd  DC 2.5 $ $x=-4700 $y=2800 $w=400 $h=600
VV1 Input_1 Gnd  BIT({00001111} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=-3300 $y=2000
$w=400 $h=600
VV2 Input_2 Gnd  BIT({00110011} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=-3000 $y=900
$w=400 $h=600
VV4 Input_3 Gnd  BIT({01010101} PW=10n ON=2.5 RT=100p FT=100p LT=10n HT=10n) $ $x=-3000 $y=-500
$w=400 $h=600
.PRINT   V(Carry) $ $x=3950 $y=1750 $w=1500 $h=300
.PRINT   V(Input_1) $ $x=-3150 $y=3050 $w=300 $h=1500 $r=270
.PRINT   V(Input_2) $ $x=-4650 $y=1350 $w=1500 $h=300 $r=180
.PRINT   V(Input_3) $ $x=-4650 $y=-50 $w=1500 $h=300 $r=180
.PRINT   V(Sum) $ $x=10050 $y=-2150 $w=1500 $h=300 $r=180 $m

********* Simulation Settings - Analysis Section *********
.tran 0.1n 80n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

CHAPTER 3

IMPLEMENTATION OF CMOS Output Wired Logic (COWL)

3.1. ANALYSIS OF OR GATE USING COWL:

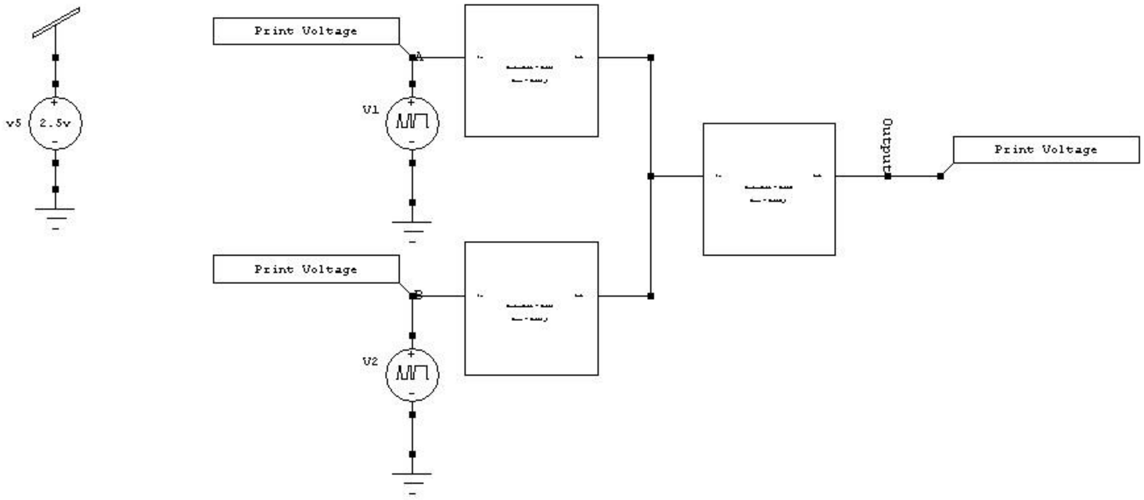

Fig. 3.1**:** OR gate circuit diagram using COWL

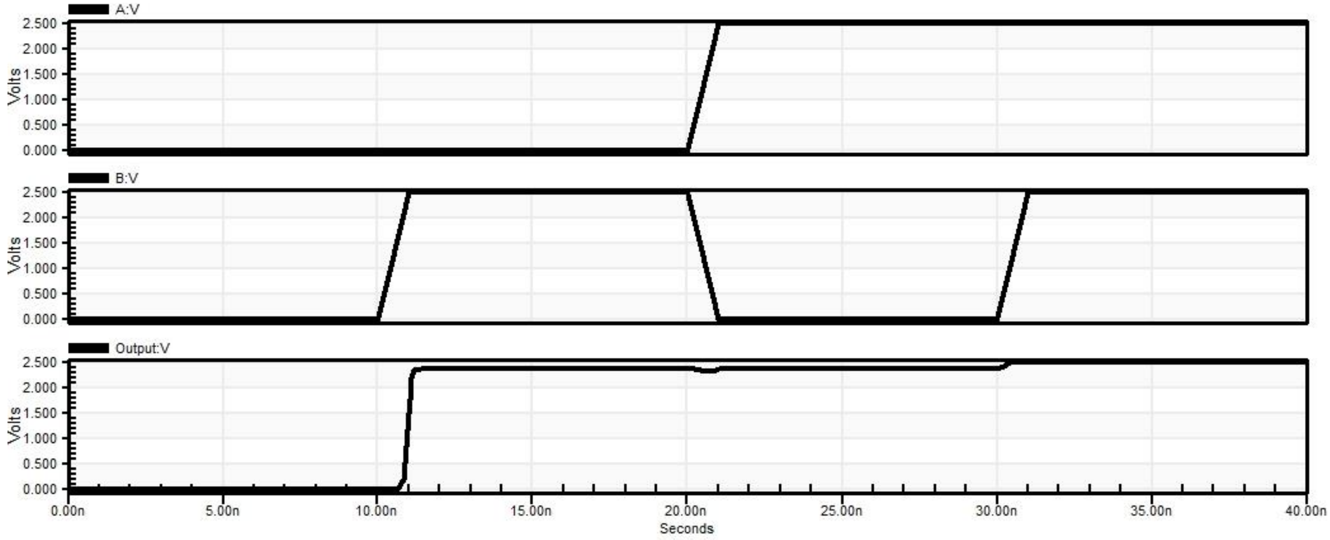

Fig. 3.2**:** OR gate output waveform using COWL

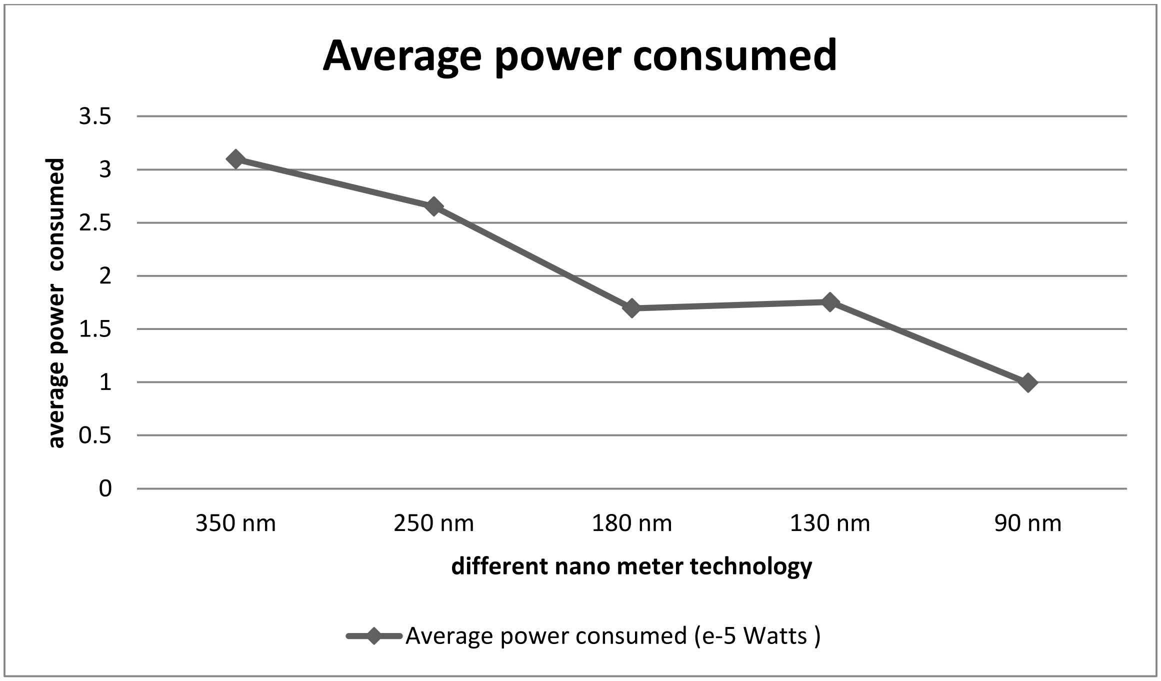

Fig. 3.3**:** Average power consumed by OR Gate using COWL in different NM

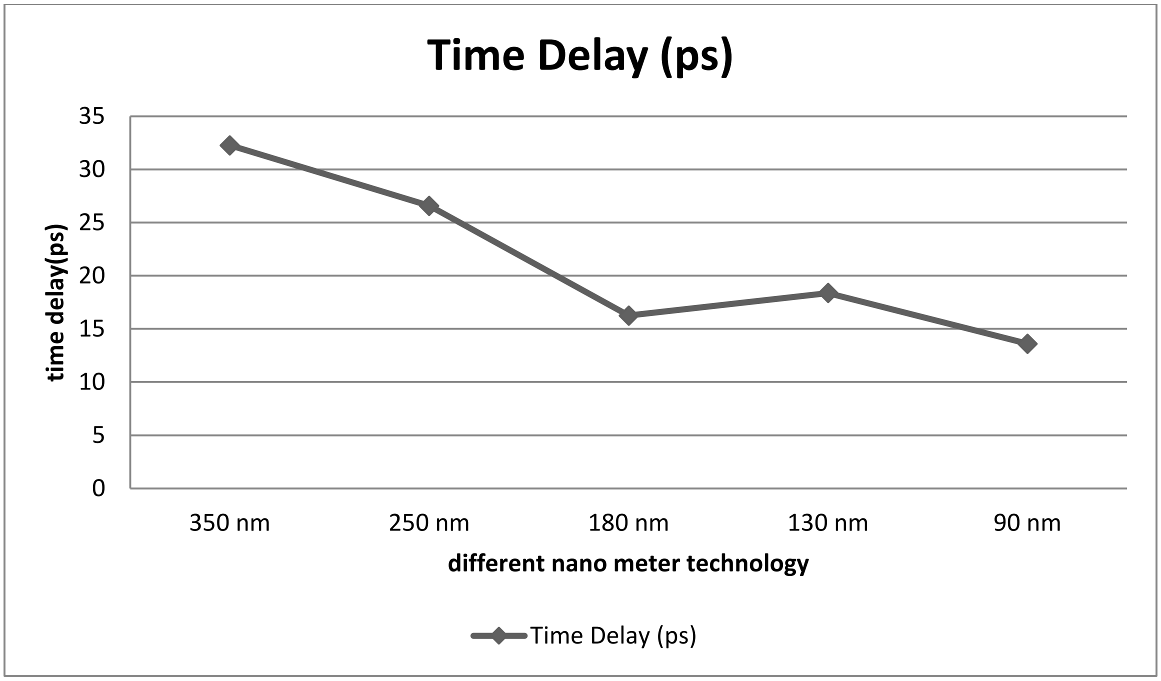

Fig. 3.4**:** Time delay of OR Gate using COWL in different NM

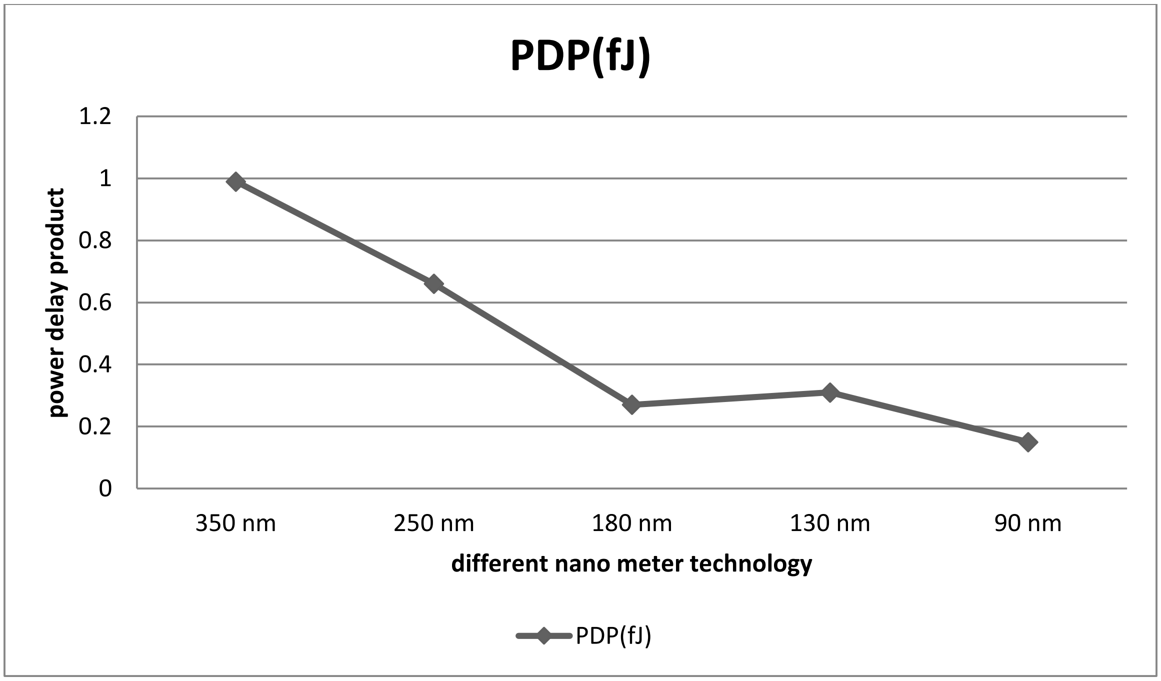

Fig. 3.5**:** Power Delay Product of OR Gate using COWL in different NM

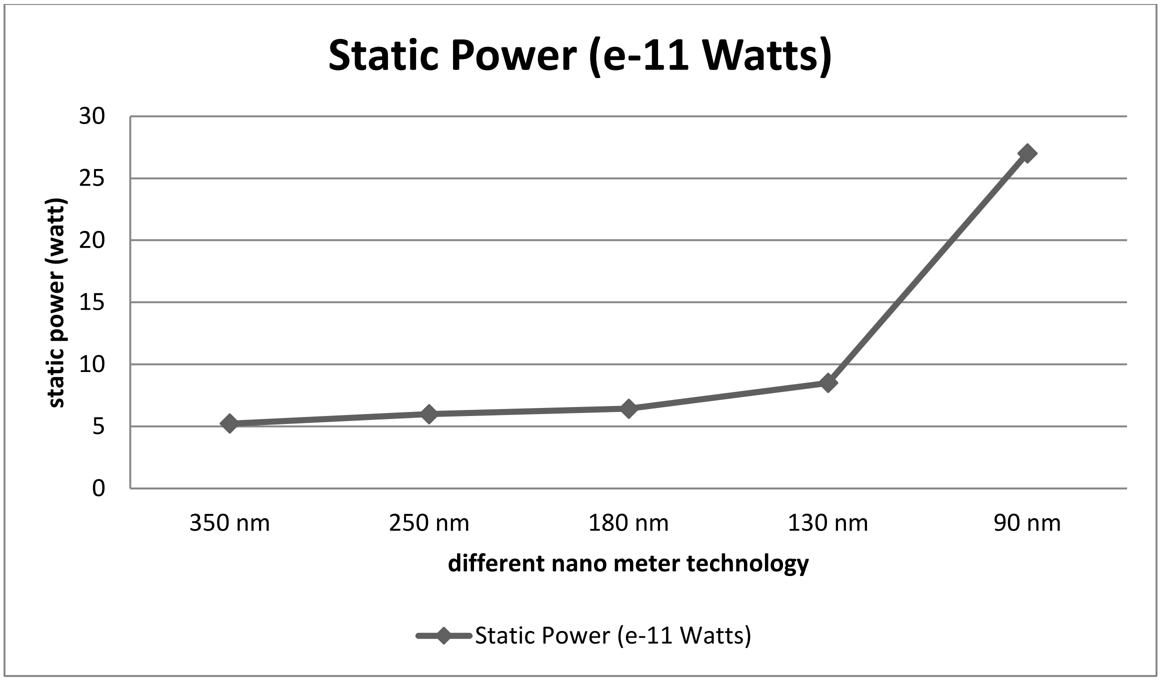

Fig. 3.6**:** Static Power of OR Gate using COWL in different NM

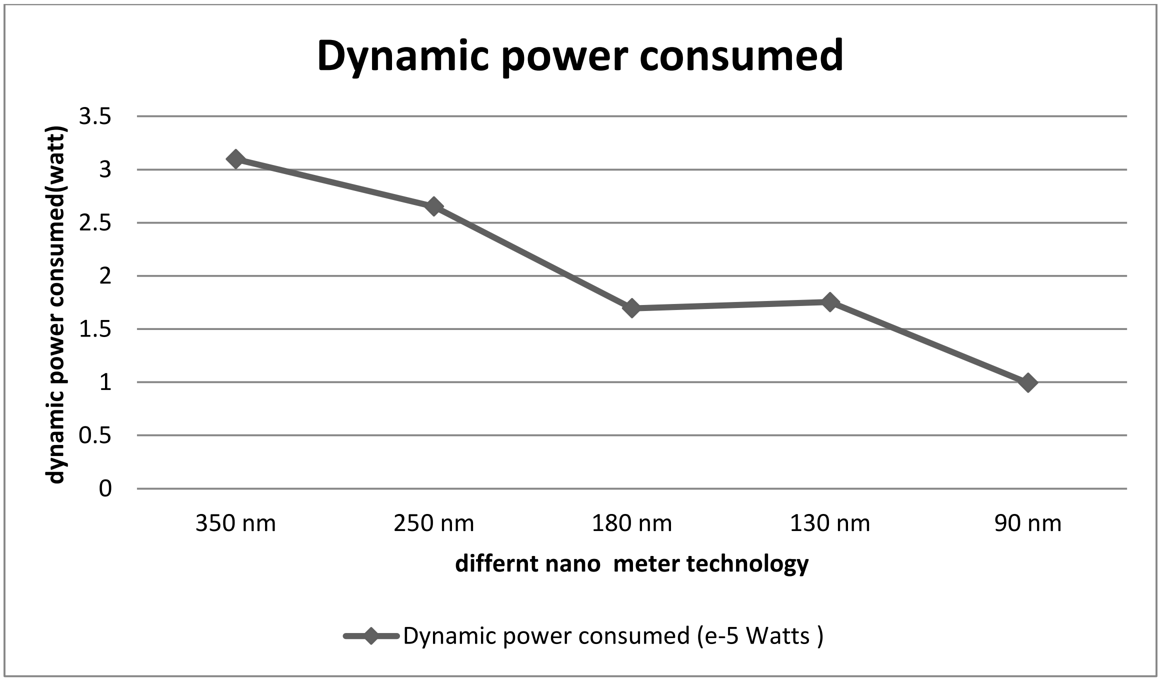

Fig. 3.7**:** Dynamic Power of OR Gate using COWL in different NM

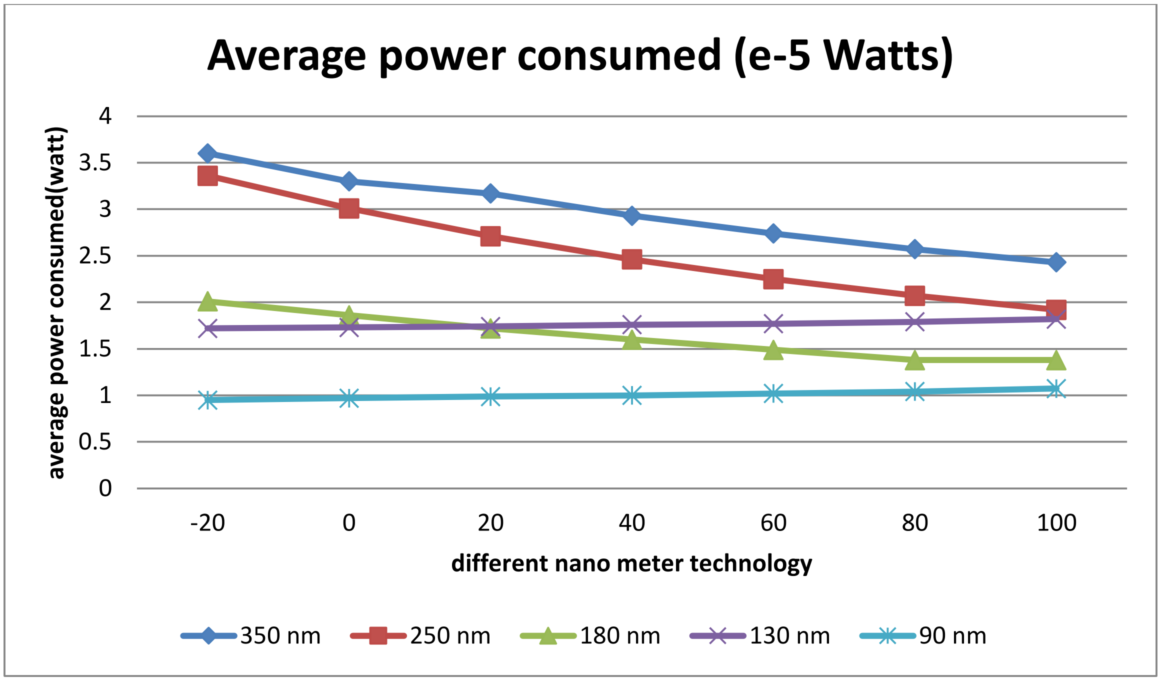

Fig. 3.8**:** Average power consumed by of OR Gate using COWL in different NM and in different Temperature

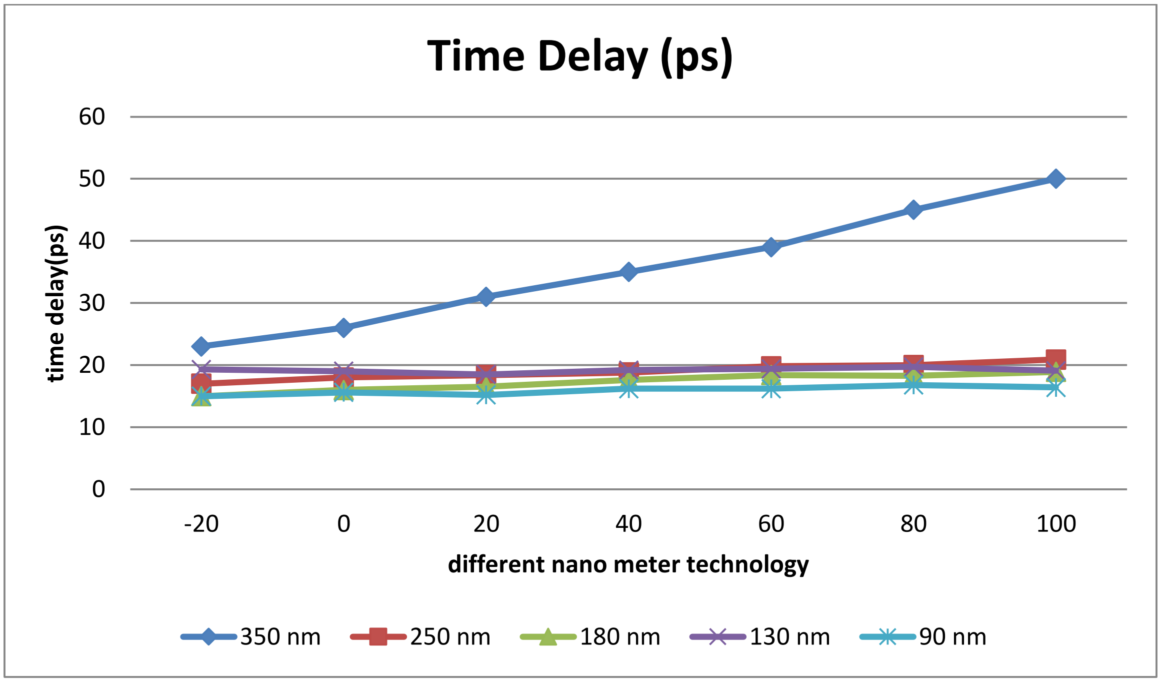

Fig. 3.9**:** Time Delay of OR Gate using COWL in different NM and in different Temperature

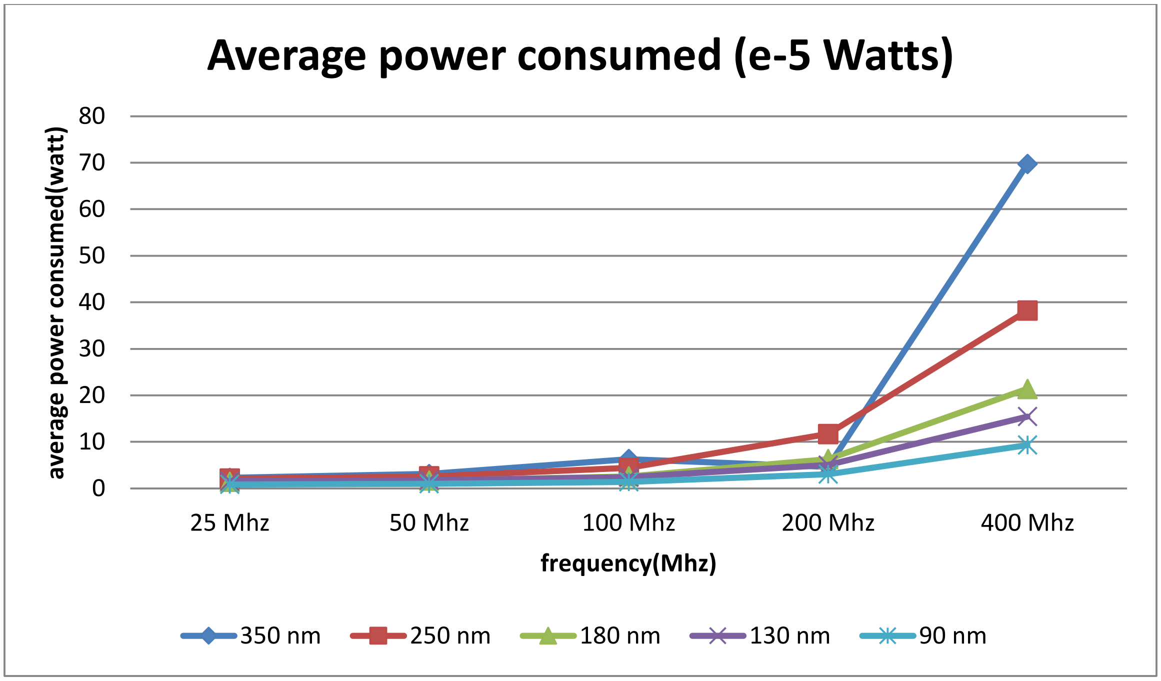

Fig. 3.10**:** Average power consumed by of OR Gate using COWL in different NM and in different Frequency

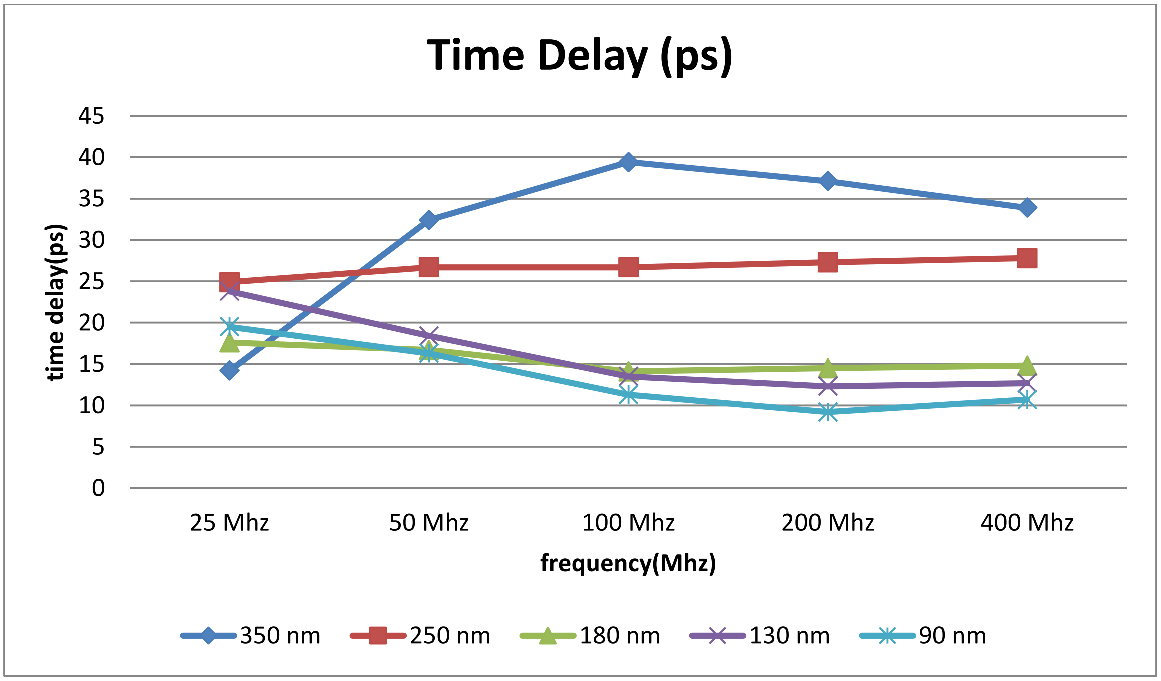

Fig. 3.11**:** Time Delay of OR Gate using COWL in different NM and in different Frequency

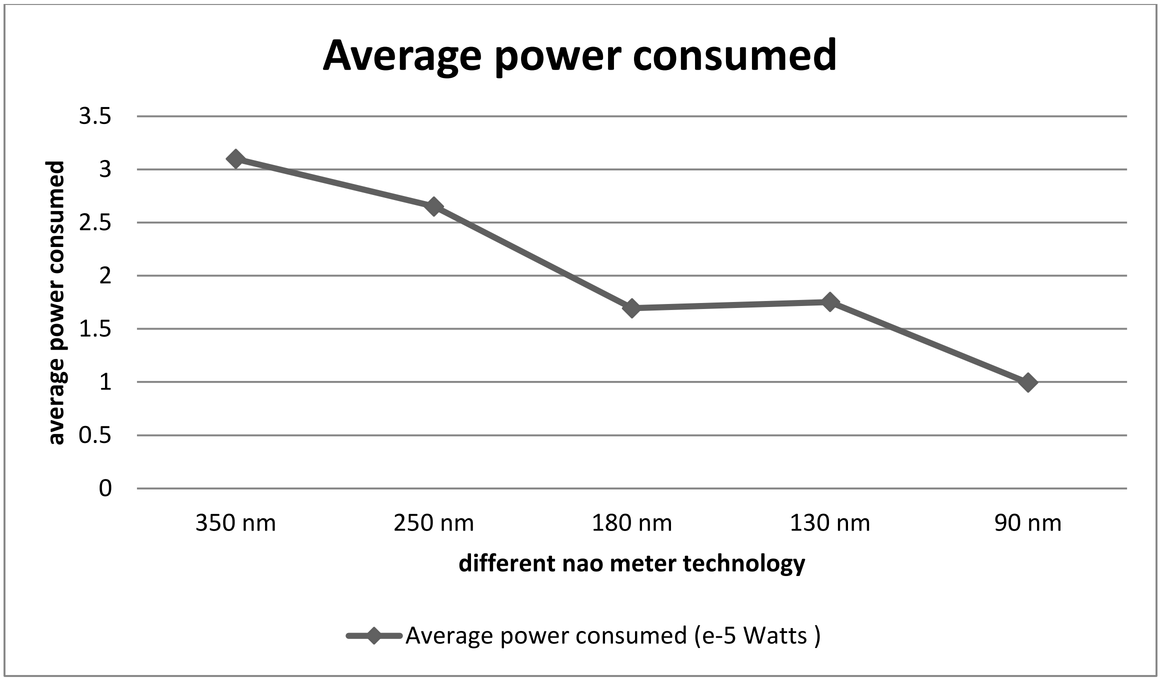

Fig. 3.12**:** Average power consumed by OR Gate using COWL in different NM (Post Layout)

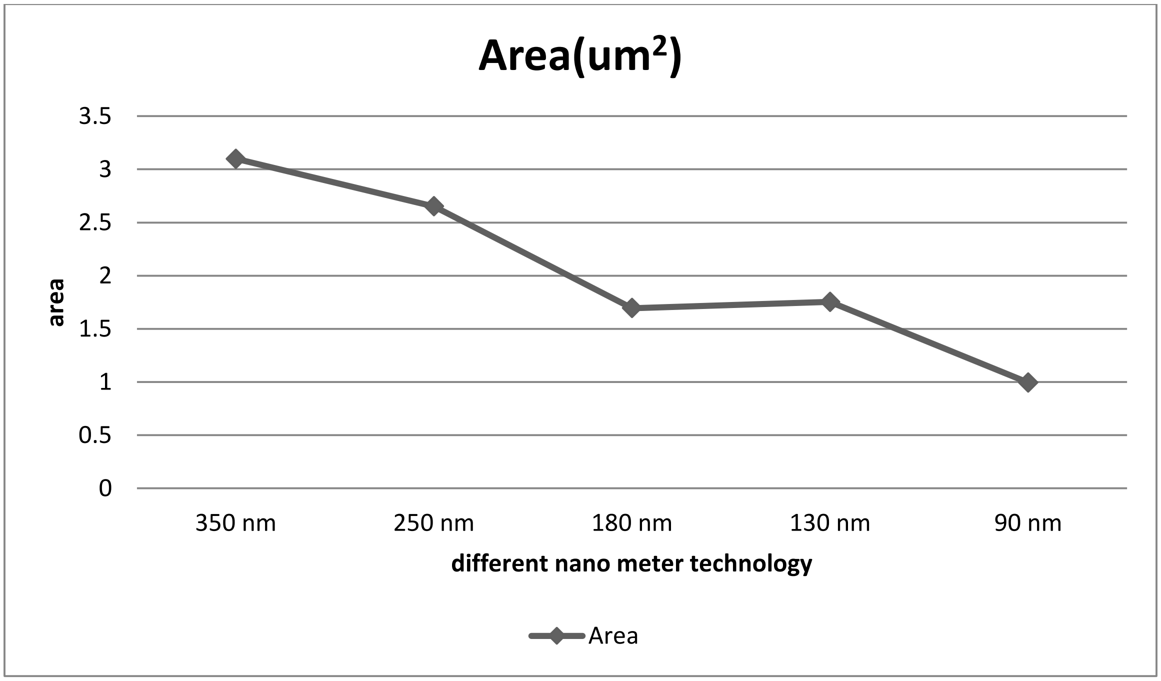

Fig. 3.13**:** Average power consumed by OR Gate using COWL in different NM

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

*************** Subcircuits *****************
.subckt Cell0 In Out Gnd Vdd
MNMOS_1 Out In Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=-100 $y=-600
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=900n L=240n AS=810f PS=3.6u AD=810f PD=3.6u $ $x=-100 $y=600
$w=400 $h=600
.ends

.subckt Cell3 In Out Gnd Vdd
MNMOS_1 Out In Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3600 $y=-300
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=1.2u L=240n AS=1.08p PS=4.2u AD=1.08p PD=4.2u $ $x=3600 $y=900
$w=400 $h=600
.ends


***** Top Level *****
XCell0_1 A N_1 Gnd Vdd Cell0 $ $x=-7200 $y=500 $w=1800 $h=1000
XCell0_2 B N_1 Gnd Vdd Cell0 $ $x=-7200 $y=-1300 $w=1800 $h=1000
XCell3_1 N_1 Output Gnd Vdd Cell3 $ $x=-5400 $y=-400 $w=1800 $h=1000
Vv5 Vdd Gnd  DC 2.5 $ $x=-10800 $y=100 $w=400 $h=600
VV1 A Gnd  BIT({0011} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-8100 $y=100 $w=400 $h=600
VV2 B Gnd  BIT({0101} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-8100 $y=-1800 $w=400
$h=600
.PRINT   V(A) $ $x=-8850 $y=750 $w=1500 $h=300 $r=180
.PRINT   V(B) $ $x=-8850 $y=-1050 $w=1500 $h=300 $r=180
.PRINT   V(Output) $ $x=-3350 $y=-150 $w=1500 $h=300 $r=180 $m

********* Simulation Settings - Analysis Section *********
.tran 0.1n 40n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

3.2ANALYSISOF AND GATE USING COWL

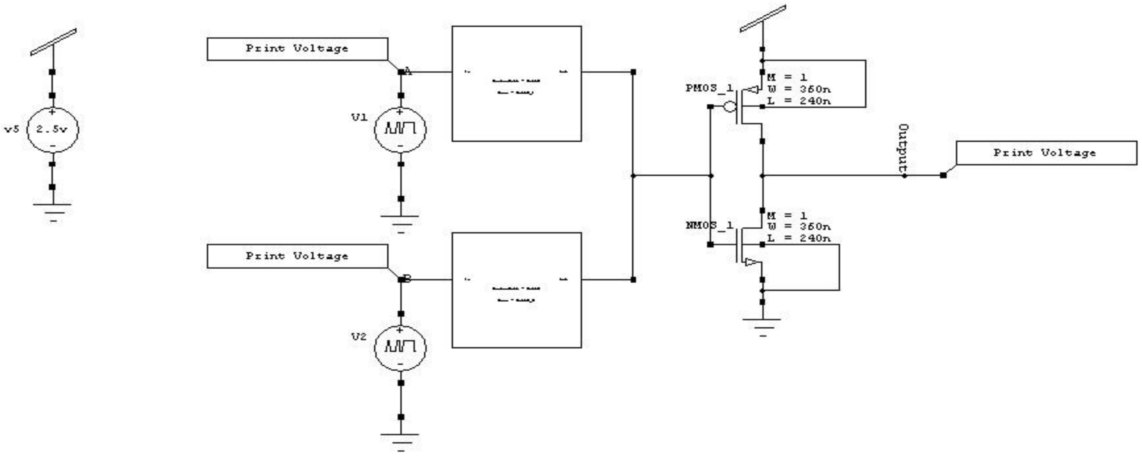

Fig. 3.14**:** AND gate circuit diagram using COWL

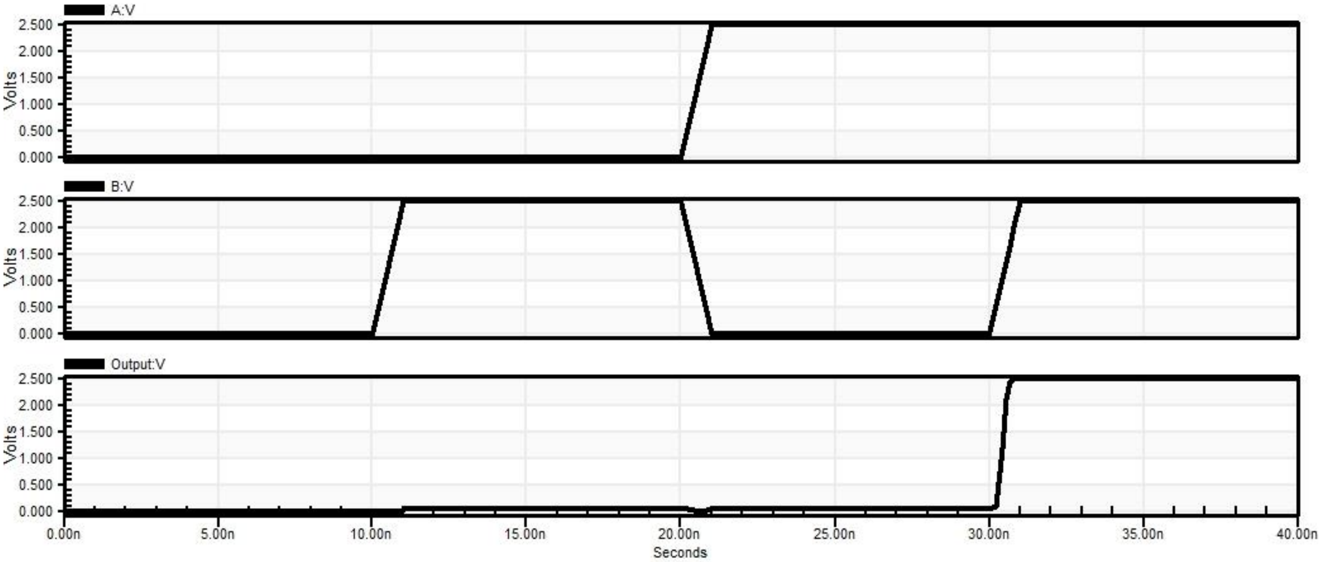

Fig. 3.15**:** AND gate output waveform using COWL

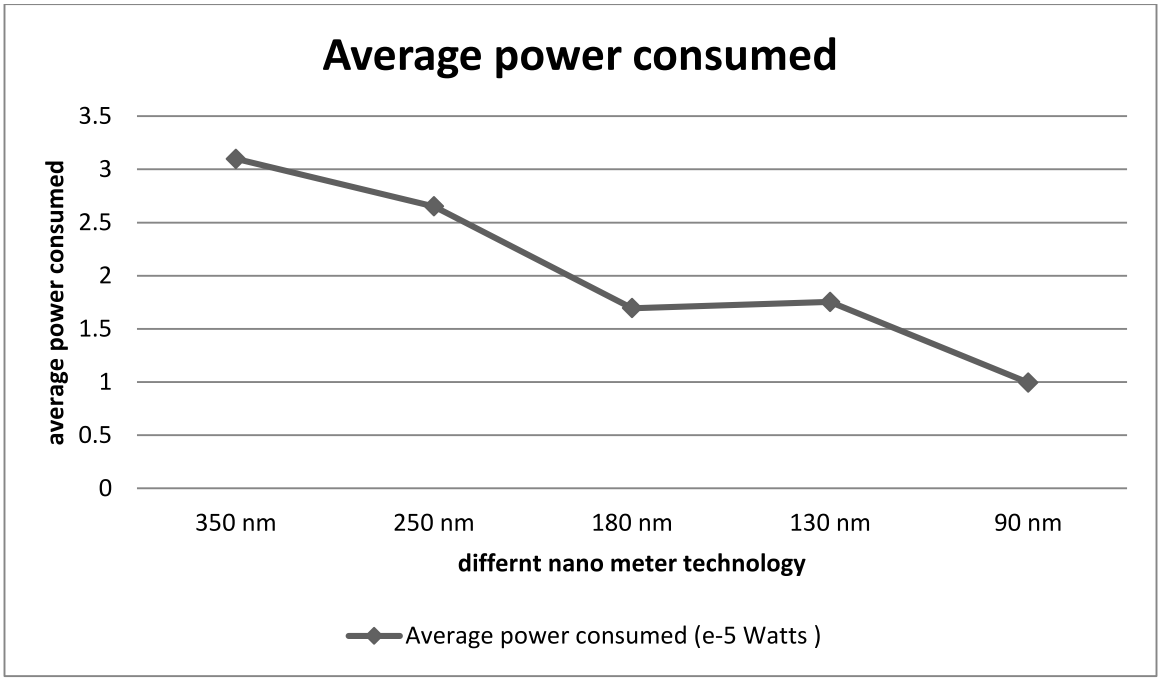

Fig. 3.16**:** Average power consumed by AND Gate using COWL in different NM

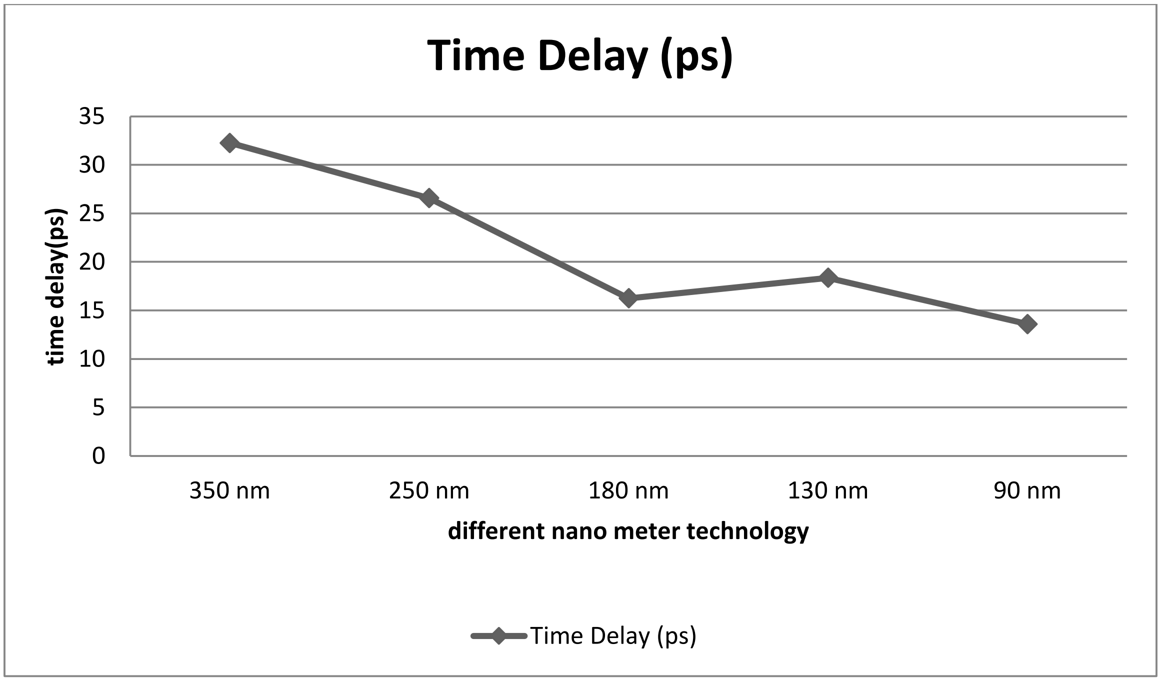

Fig. 3.17**:** Time delay of AND Gate using COWL in different NM

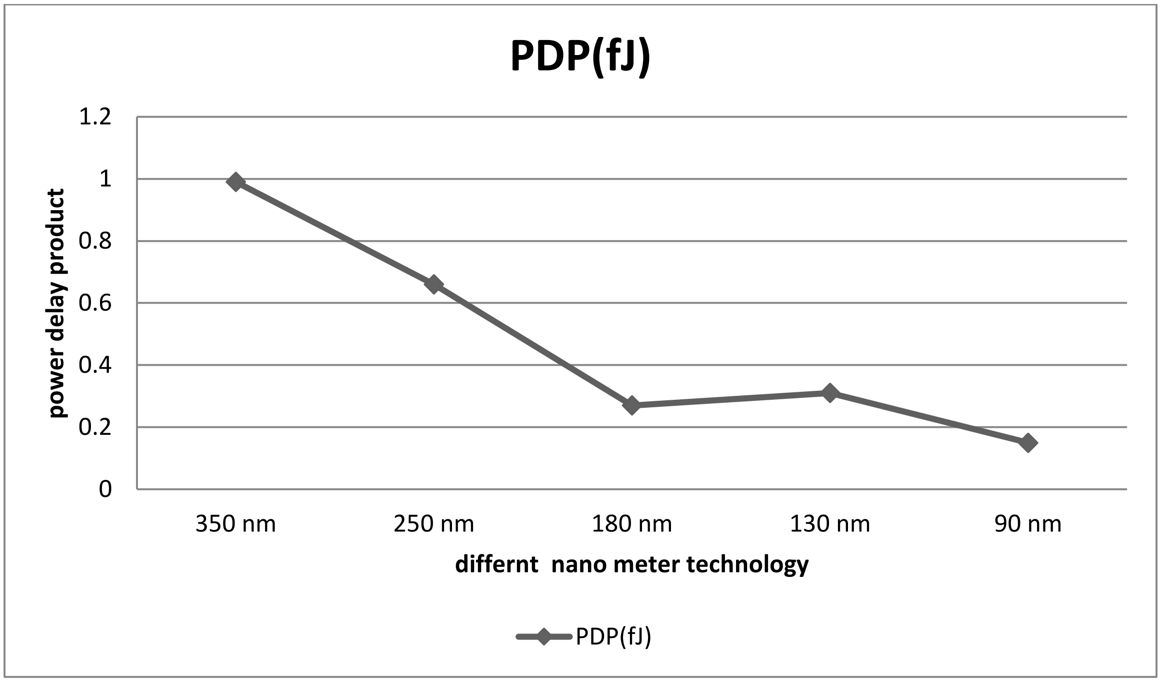

Fig. 3.18**:** Power Delay Product of AND Gate using COWL in different NM

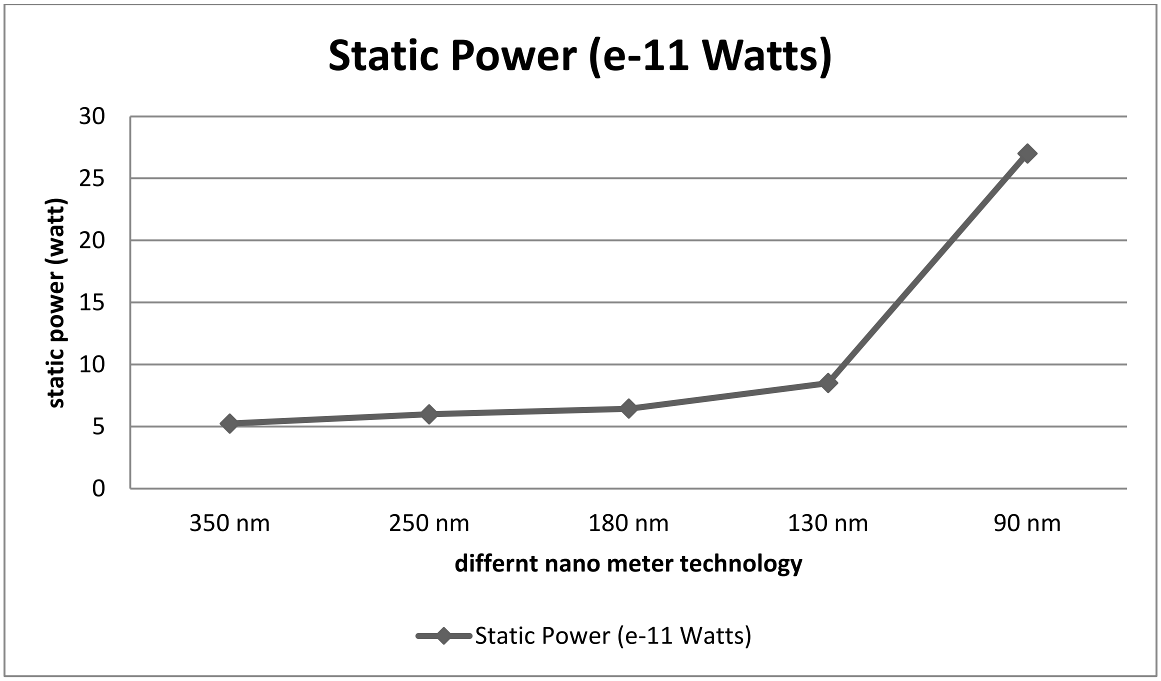

Fig. 3.19**:** Static Power of AND Gate using COWL in different NM

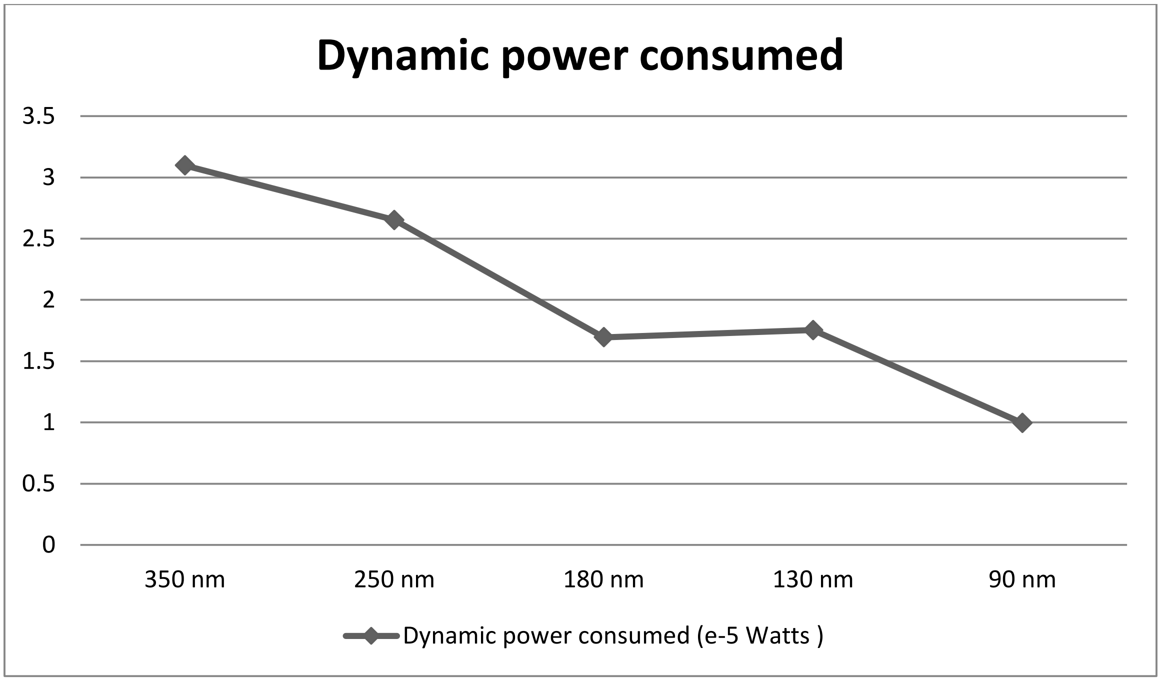

Fig. 3.20**:** Dynamic Power of AND Gate using COWL in different N

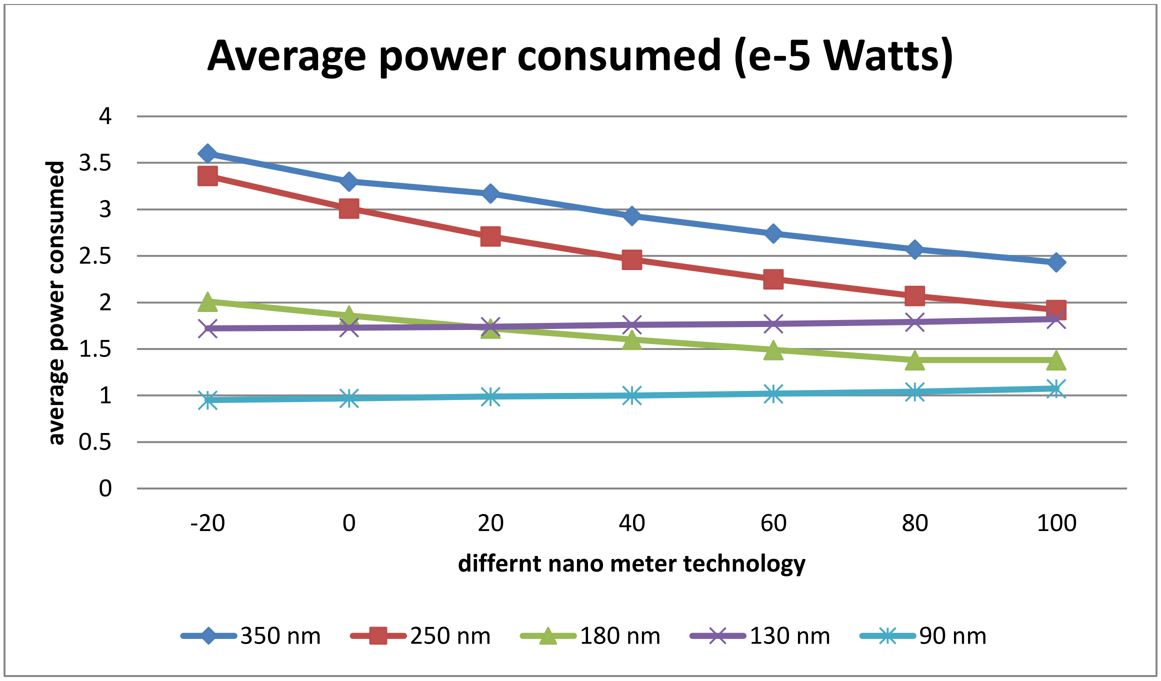

Fig. 3.21**:** Average power consumed by of AND Gate using COWL in different NM and in different Temperature

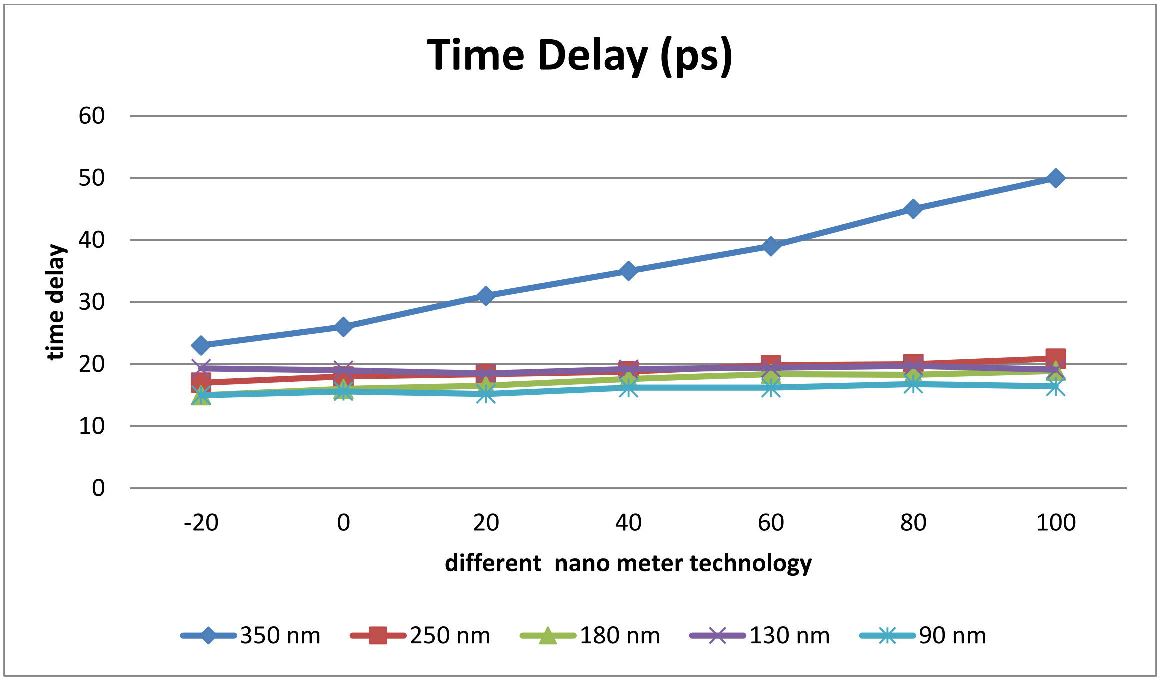

Fig. 3.22**:** Time Delay of AND Gate using COWL in different NM and in different Temperature

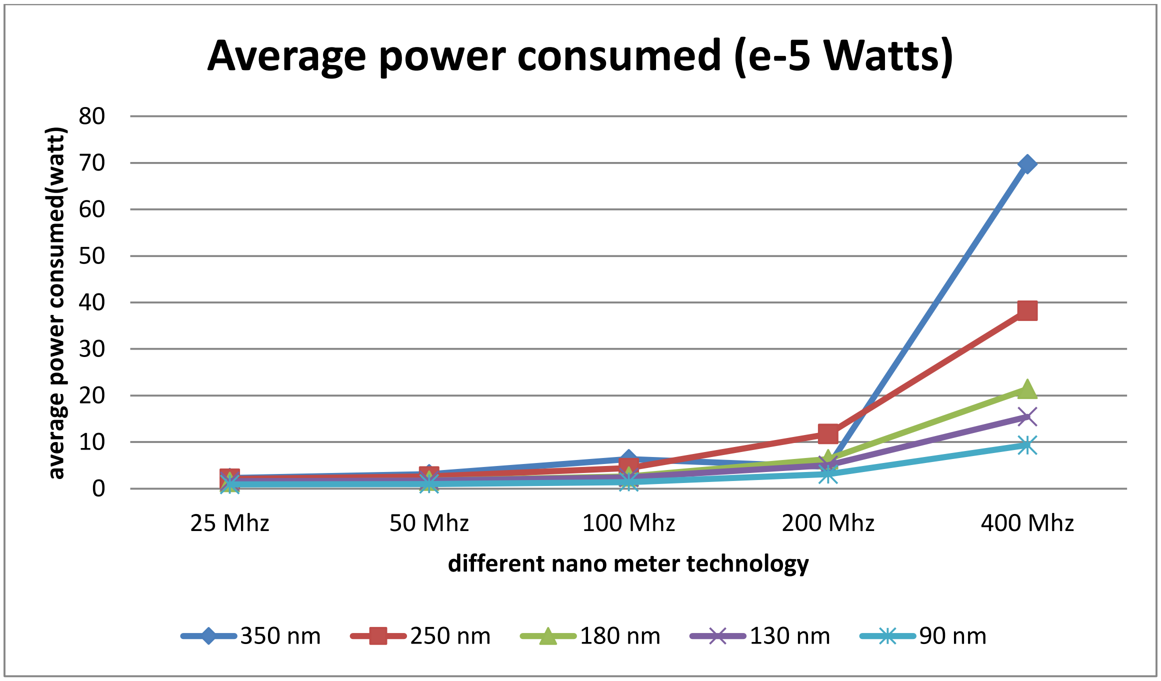

Fig.3**:23** Average power consumed by of AND Gate using COWL in different NM and in different Frequency

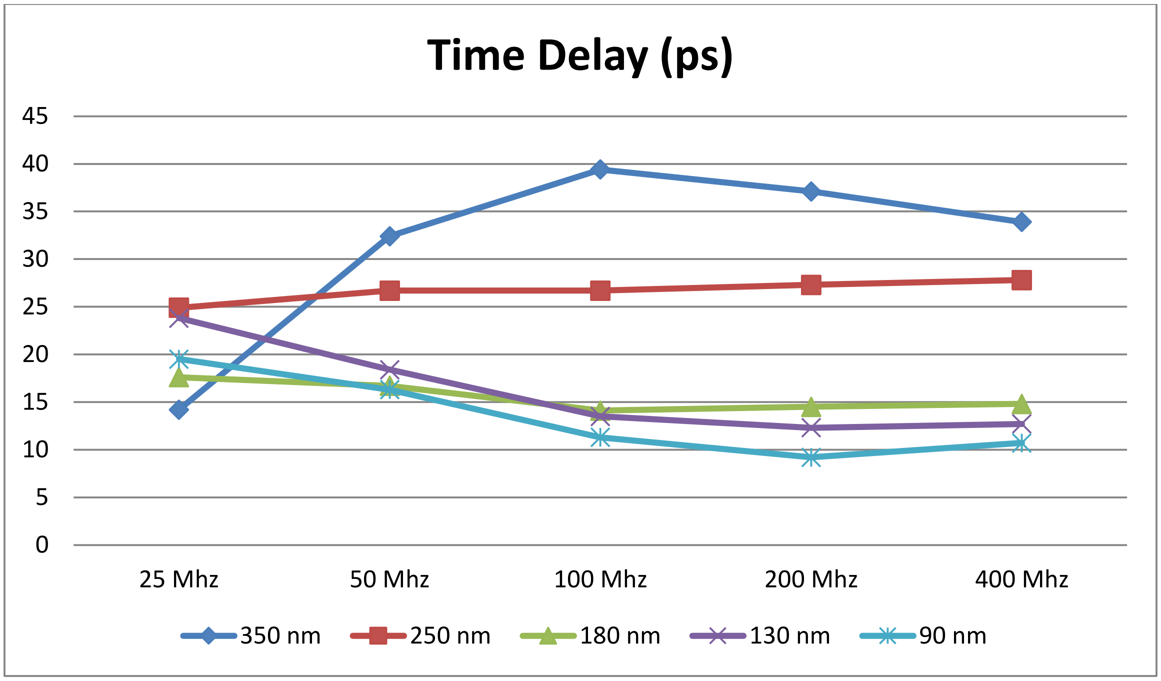

Fig. 3.24**:** Time Delay of AND Gate using COWL in different NM and in different Frequency

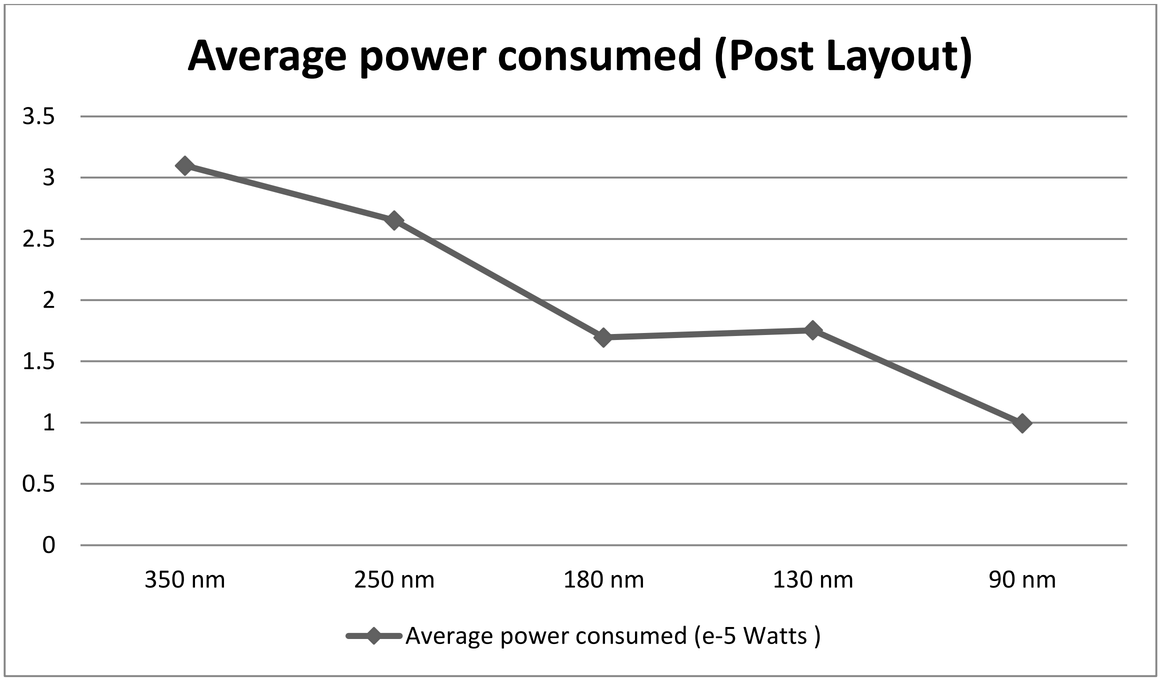

Fig. 3.25**:** Average power consumed by AND Gate using COWL in different NM (Post Layout)

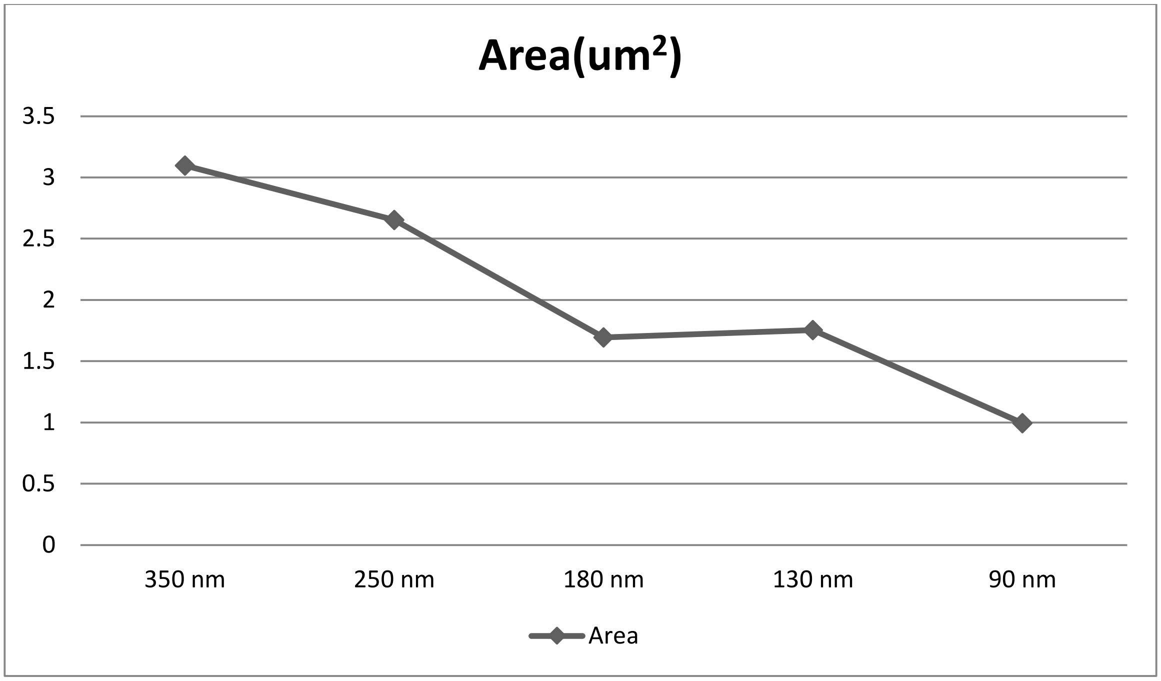

Fig. 3.26**:** Average power consumed by AND Gate using COWL in different NM

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

*************** Subcircuits *****************
.subckt Cell3 In Out Gnd Vdd
MNMOS_1 Out In Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3600 $y=-300
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=1.2u L=240n AS=1.08p PS=4.2u AD=1.08p PD=4.2u $ $x=3600 $y=900
$w=400 $h=600
.ends


***** Top Level *****
XCell3_2 A N_1 Gnd Vdd Cell3 $ $x=-700 $y=400 $w=1800 $h=1000
XCell3_3 B N_1 Gnd Vdd Cell3 $ $x=-700 $y=-1400 $w=1800 $h=1000
MNMOS_1 Output N_1 Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1000
$y=-1000 $w=400 $h=600
MPMOS_1 Output N_1 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1000
$y=200 $w=400 $h=600
Vv5 Vdd Gnd  DC 2.5 $ $x=-4300 $y=0 $w=400 $h=600
```

```
VV1 A Gnd  BIT({0011} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-1600 $y=0 $w=400 $h=600
VV2 B Gnd  BIT({0101} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-1600 $y=-1900 $w=400
$h=600
.PRINT   V(A) $ $x=-2350 $y=650 $w=1500 $h=300 $r=180
.PRINT   V(B) $ $x=-2350 $y=-1150 $w=1500 $h=300 $r=180
.PRINT   V(Output) $ $x=3350 $y=-250 $w=1500 $h=300 $r=180 $m

********* Simulation Settings - Analysis Section *********
.tran 0.1n 40n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

3.3ANALYSIS OF XOR GATE USING COWL

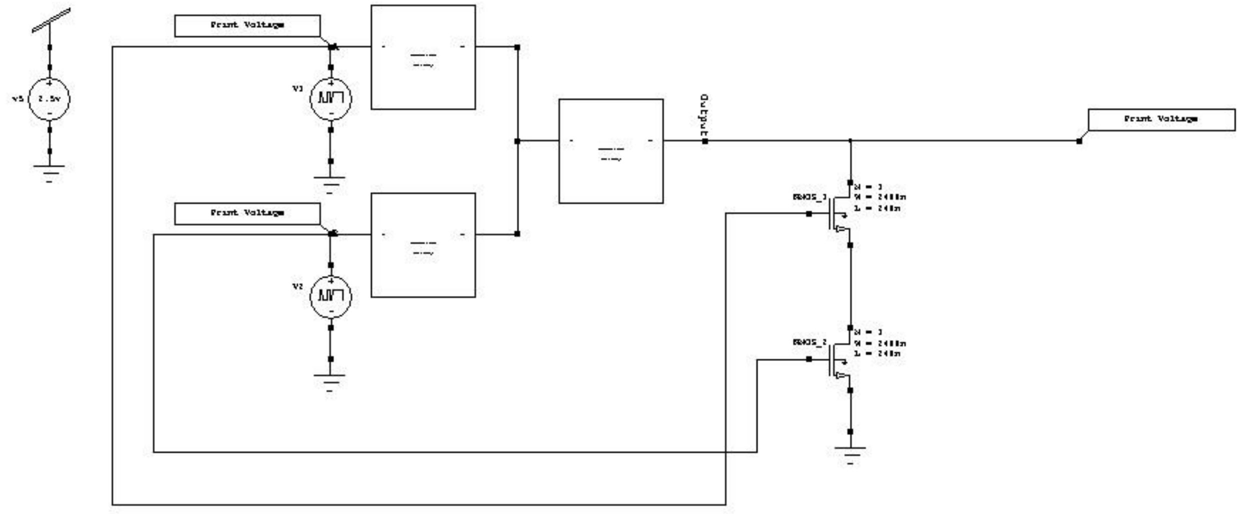

Fig. 3.27**:** XOR gate circuit diagram using COWL

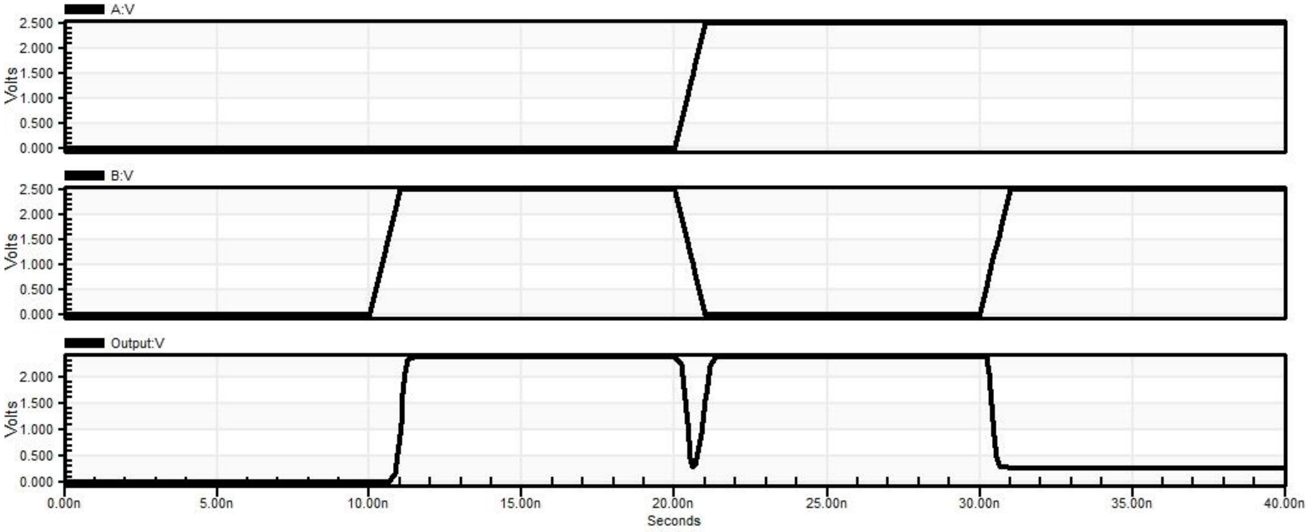

Fig. 3.28**:** XOR gate output waveform using COWL

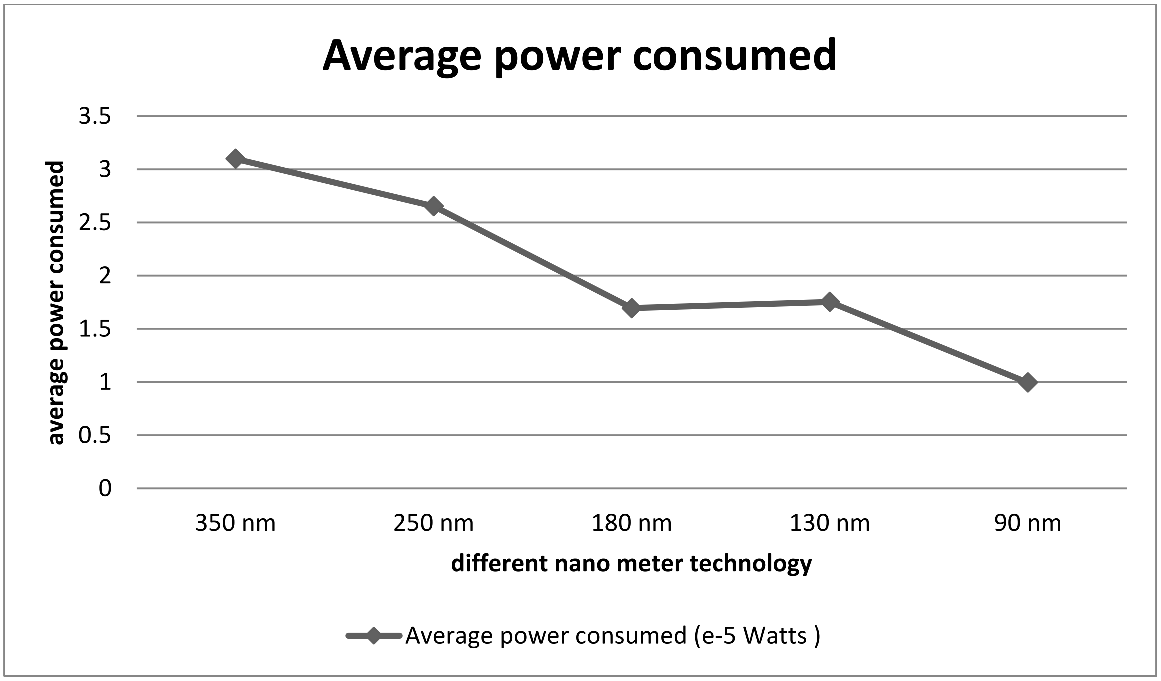

Fig. 3.29**:** Average power consumed by XOR Gate using COWL in different NM

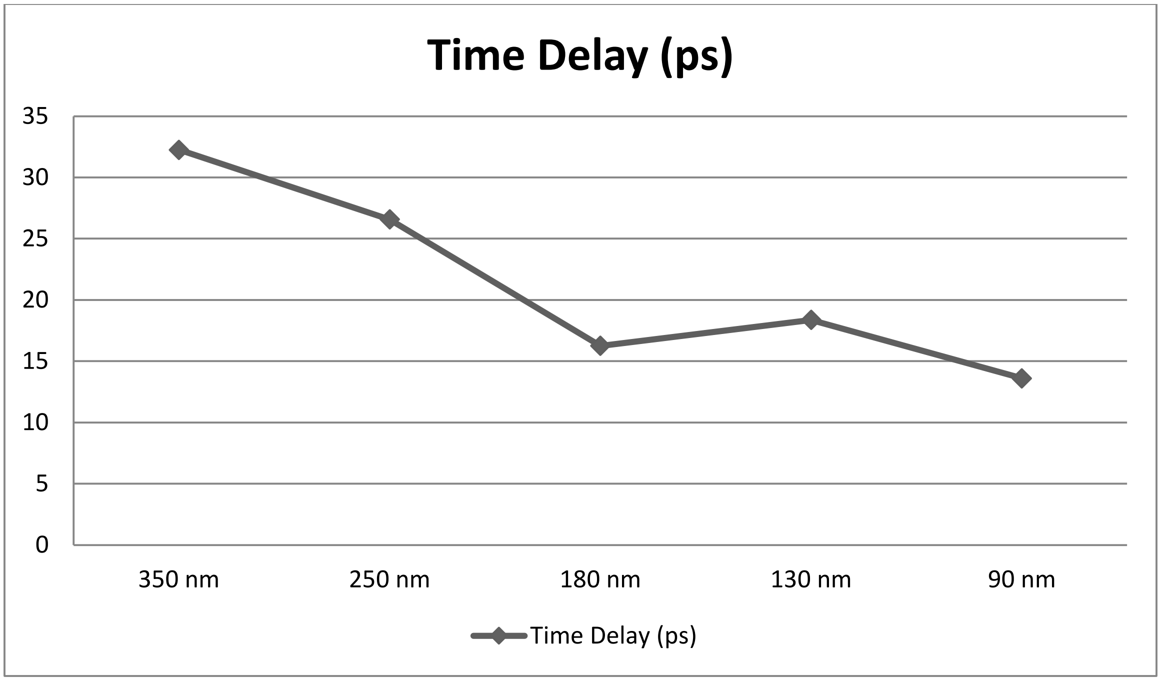

Fig. 3.30**:** Time delay of XOR Gate using COWL in different NM

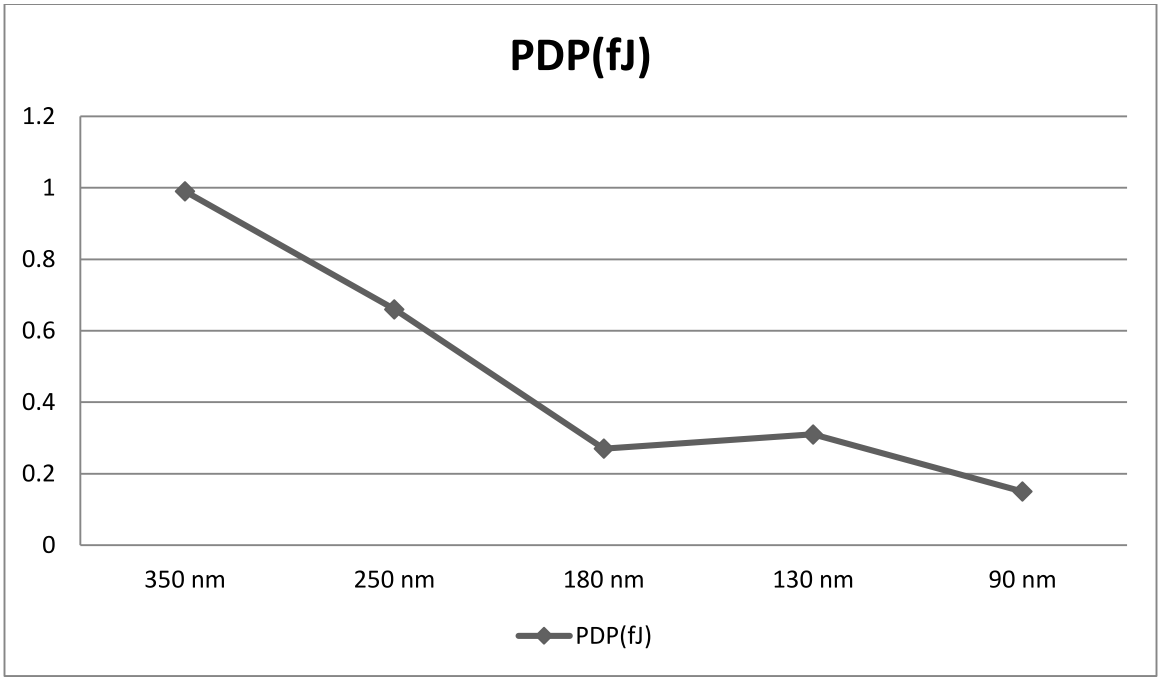

Fig. 3.31**:** Power Delay Product of XOR Gate using COWL in different NM

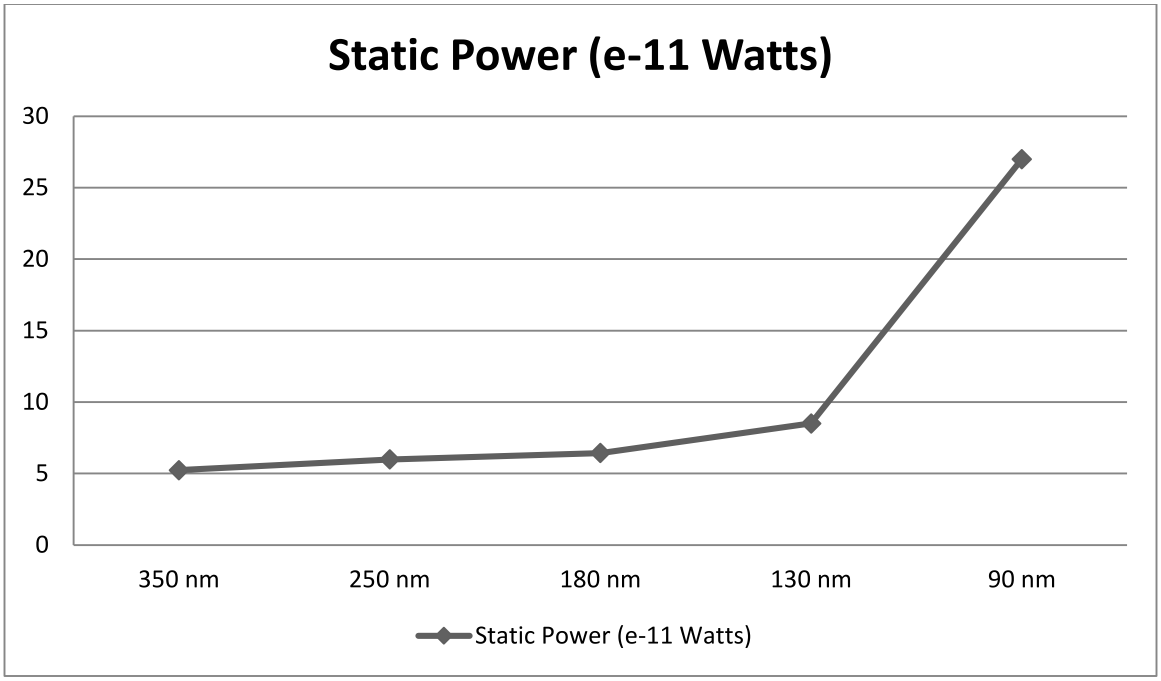

Fig. 3.32**:** Static Power of XOR Gate using COWL in different NM

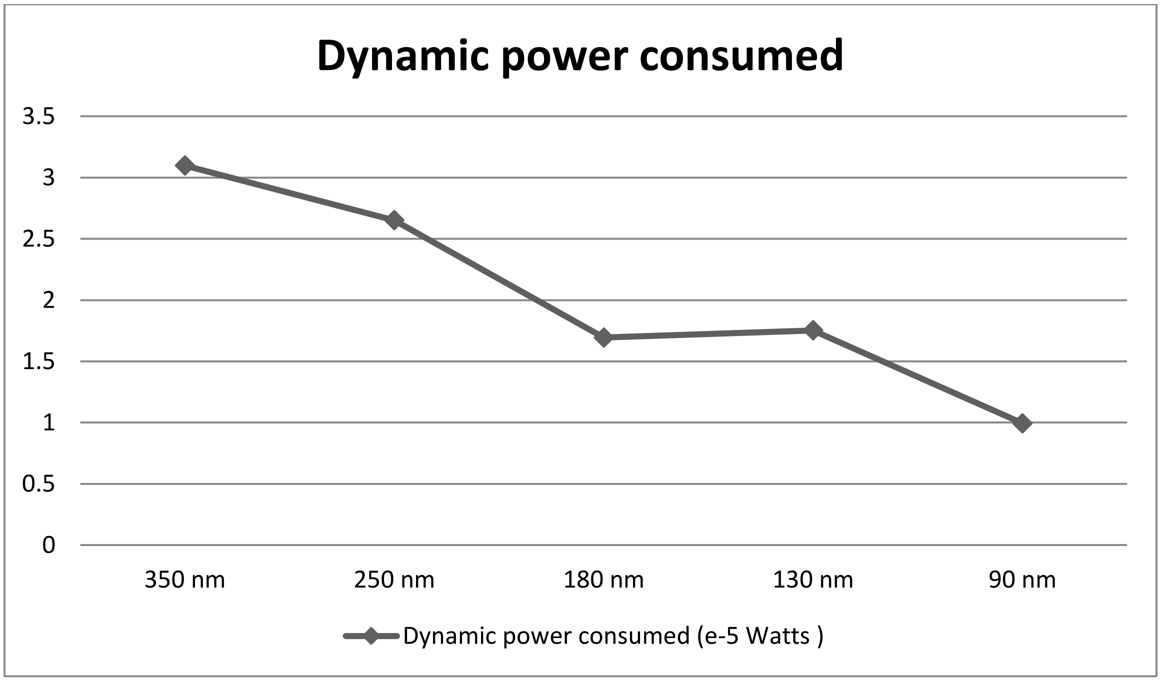

Fig. 3.33**:** Dynamic Power of XOR Gate using COWL in different NM

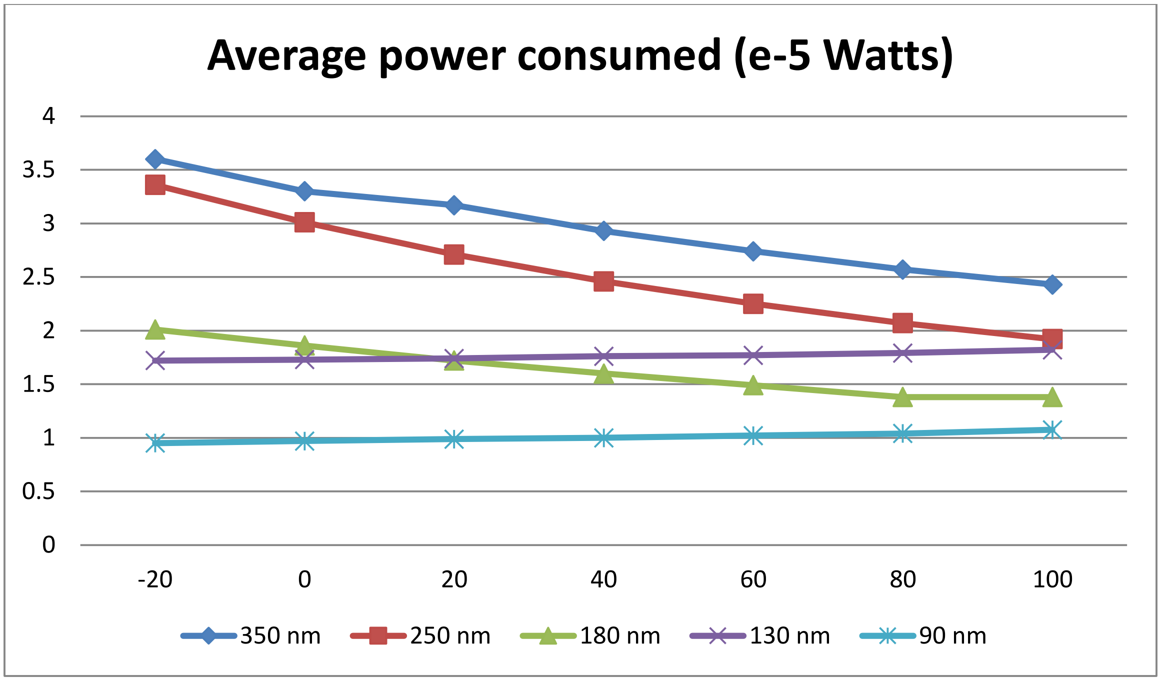

Fig. 3.34**:** Average power consumed by of XOR Gate using COWL in different NM and in different Temperature

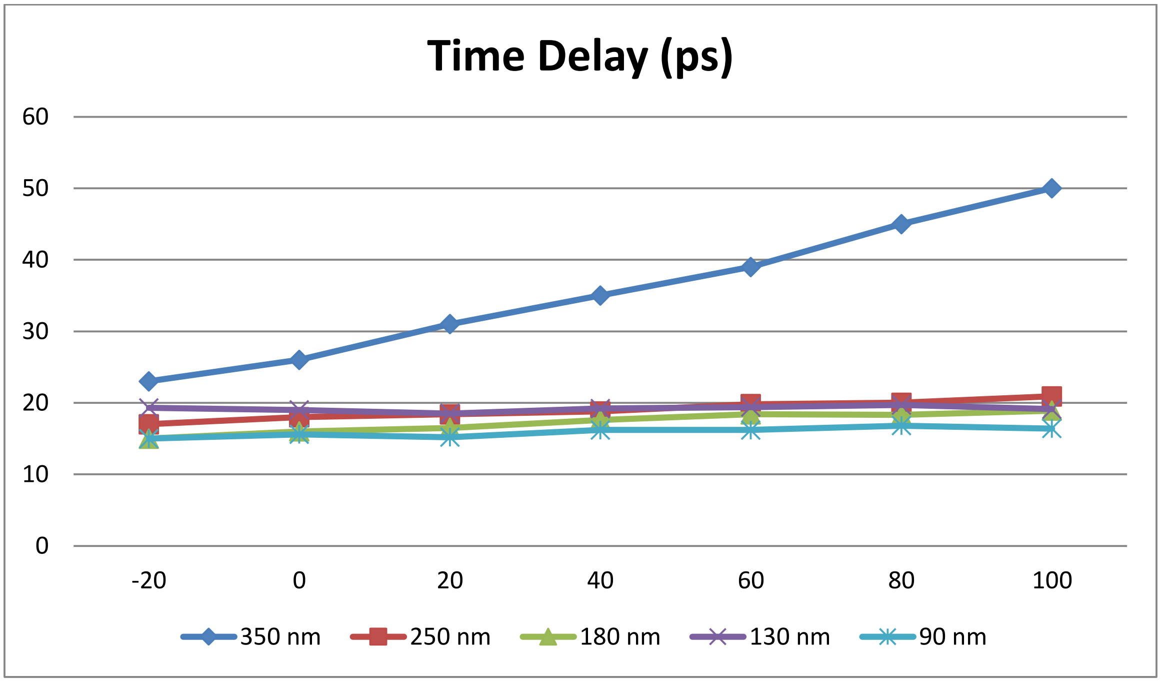

Fig.3.35**:** Time Delay of XOR Gate using COWL in different NM and in different Temperature

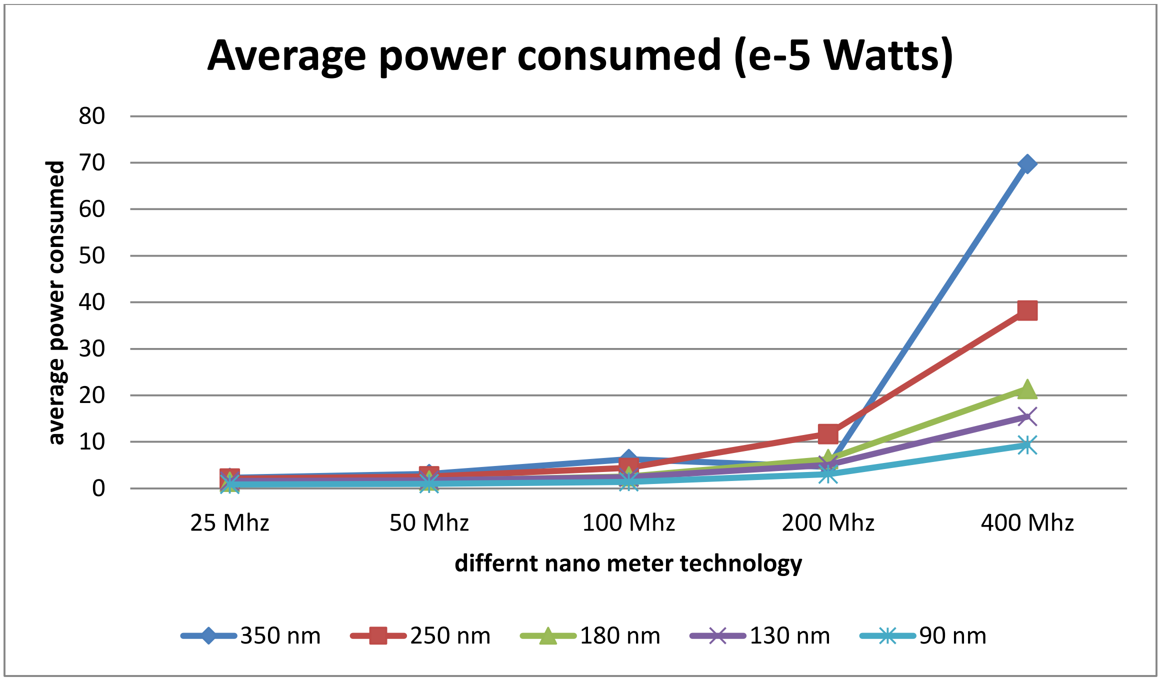

Fig. 3.36: Average power consumed by of XOR Gate using COWL in different NM and in different Frequency

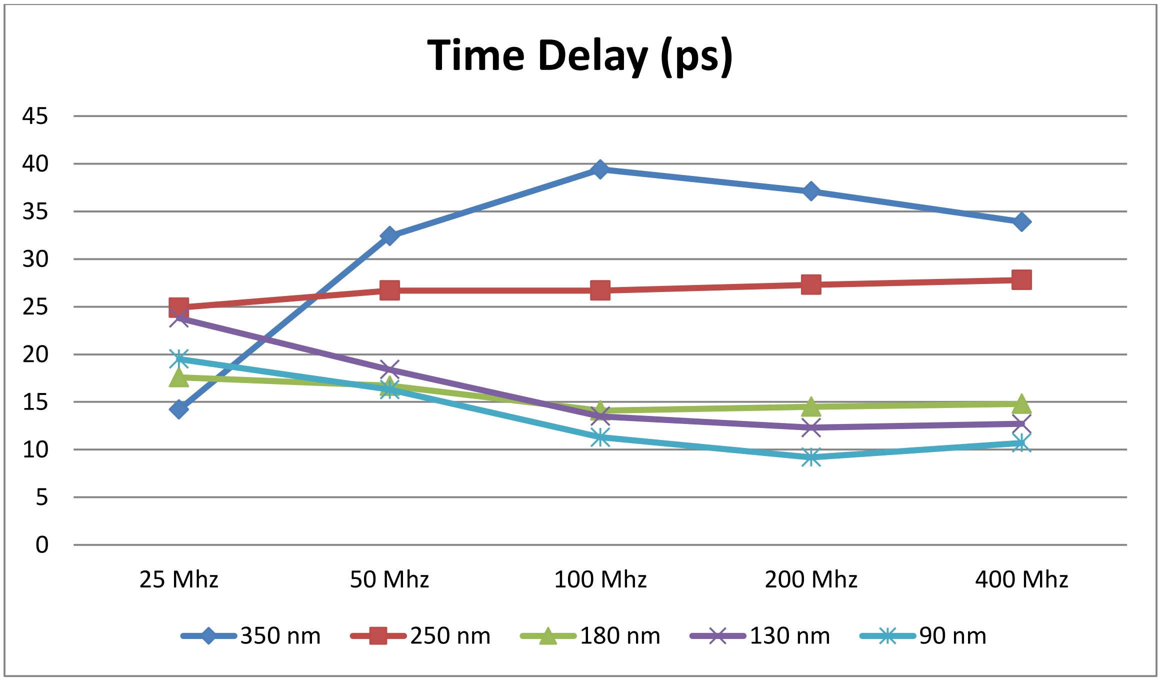

Fig. 3.37: Time Delay of XOR Gate using COWL in different NM and in different Frequency

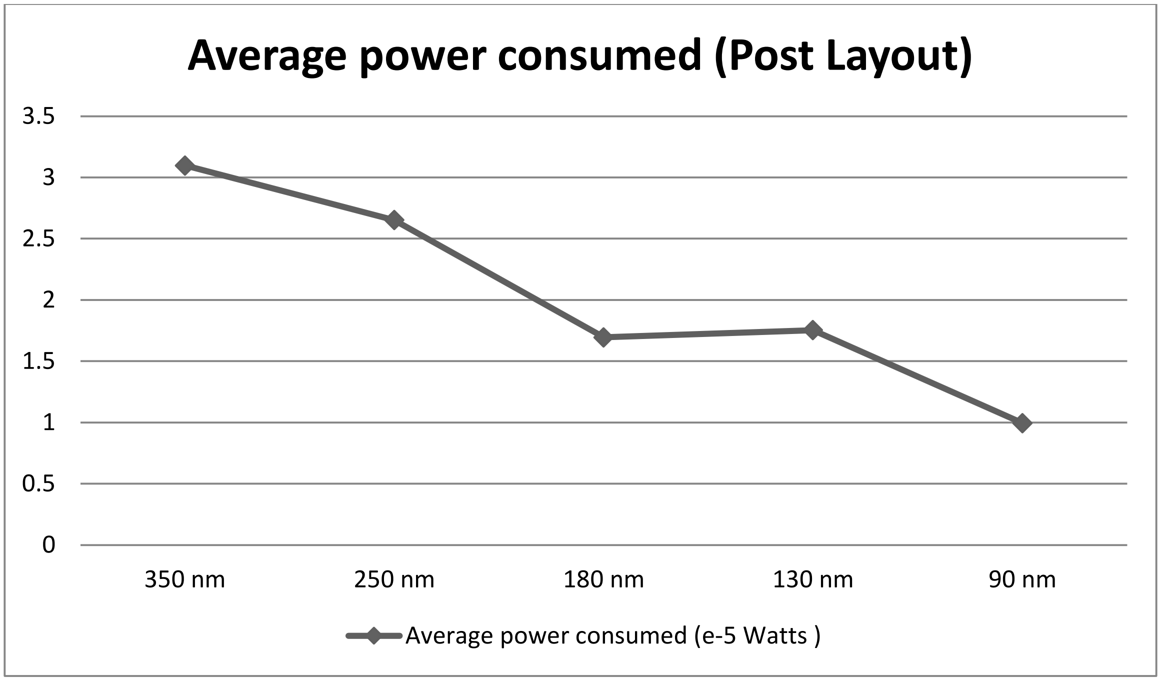

Fig. 3.38**:** Average power consumed by XOR Gate using COWL in different NM (Post Layout)

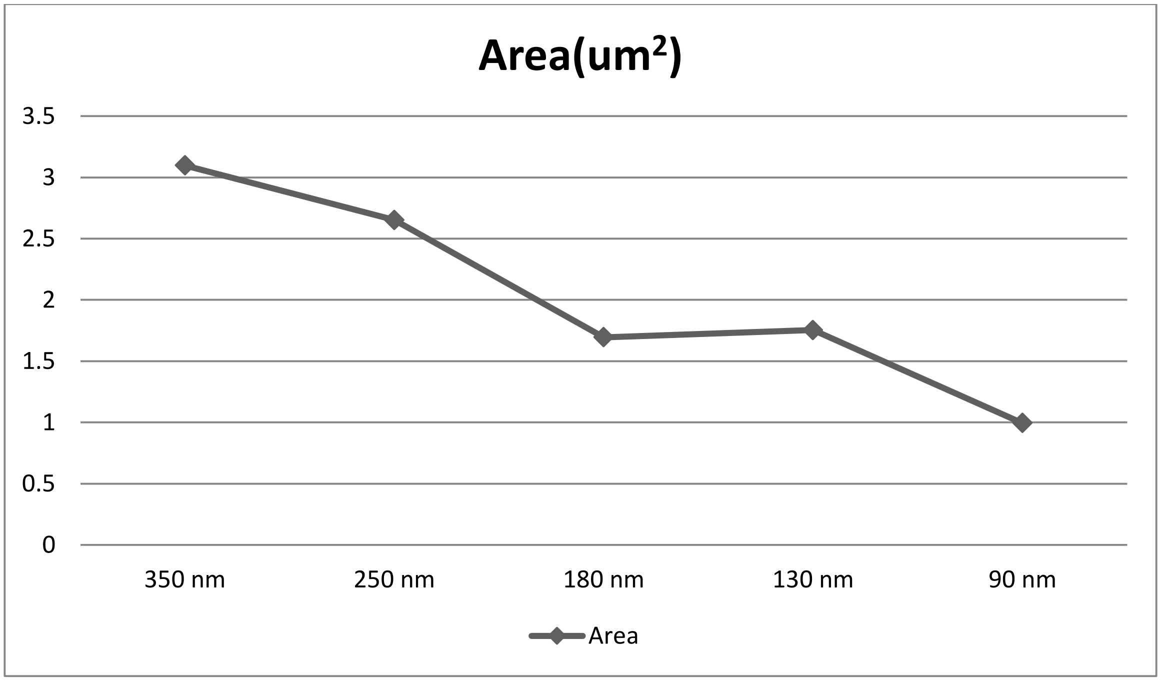

Fig. 3.39**:** Average power consumed by XOR Gate using COWL in different NM

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

*************** Subcircuits *****************
.subckt Cell0 In Out Gnd Vdd
MNMOS_1 Out In Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=-100 $y=-600
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=900n L=240n AS=810f PS=3.6u AD=810f PD=3.6u $ $x=-100 $y=600
$w=400 $h=600
.ends

.subckt Cell3 In Out Gnd Vdd
MNMOS_1 Out In Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3600 $y=-300
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=1.2u L=240n AS=1.08p PS=4.2u AD=1.08p PD=4.2u $ $x=3600 $y=900
$w=400 $h=600
.ends


***** Top Level *****
XCell0_1 A N_1 Gnd Vdd Cell0 $ $x=3200 $y=2700 $w=1800 $h=1000
XCell0_2 B N_1 Gnd Vdd Cell0 $ $x=3200 $y=900 $w=1800 $h=1000
XCell3_1 N_1 Output Gnd Vdd Cell3 $ $x=5000 $y=1800 $w=1800 $h=1000
MNMOS_1 Output A N_2 0 NMOS W=2.4u L=240n AS=2.16p PS=6.6u AD=2.16p PD=6.6u $ $x=7100 $y=1200
$w=400 $h=600
MNMOS_2 N_2 B Gnd 0 NMOS W=2.4u L=240n AS=2.16p PS=6.6u AD=2.16p PD=6.6u $ $x=7100 $y=-200
$w=400 $h=600
Vv5 Vdd Gnd  DC 2.5 $ $x=-400 $y=2300 $w=400 $h=600
VV1 A Gnd  BIT({0011} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=2300 $y=2300 $w=400 $h=600
VV2 B Gnd  BIT({0101} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=2300 $y=400 $w=400 $h=600
.PRINT   V(A) $ $x=1550 $y=2950 $w=1500 $h=300 $r=180
.PRINT   V(B) $ $x=1550 $y=1150 $w=1500 $h=300 $r=180
.PRINT   V(Output) $ $x=10250 $y=2050 $w=1500 $h=300 $r=180 $m

********* Simulation Settings - Analysis Section *********
.tran 0.1n 40n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

3.4ANALYSIS OFMAJORITY GATE USING COWL

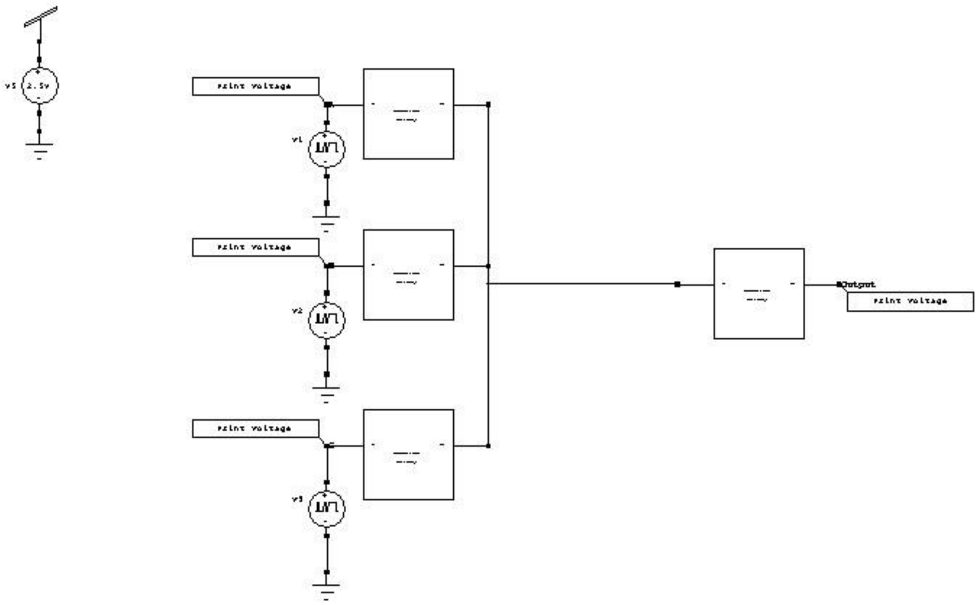

Fig. 3.40**:** Majority gate circuit diagram using COWL

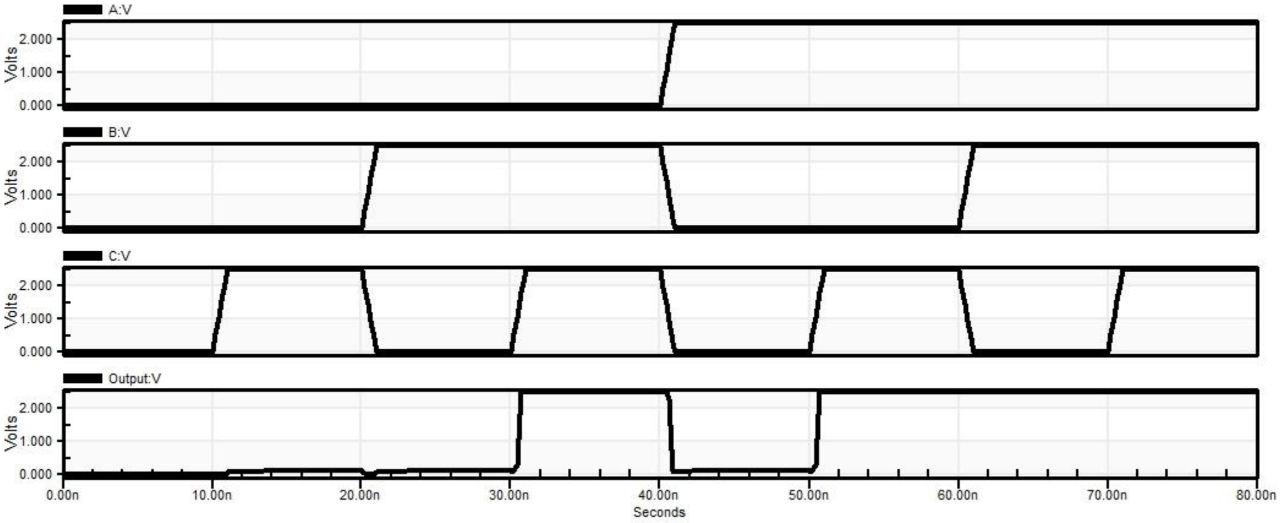

Fig. 3.41**:** Majority gate output waveform using COWL

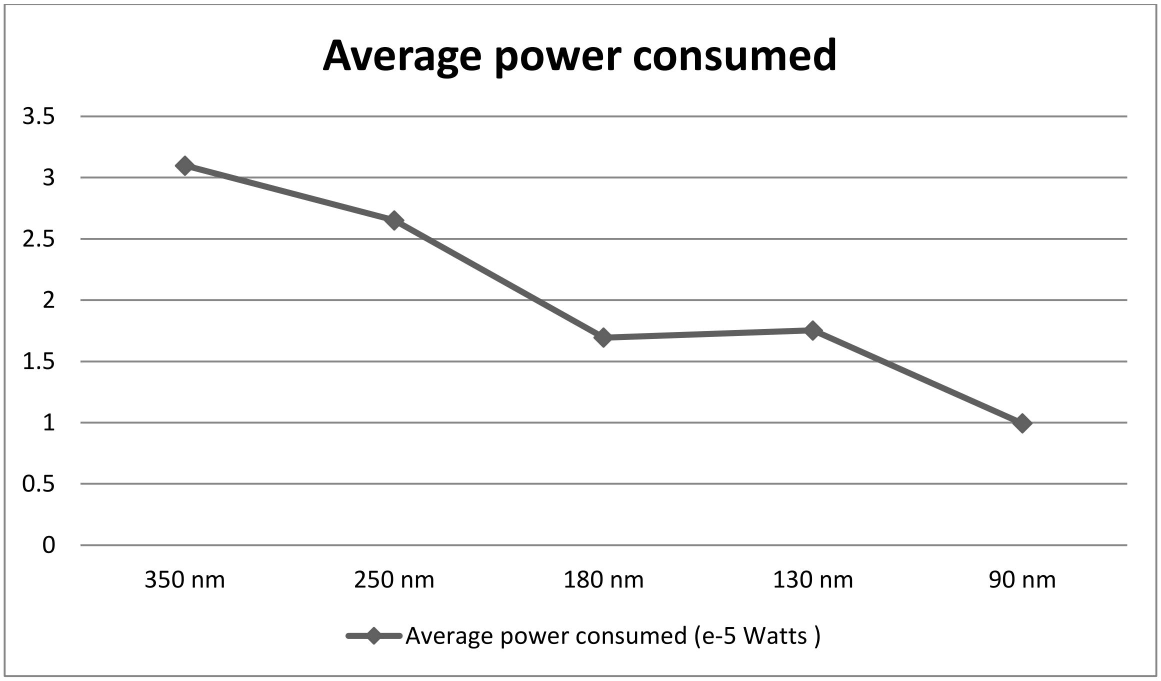

Fig. 3.42**:** Average power consumed by Majority Gate using COWL in different NM

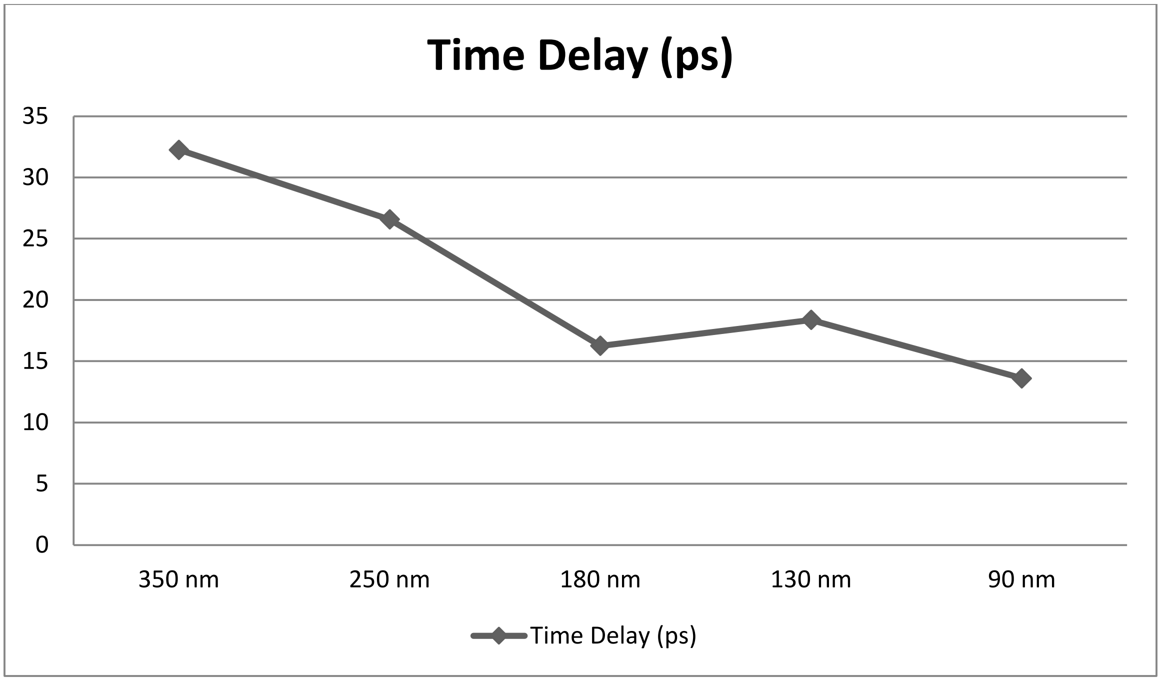

Fig. 3.43**:** Time delay of Majority Gate using COWL in different NM

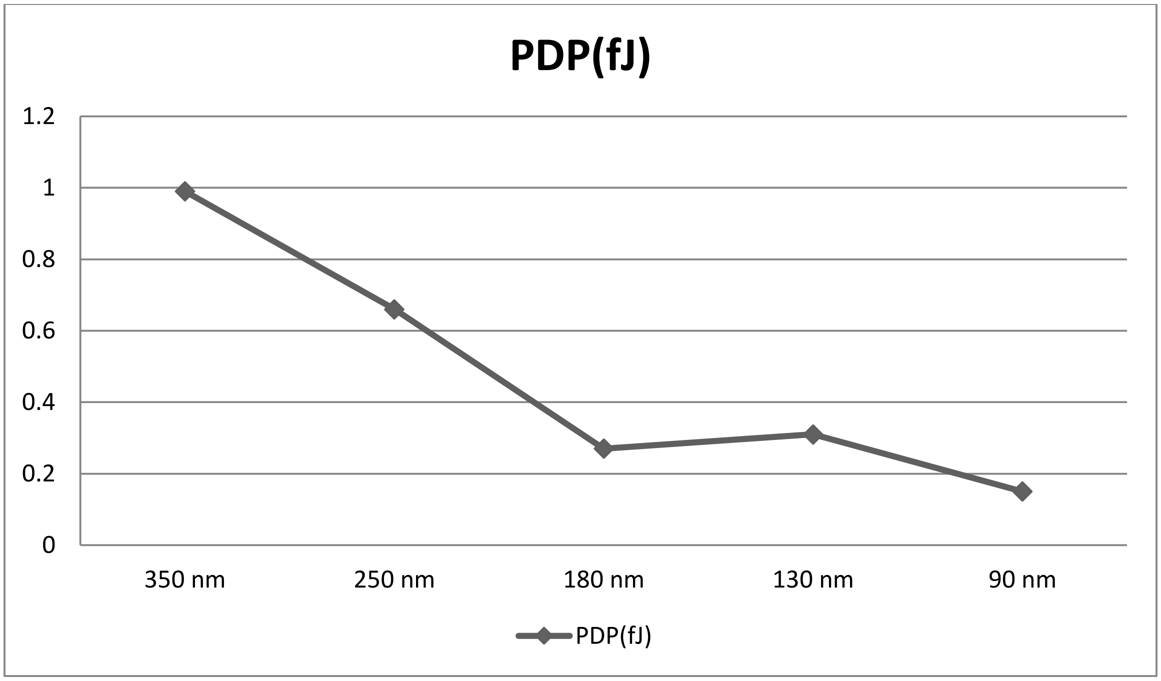

Fig. 3.44**:** Power Delay Product ofMajority Gate using COWL in different NM

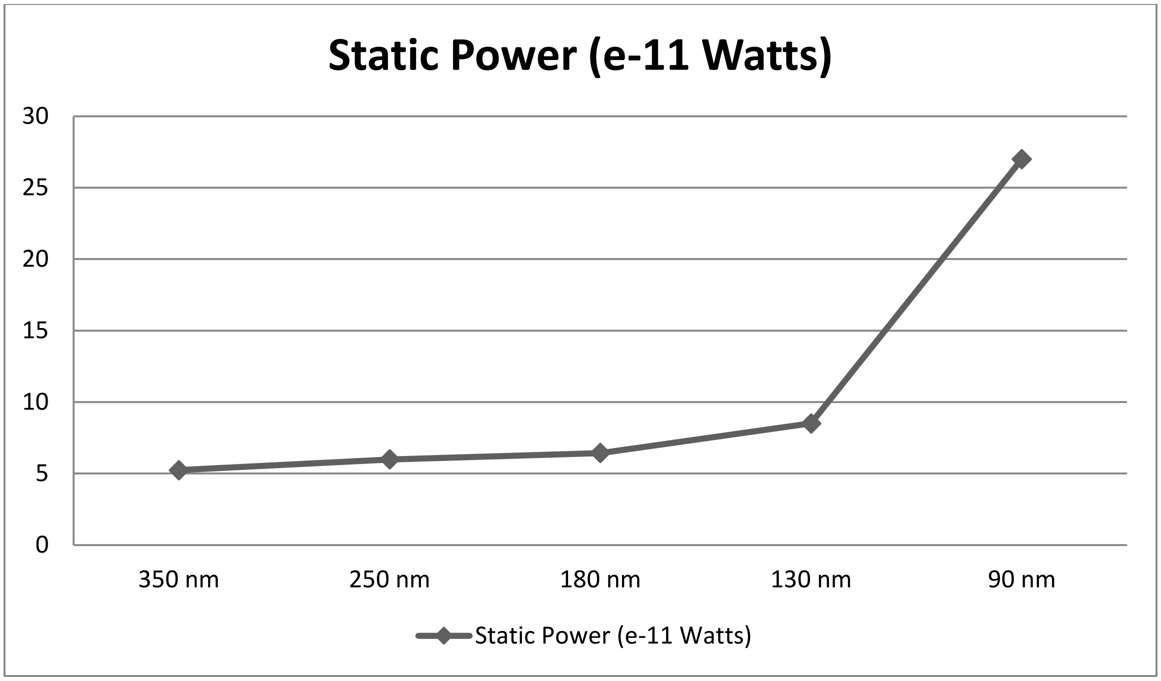

Fig. 3.45**:** Static Power of Majority Gate using COWL in different NM

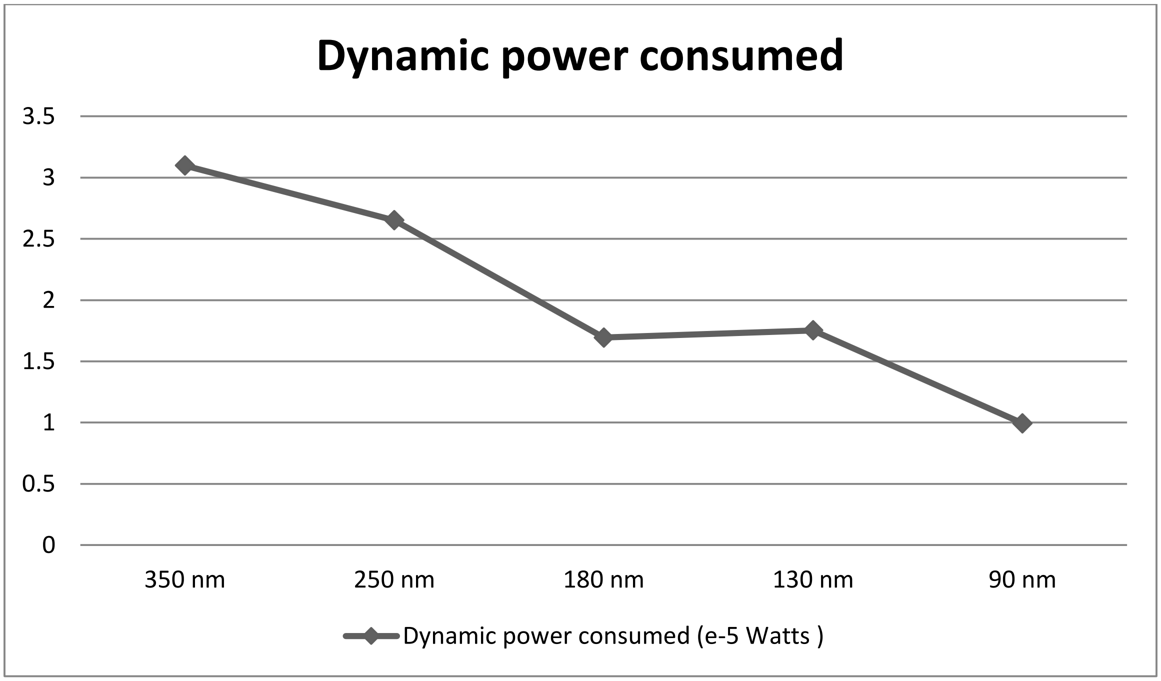

Fig. 3.46**:** Dynamic Power of Majority Gate using COWL in different NM

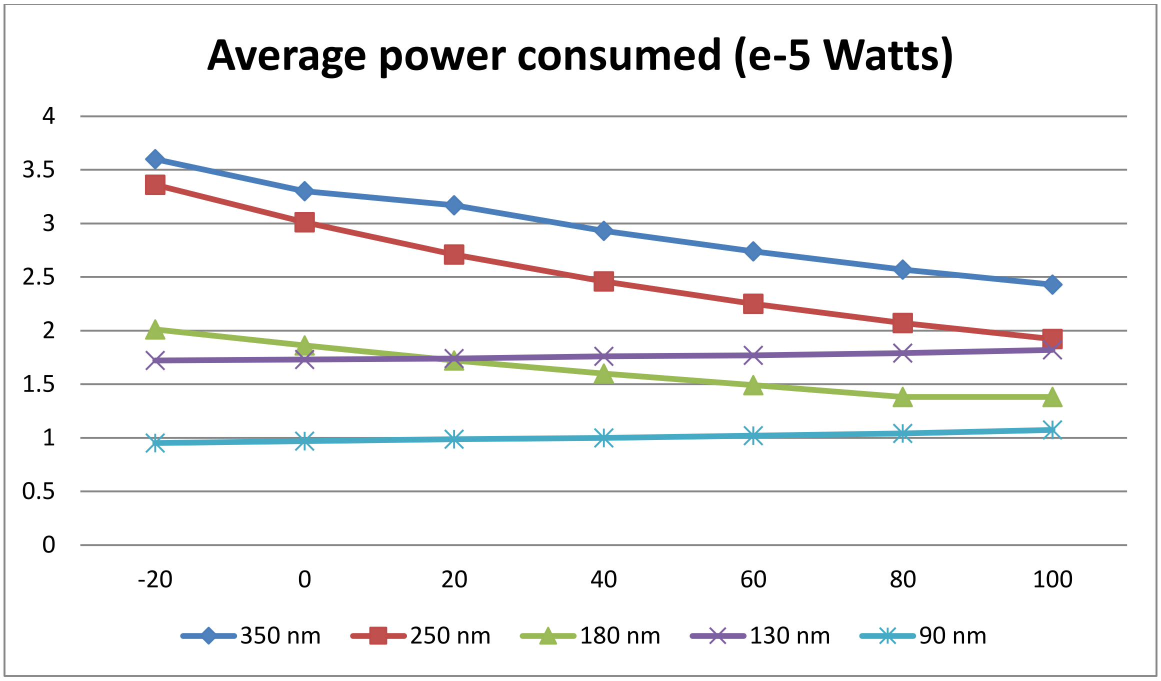

Fig. 3.47**:** Average power consumed by of Majority Gate using COWL in different NM and in different Temperature

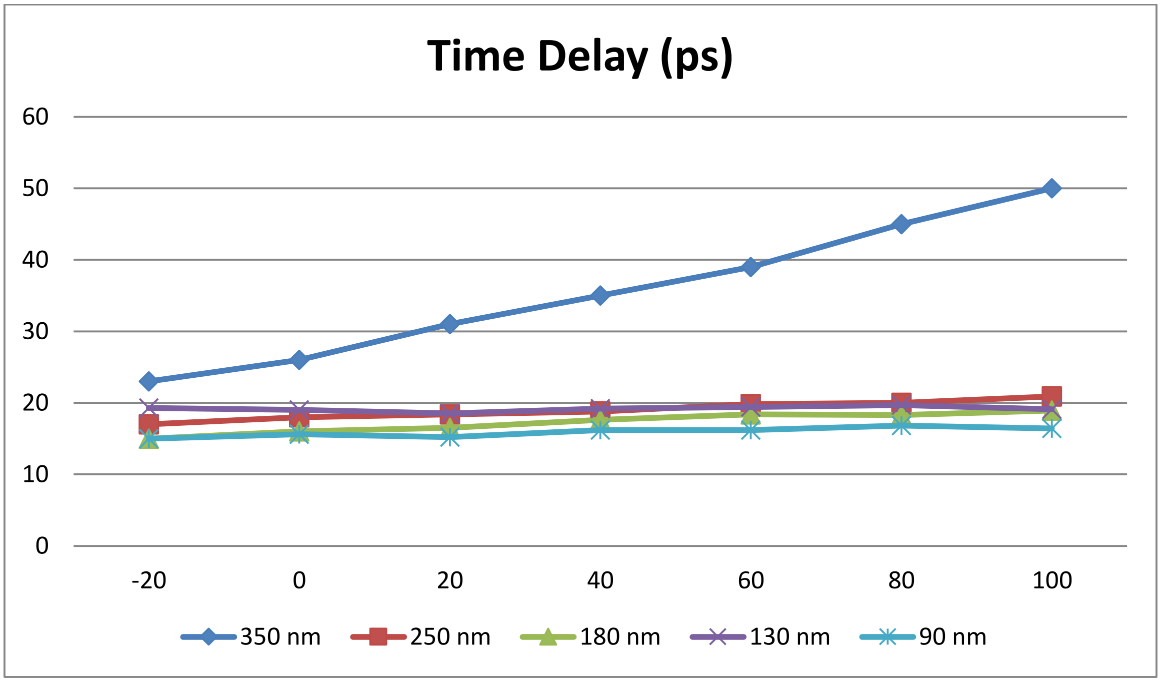

Fig. 3.48**:** Time Delay of Majority Gate using COWL in different NM and in different Temperature

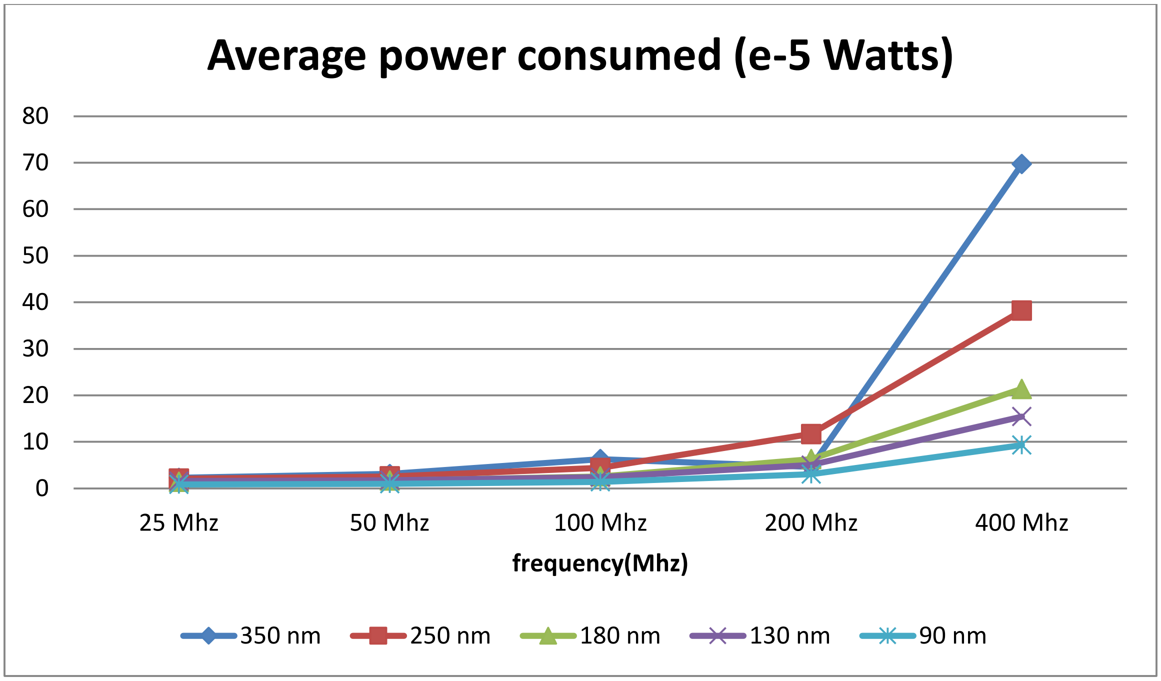

Fig. 3.49**:** Average power consumed by of Majority Gate using COWL in different NM and in different Frequency

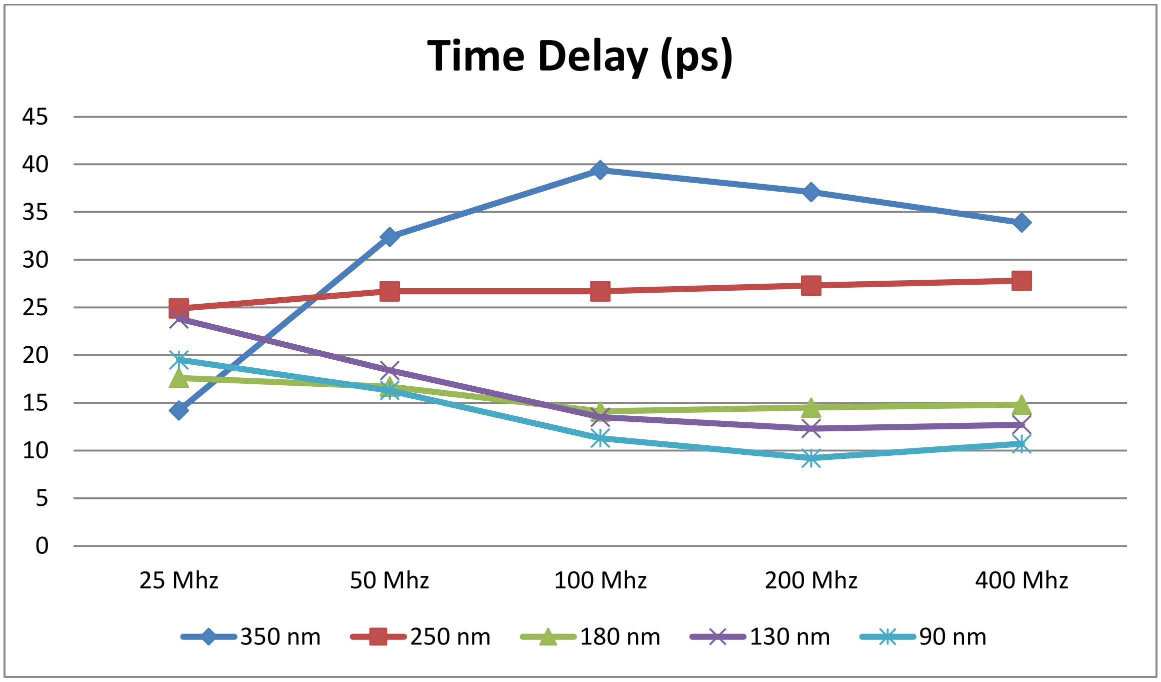

Fig. 3.50**:** Time Delay of Majority Gate using COWL in different NM and in different Frequency

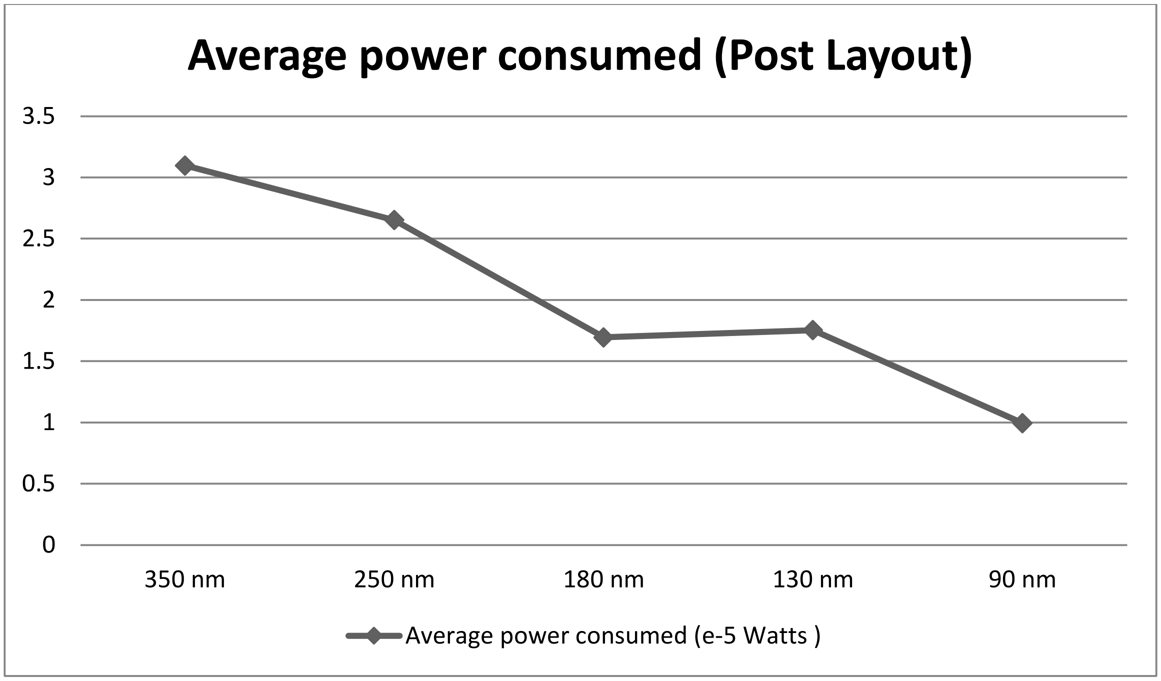

Fig. 3.51**:** Average power consumed by Majority Gate using COWL in different NM (Post Layout)

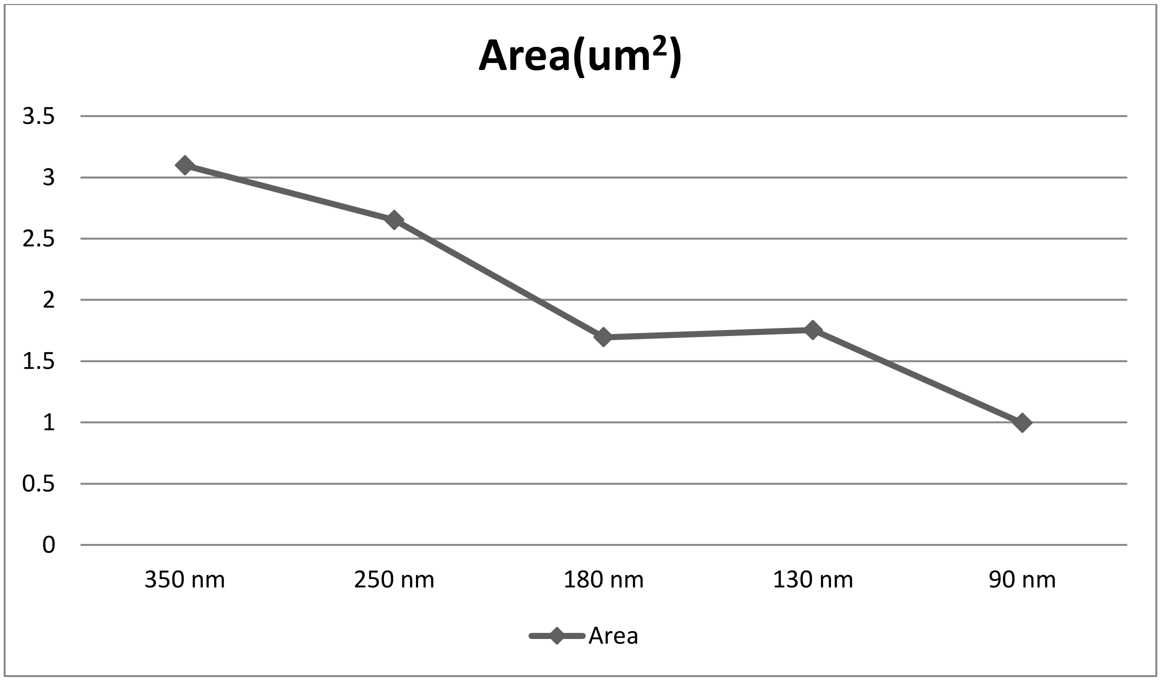

Fig. 3.52**:** Average power consumed by Majority Gate using COWL in different NM

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

*************** Subcircuits *****************
.subckt Cell0 In Out Gnd Vdd
MNMOS_1 Out In Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=-100 $y=-600
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=900n L=240n AS=810f PS=3.6u AD=810f PD=3.6u $ $x=-100 $y=600
$w=400 $h=600
.ends


***** Top Level *****
XCell0_1 A N_1 Gnd Vdd Cell0 $ $x=7800 $y=7400 $w=1800 $h=1000
XCell0_2 B N_1 Gnd Vdd Cell0 $ $x=7800 $y=5600 $w=1800 $h=1000
XCell0_3 C N_1 Gnd Vdd Cell0 $ $x=7800 $y=3600 $w=1800 $h=1000
XCell0_4 N_1 Output Gnd Vdd Cell0 $ $x=11700 $y=5400 $w=1800 $h=1000
Vv5 Vdd Gnd  DC 2.5 $ $x=3700 $y=7700 $w=400 $h=600
VV1 A Gnd  BIT({00001111} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=6900 $y=7000 $w=400
$h=600
VV2 B Gnd  BIT({00110011} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=6900 $y=5100 $w=400
$h=600
VV3 C Gnd  BIT({01010101} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=6900 $y=3000 $w=400
$h=600
.PRINT   V(A) $ $x=6150 $y=7650 $w=1500 $h=300 $r=180
.PRINT   V(B) $ $x=6150 $y=5850 $w=1500 $h=300 $r=180
.PRINT   V(C) $ $x=6150 $y=3850 $w=1500 $h=300 $r=180
.PRINT   V(Output) $ $x=13350 $y=5350 $w=1500 $h=300

********* Simulation Settings - Analysis Section *********
.tran 0.1n 80n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

IMPLEMENTATION OF CMOS Output Wired logic (COWL)

3.5ANALYSISOF FULL ADDERUSINGCOWL

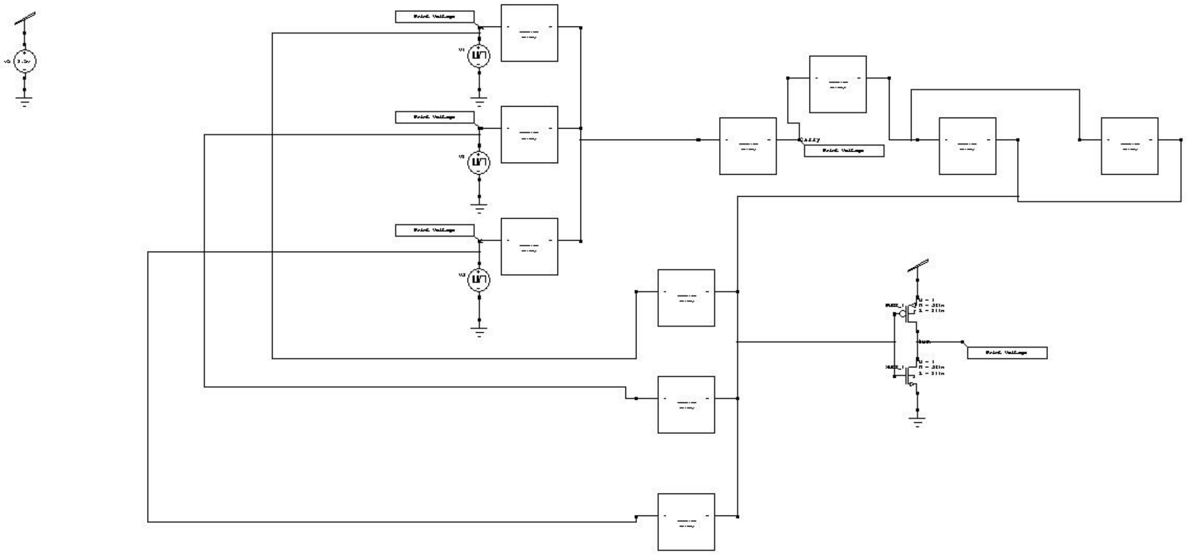

Fig.3.53**:** Full Adder circuit diagram using COWL

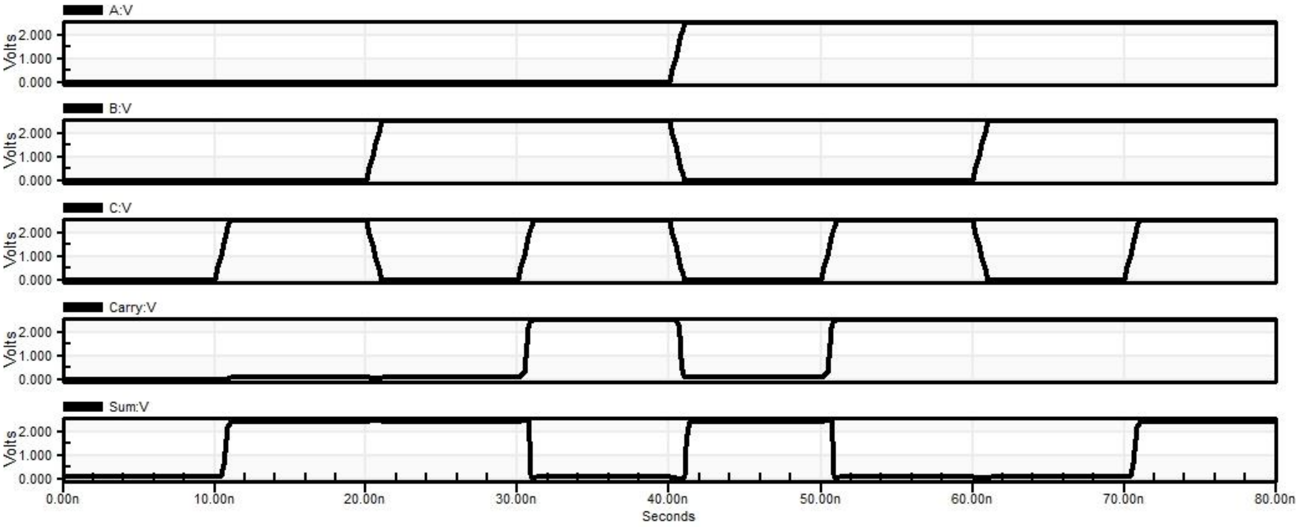

Fig. 3.54**:** Full Adder output waveform using COWL

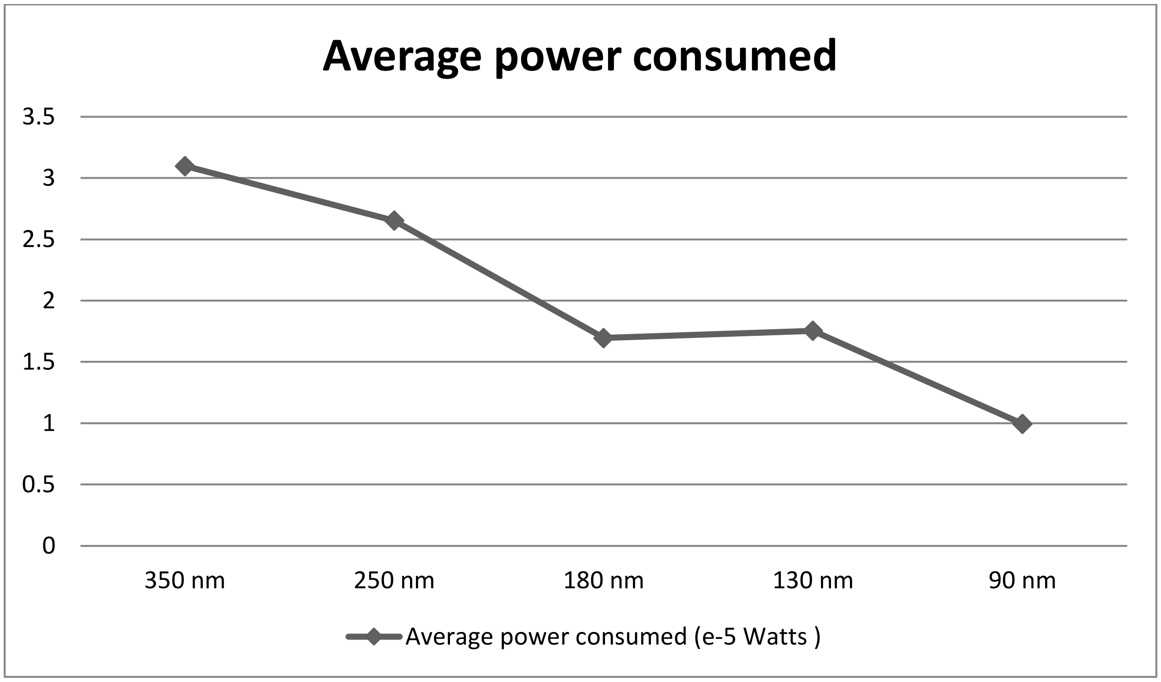

Fig. 3.55**:** Average power consumed by Full Adder using COWL in different NM

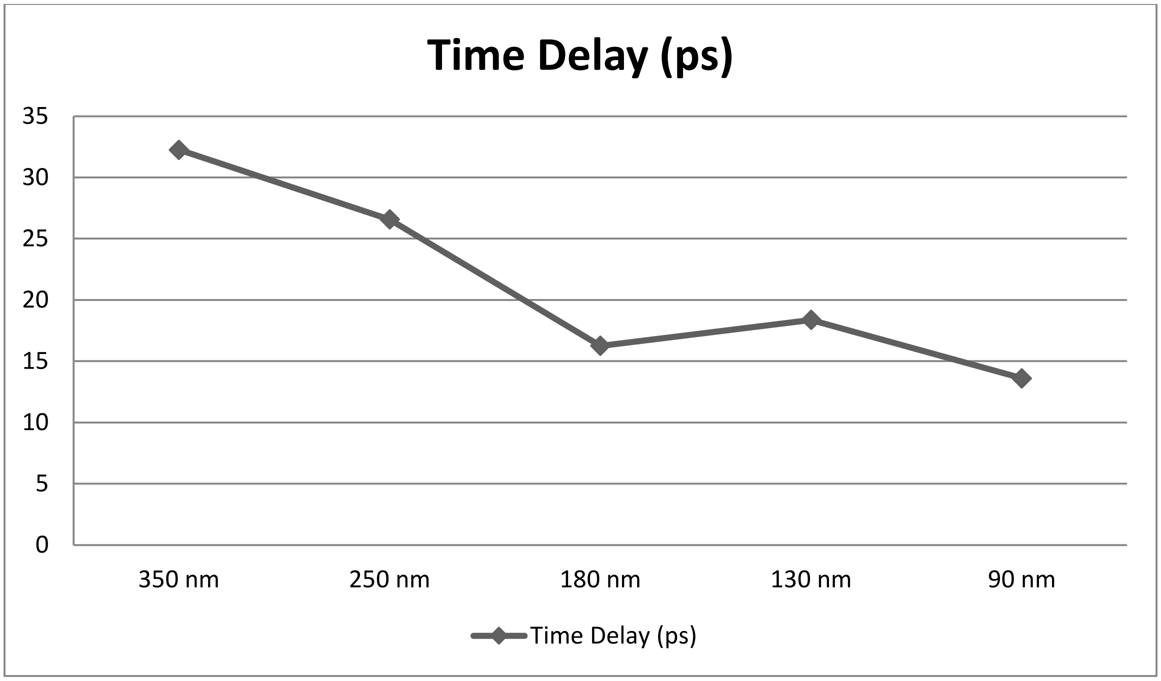

Fig. 3.56**:** Time delay of Full Adder using COWL in different NM

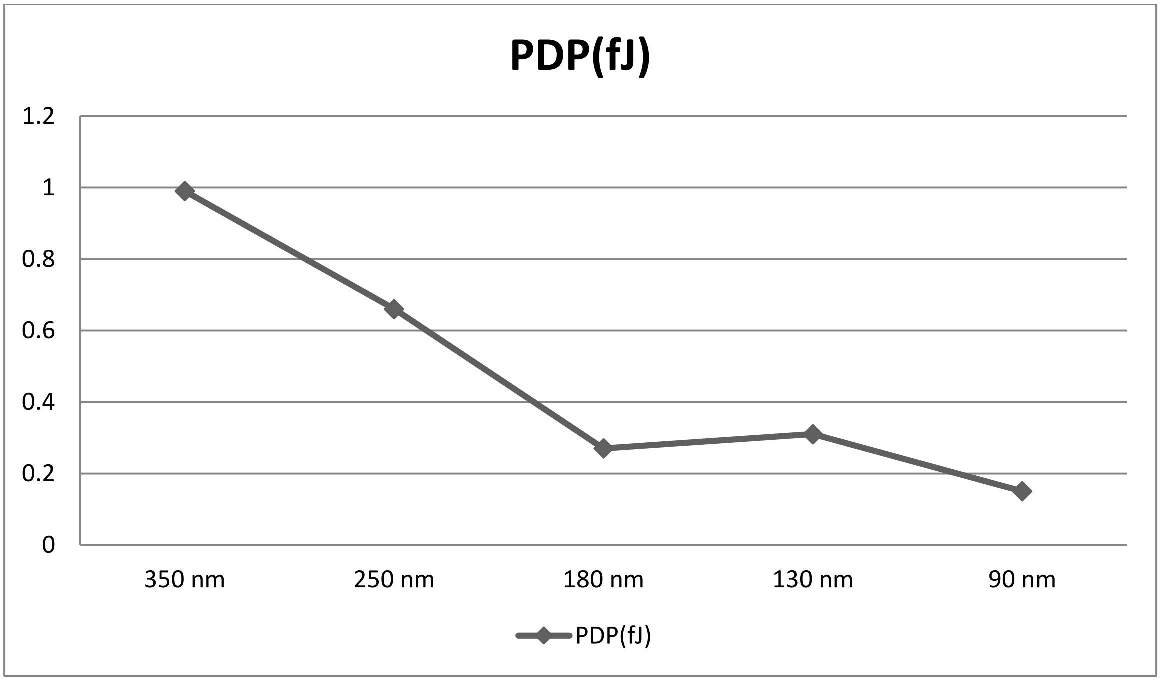

Fig. 3.57**:** Power Delay Product of Full Adder using COWL in different NM

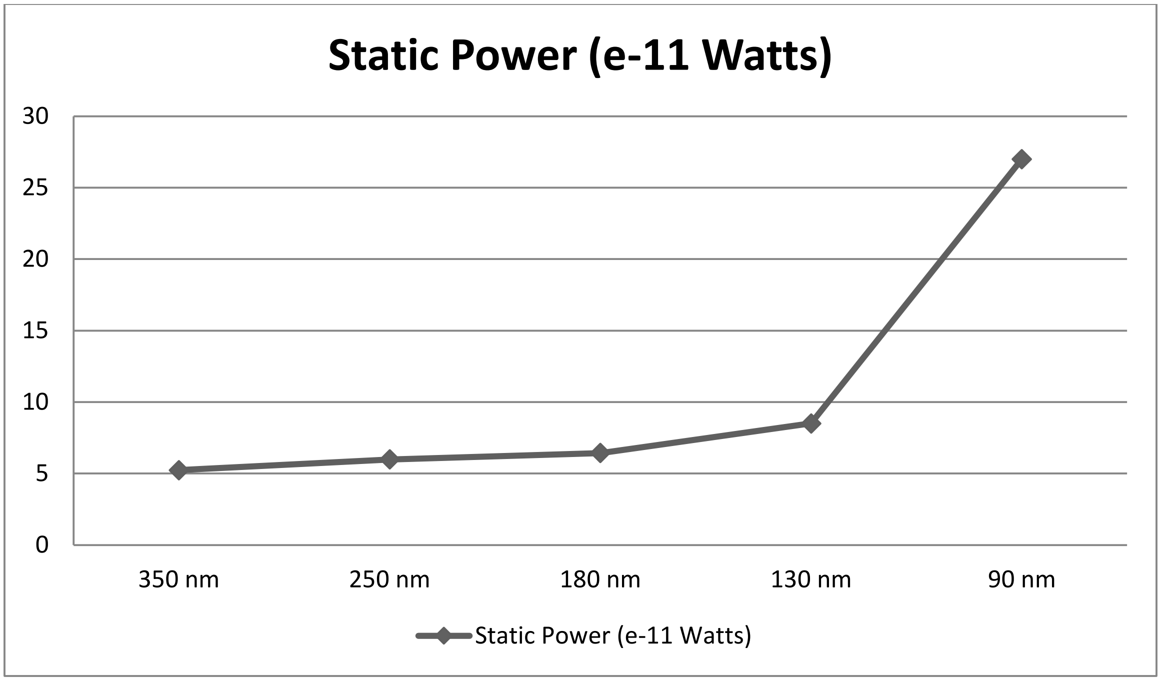

Fig. 3.58**:** Static Power of Full Adder using COWL in different NM

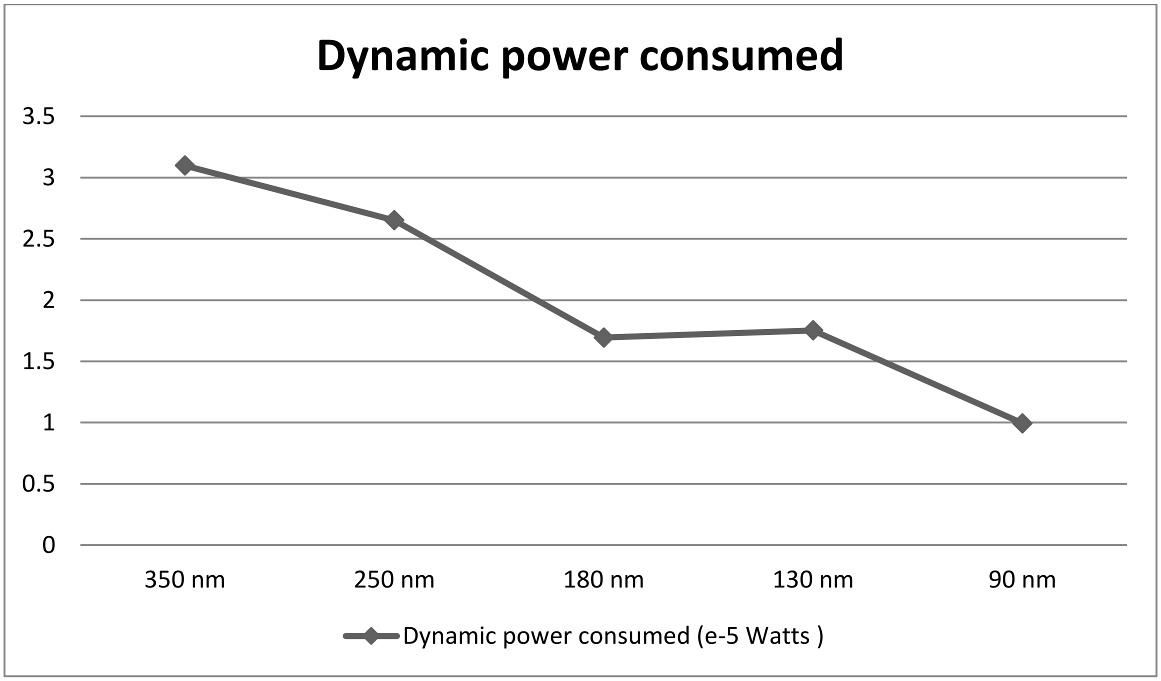

Fig. 3.59**:** Dynamic Power of Full Adder using COWL in different NM

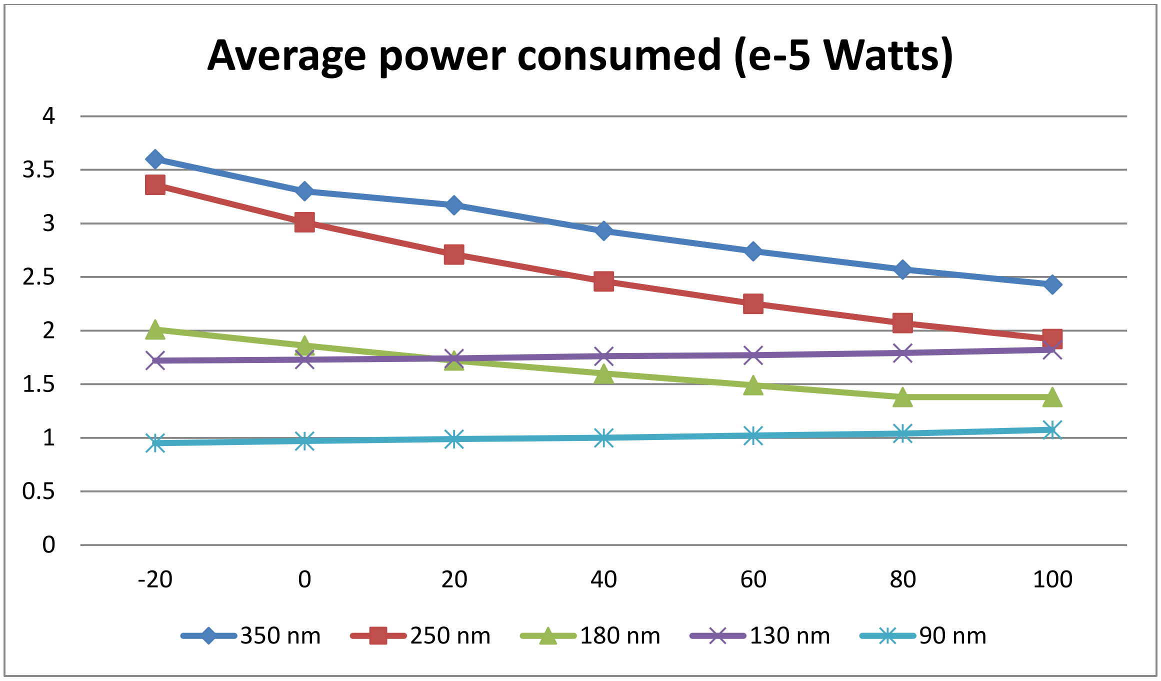

Fig. 3.60**:** Average power consumed by of Full Adder using COWL in different NM and in different Temperature

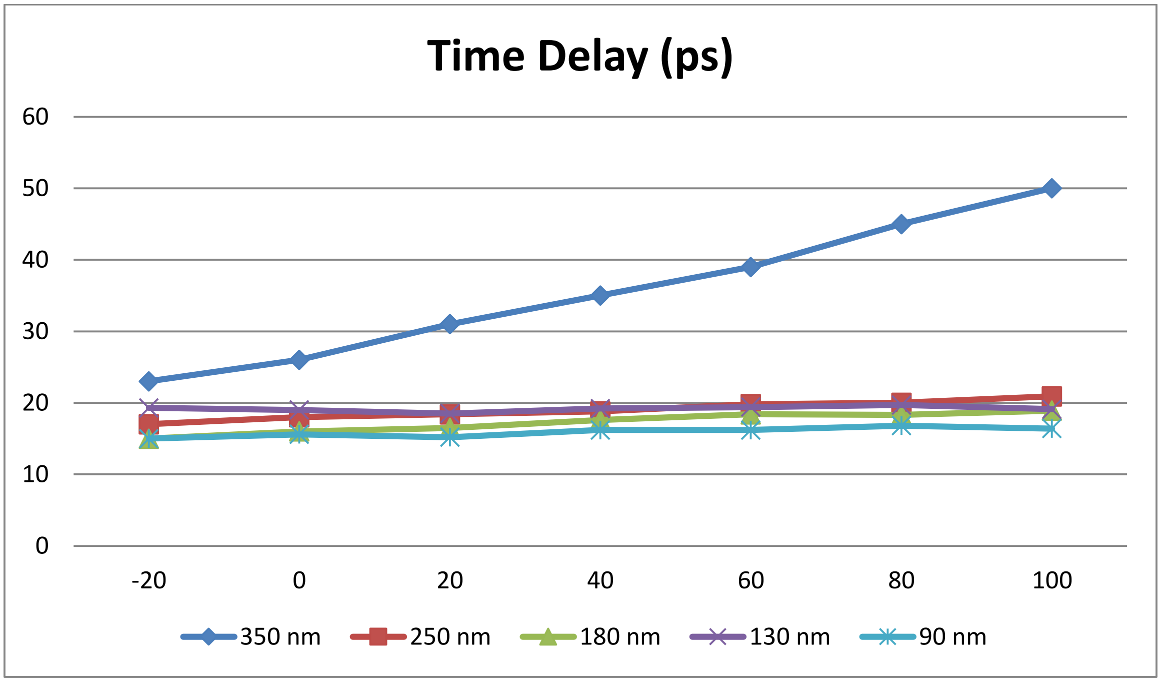

Fig. 3.61**:** Time Delay of Full Adder using COWL in different NM and in different Temperature

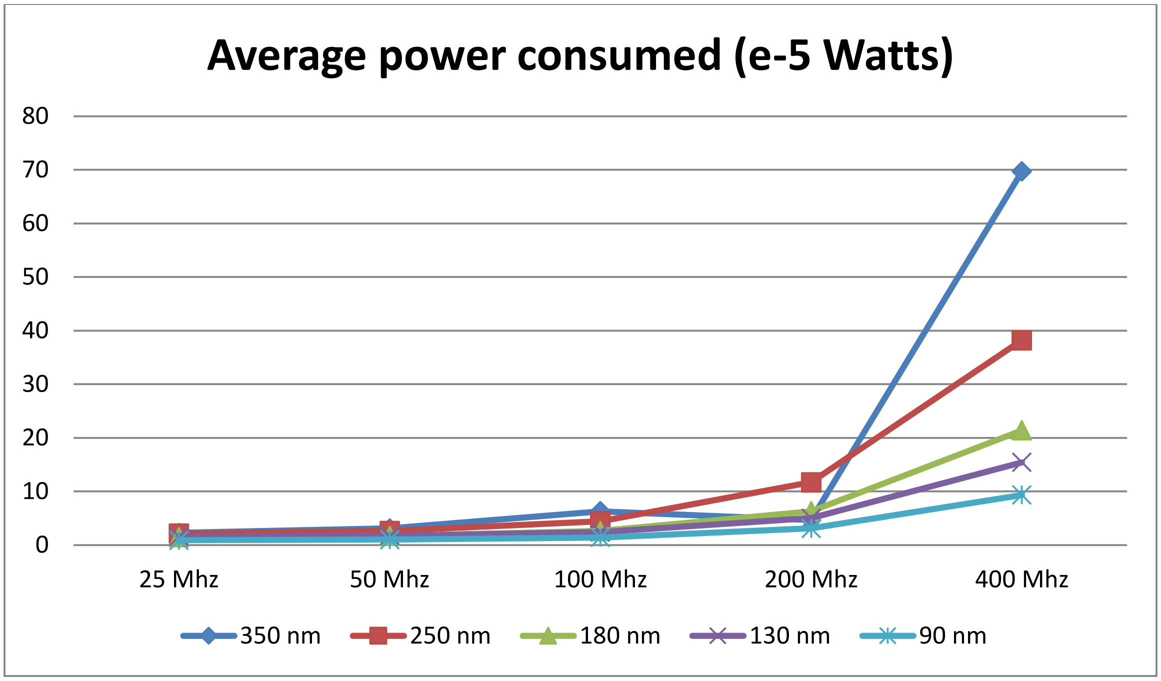

Fig. 3.62**:** Average power consumed by of Full Adder using COWL in different NM and in different Frequency

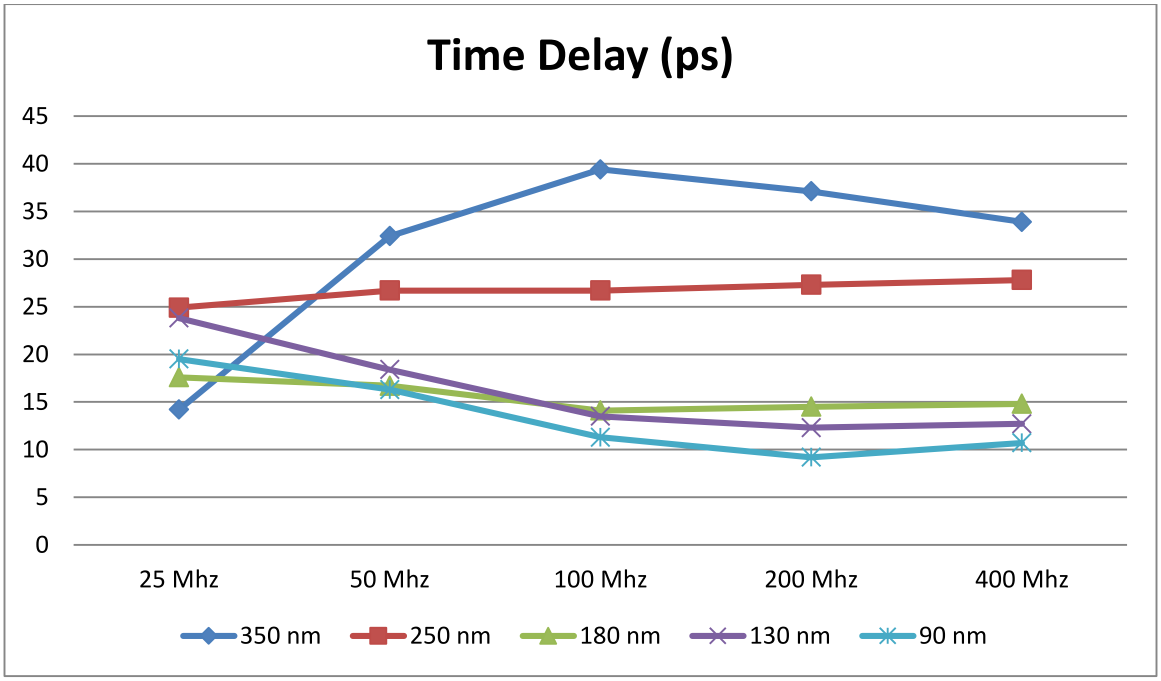

Fig. 3.63**:** Time Delay of Full Adder using COWL in different NM and in different Frequency

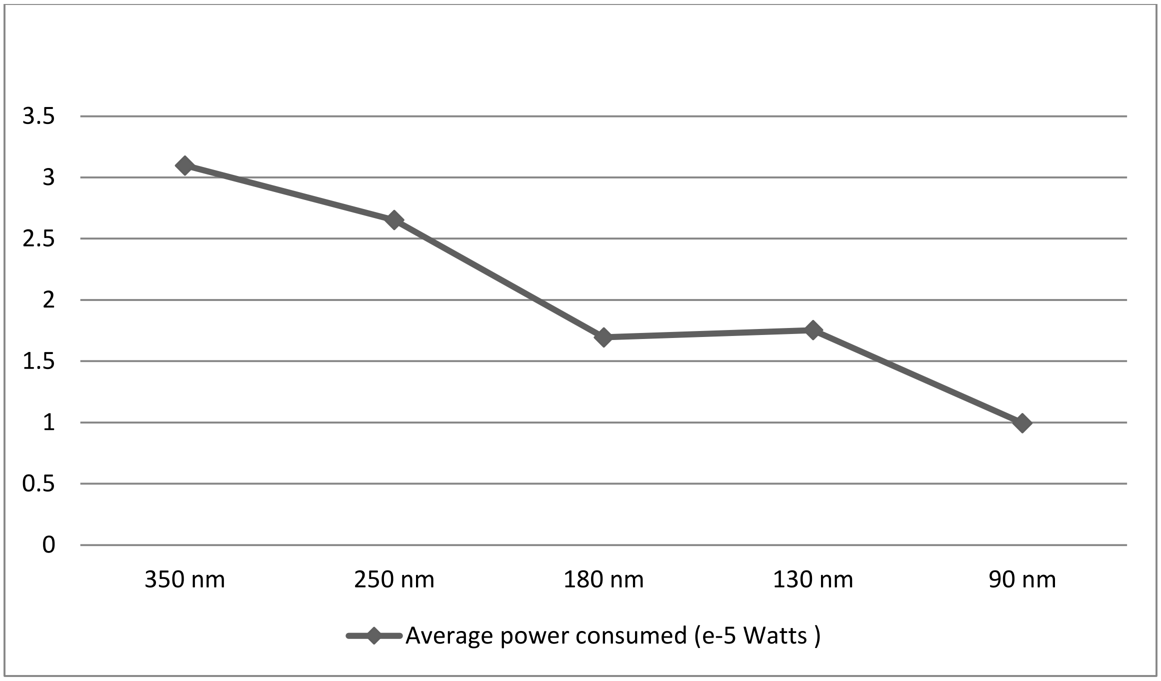

Fig. 3.64**:** Average power consumed by Full Adder using COWL in different NM (Post Layout)

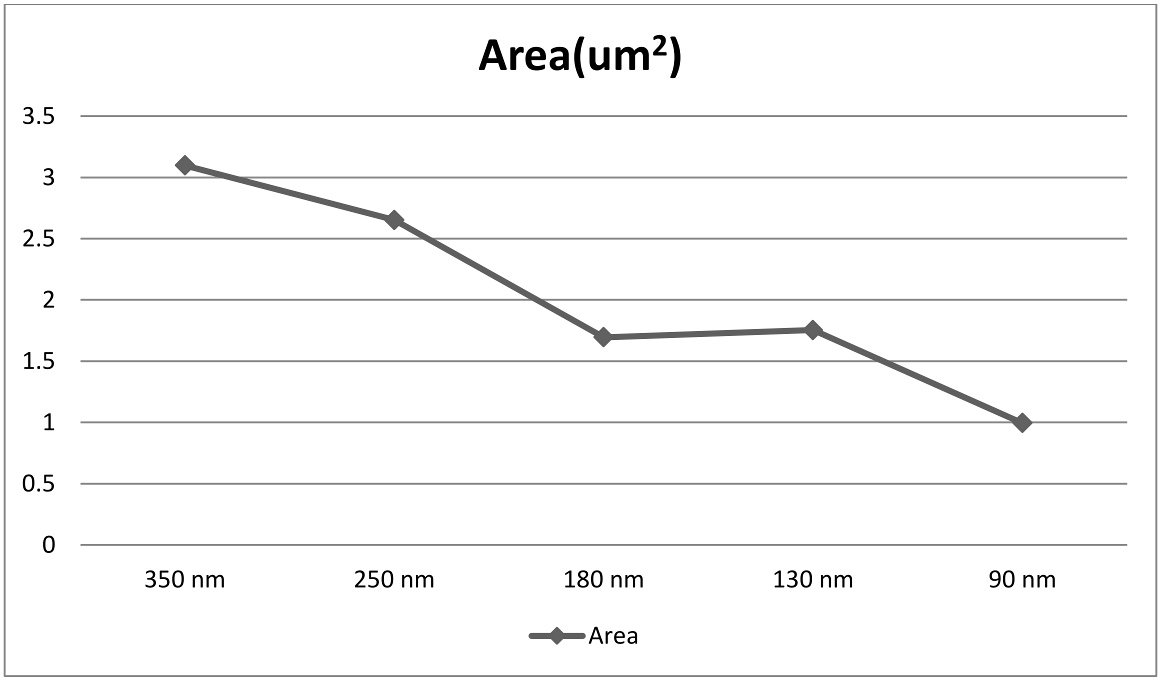

Fig. 3.65**:** Average power consumed by Full Adder using COWL in different NM

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra
Krishnapura\TSMC 250nm CMOS.md"

*************** Subcircuits *****************
.subckt Cell0 In Out Gnd Vdd
MNMOS_1 Out In Gnd Gnd NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=-100 $y=-600
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=900n L=240n AS=810f PS=3.6u AD=810f PD=3.6u $ $x=-100 $y=600
$w=400 $h=600
.ends
***** Top Level *****
XCell0_1 A N_1 Gnd Vdd Cell0 $ $x=500 $y=400 $w=1800 $h=1000
XCell0_2 B N_1 Gnd Vdd Cell0 $ $x=500 $y=-1400 $w=1800 $h=1000
XCell0_3 C N_1 Gnd Vdd Cell0 $ $x=500 $y=-3400 $w=1800 $h=1000
XCell0_4 N_1 Carry Gnd Vdd Cell0 $ $x=4400 $y=-1600 $w=1800 $h=1000
XCell0_5 A N_3 Gnd Vdd Cell0 $ $x=3300 $y=-4300 $w=1800 $h=1000
XCell0_6 B N_3 Gnd Vdd Cell0 $ $x=3300 $y=-6200 $w=1800 $h=1000
XCell0_7 C N_3 Gnd Vdd Cell0 $ $x=3300 $y=-8300 $w=1800 $h=1000
XCell0_8 N_2 N_3 Gnd Vdd Cell0 $ $x=8300 $y=-1600 $w=1800 $h=1000
XCell0_9 N_2 N_3 Gnd Vdd Cell0 $ $x=11200 $y=-1600 $w=1800 $h=1000
XCell0_10 Carry N_2 Gnd Vdd Cell0 $ $x=6000 $y=-500 $w=1800 $h=1000
MNMOS_1 Sum N_3 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=7200 $y=-
5700 $w=400 $h=600
MPMOS_1 Sum N_3 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=7200 $y=-
4600 $w=400 $h=600
Vv5 Vdd Gnd  DC 2.5 $ $x=-10100 $y=-800 $w=400 $h=600
VV1 A Gnd  BIT({00001111} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-400 $y=0 $w=400 $h=600
VV2 B Gnd  BIT({00110011} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-400 $y=-1900 $w=400
$h=600
VV3 C Gnd  BIT({01010101} PW=10n ON=2.5 RT=1n FT=1n LT=10n HT=10n) $ $x=-400 $y=-4000 $w=400
$h=600
.PRINT   V(A) $ $x=-1150 $y=650 $w=1500 $h=300 $r=180
.PRINT   V(B) $ $x=-1150 $y=-1150 $w=1500 $h=300 $r=180
.PRINT   V(C) $ $x=-1150 $y=-3150 $w=1500 $h=300 $r=180
.PRINT   V(Carry) $ $x=6050 $y=-1650 $w=1500 $h=300
.PRINT   V(Sum) $ $x=8950 $y=-5250 $w=1500 $h=300
********* Simulation Settings - Analysis Section *********
.tran 0.1n 80n
********* Simulation Settings - Additional SPICE Commands *********

.end
```

CHAPTER :4

4.1ANALYSIS OF MINNICK COUNTER

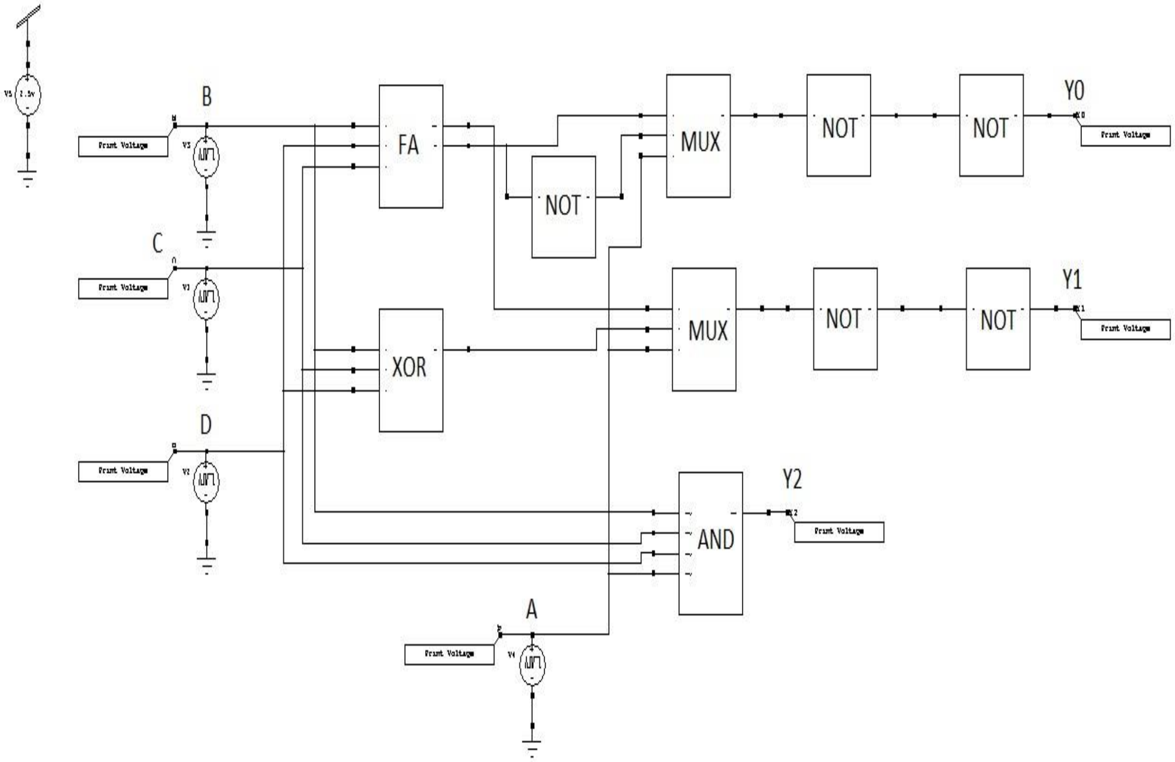

Fig.4.1: Minnick counter circuit diagram

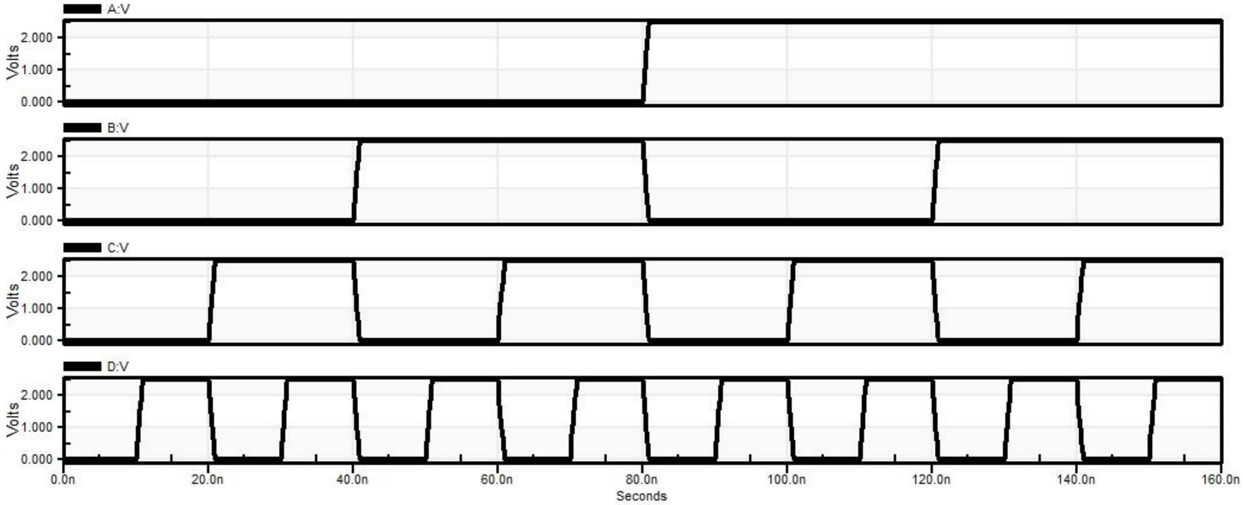

Fig.4.2**:** Minnick counter input waveform using CVDTL

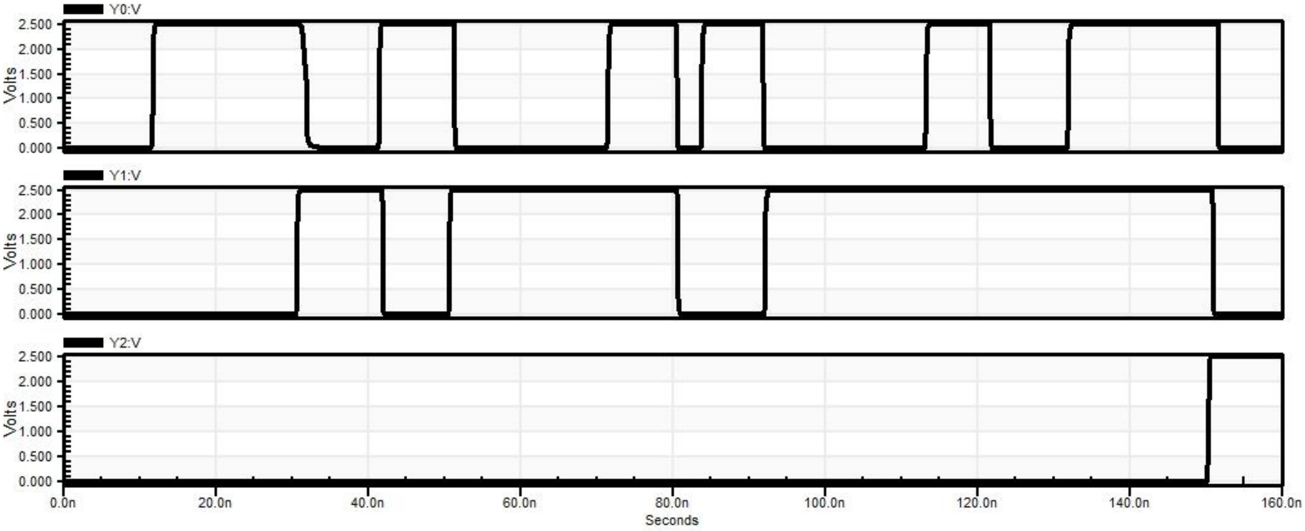

Fig. 4.3**:** Minnick counter output waveform using CVDTL

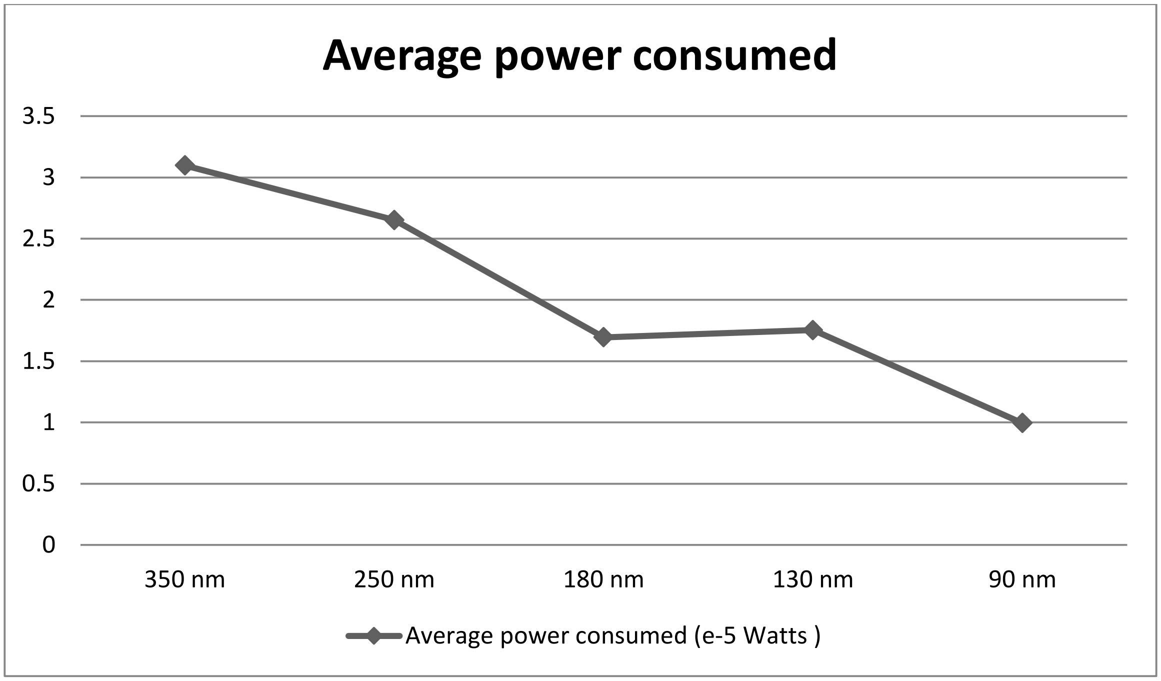

Fig. 4.4**:** Average power consumed by Minnick counter using CVDTL in different NM

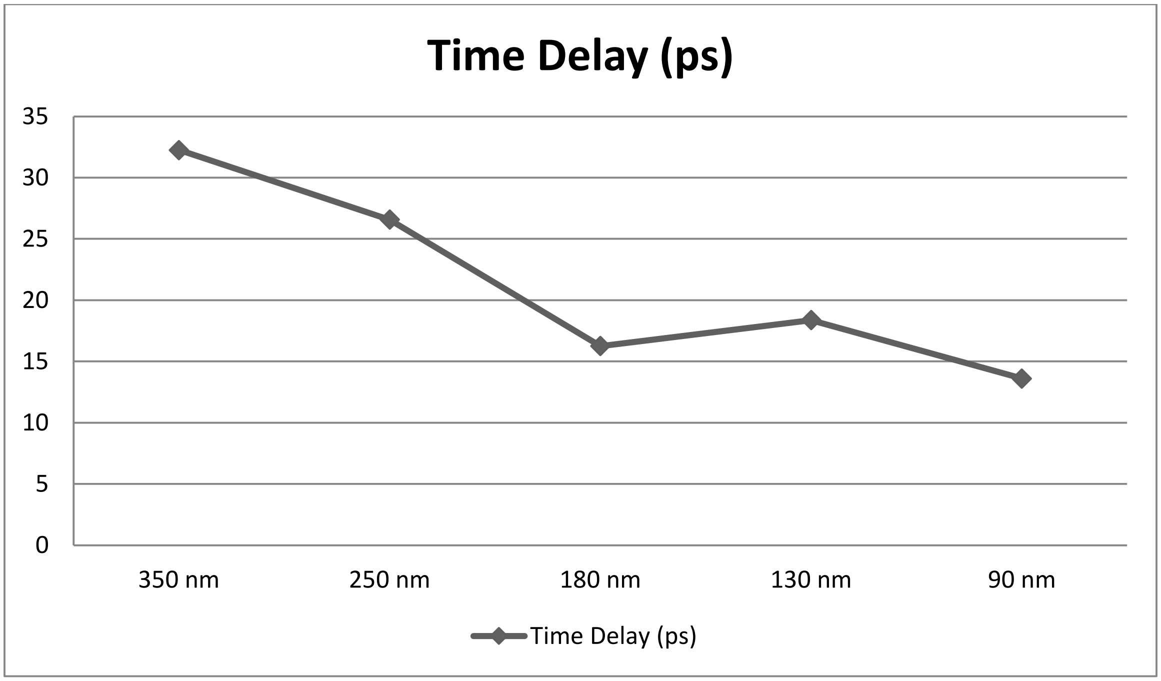

Fig. 4.5**:** Time delay in different NM

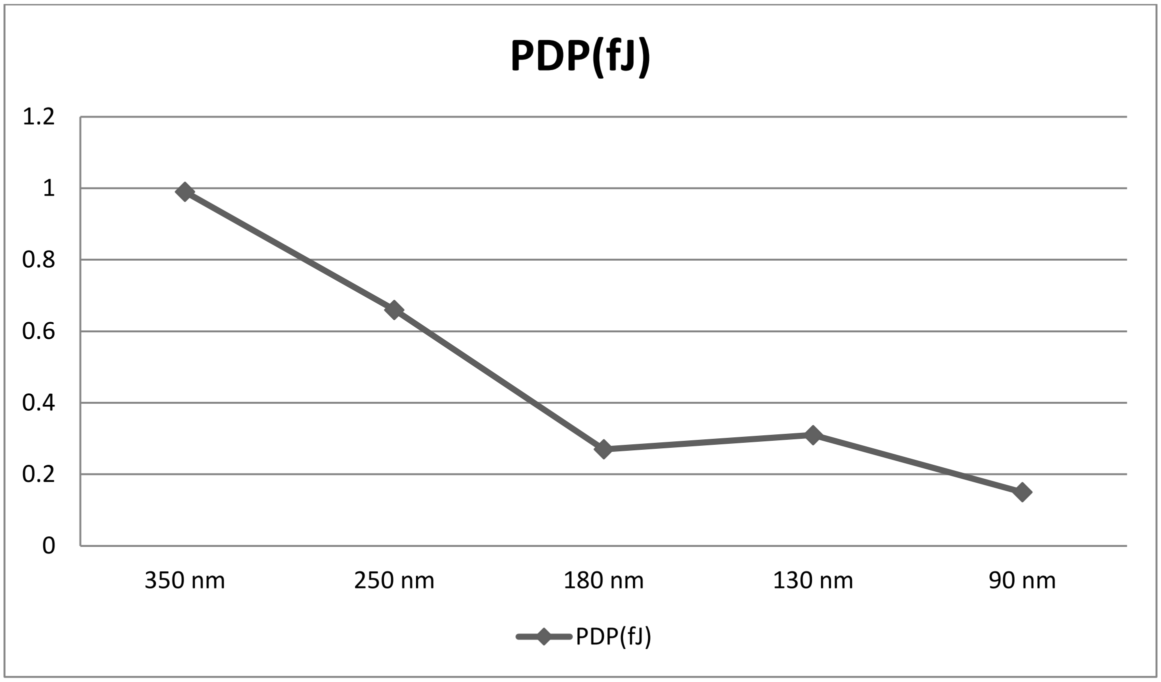

Fig. 4.6**:** Power Delay Product of Minnick counter in different NM

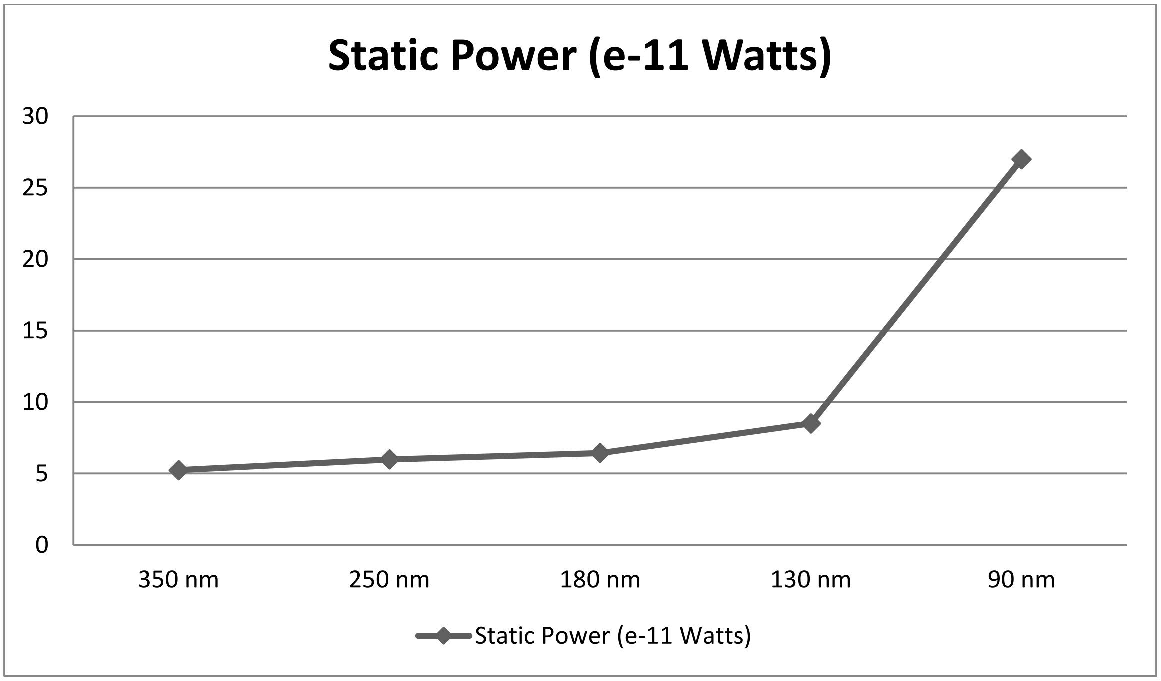

Fig. 4.7**:** Static Power of Minnick counter in different NM

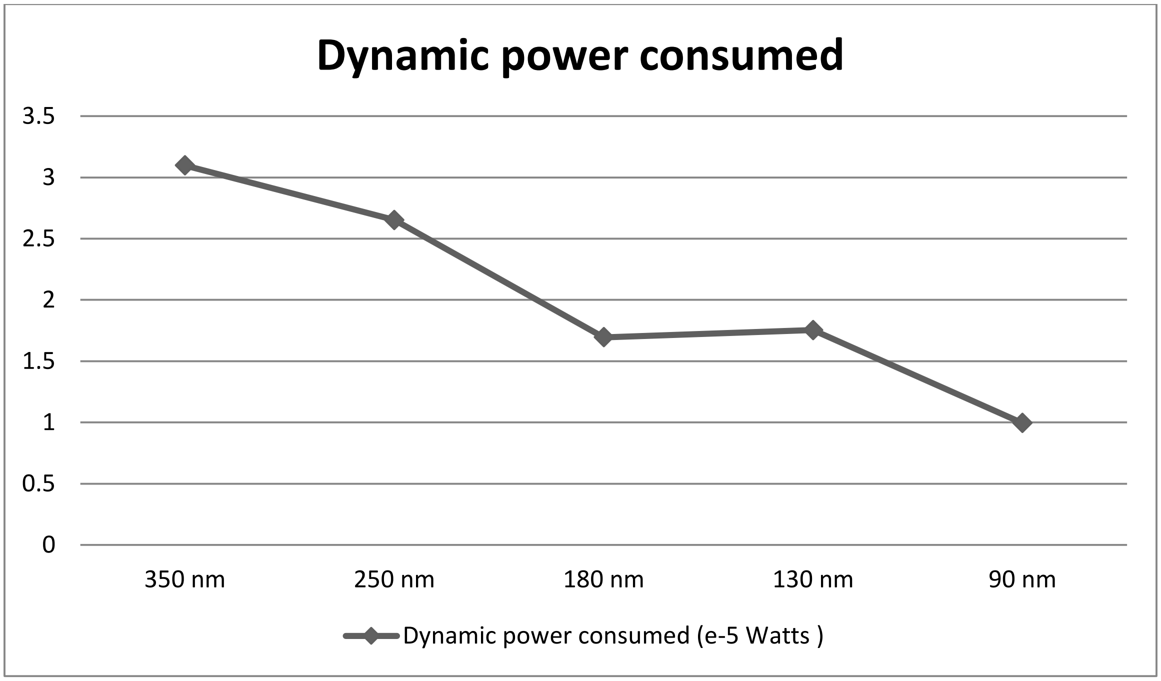

Fig. 4.8**:** Dynamic Power of Minnick counter using CVDTL in different NM

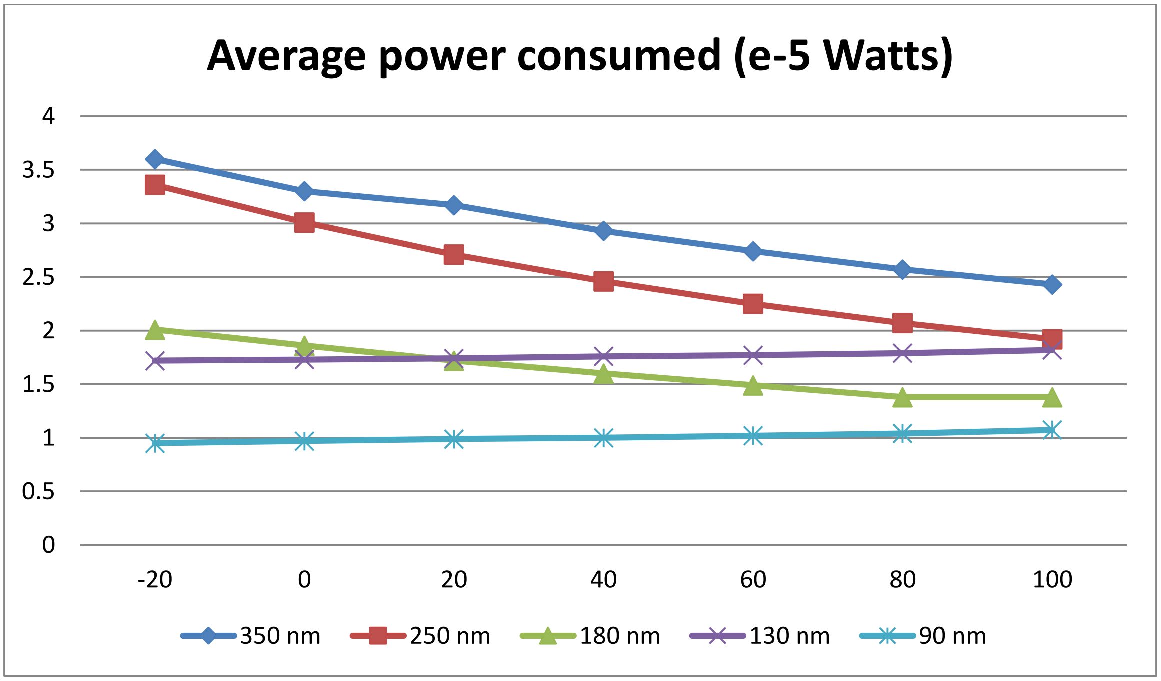

Fig. 4.9**:** Average power consumed by of Minnick counter using CVDTL in different NM and in different Temperature

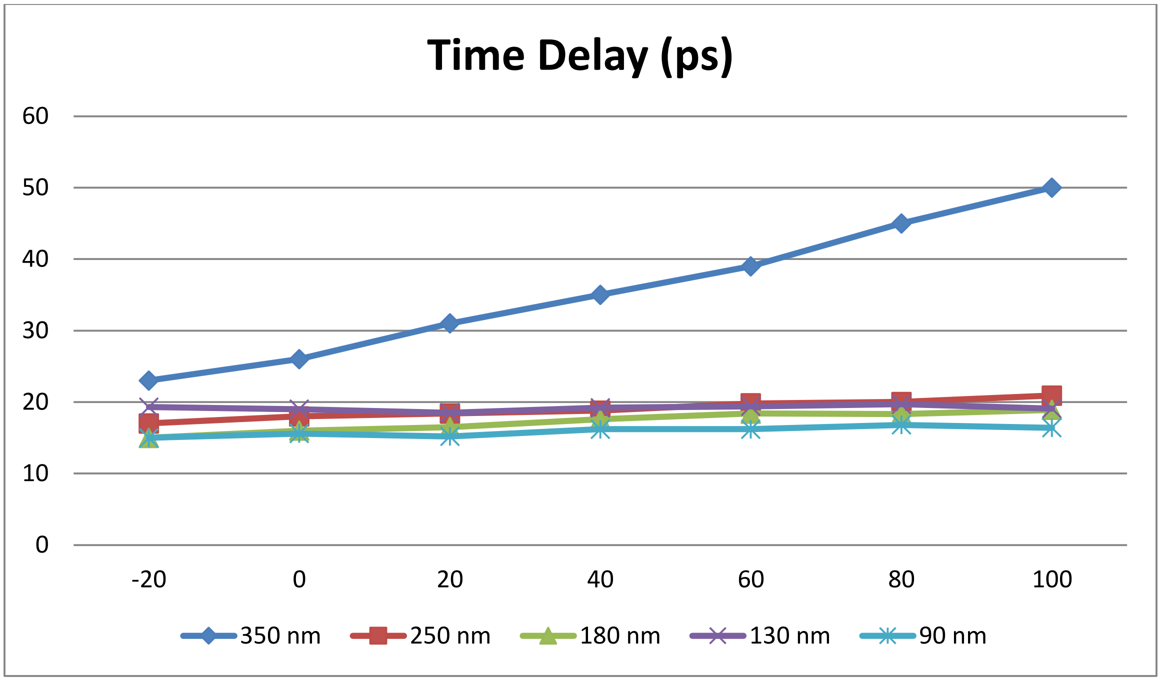

Fig. 4.10**:** Time Delay of Minnick counter using CVDTL in different NM and in different Temperature

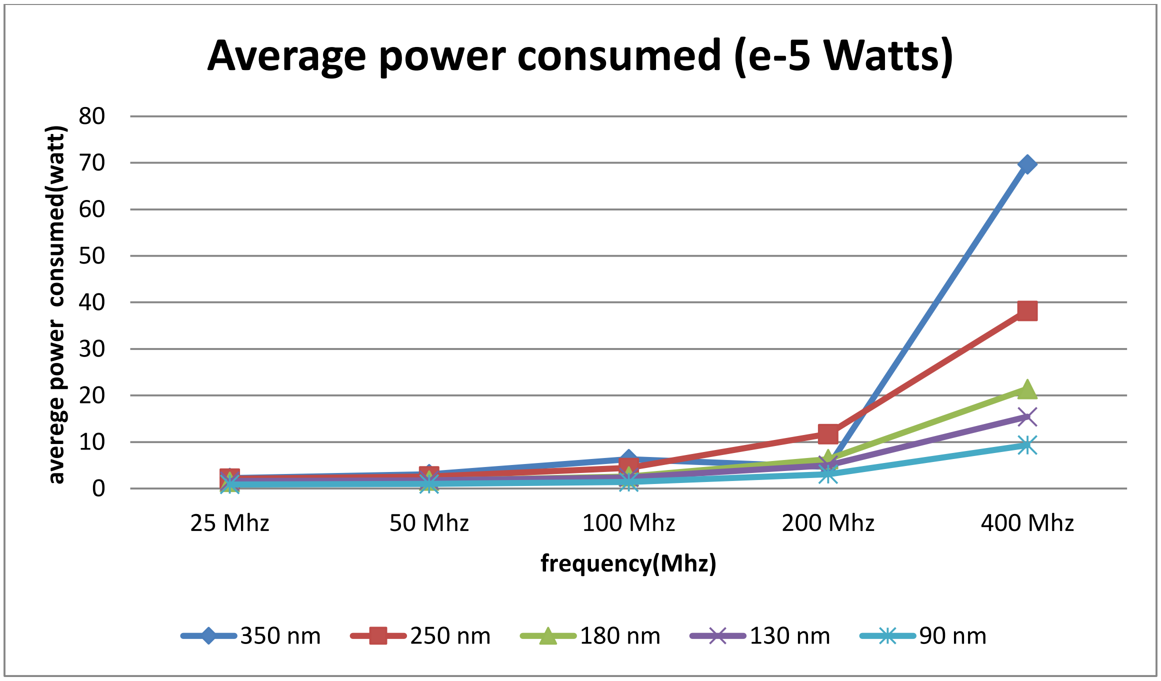

Fig. 4.11**:** Average power consumed by of Minnick counter using CVDTL in different NM and in different Frequency

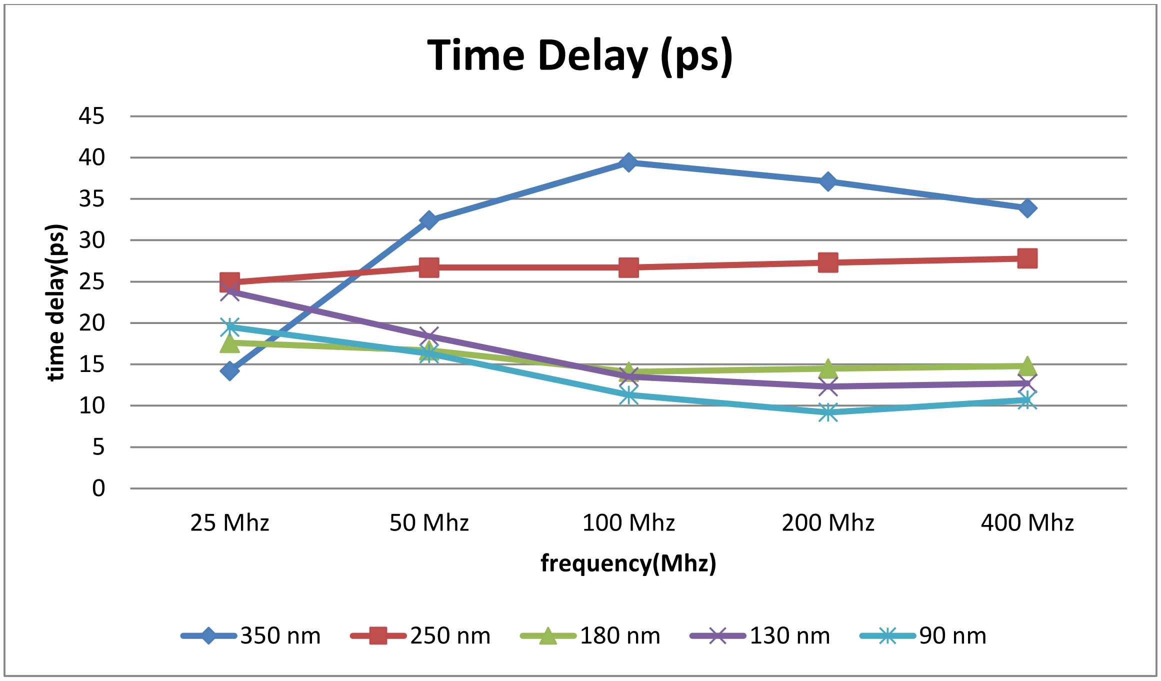

Fig. 4.12**:** Time Delay of Minnick counter using CVDTL in different NM and in different Frequency

T-Spice Code:

```
********* Simulation Settings - General Section *********
.probe
.option probev
.option probei
.option probeq
.include "C:\Users\Supratik\Documents\Tanner EDA\Tanner Tools v16.0\New Model Files\Nagendra Krishnapura\TSMC 250nm CMOS.md"

*************** Subcircuits *****************
.subckt Cell8 Input_1 Input_2 Input_3 Input_4 Output Gnd Vdd
CC1 Input_1 N_1  1p $ $x=2200 $y=1700 $w=600 $h=400
CC2 Input_2 N_1  1p $ $x=2200 $y=600 $w=600 $h=400
CC3 Input_3 N_1  1p $ $x=2200 $y=-800 $w=600 $h=400
CC4 Input_4 N_1  1p $ $x=2200 $y=-2200 $w=600 $h=400
MNMOS_1 N_2 N_1 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3800 $y=700 $w=400 $h=600
MNMOS_2 Output N_2 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=6200 $y=700 $w=400 $h=600
MPMOS_1 N_2 N_1 Vdd Vdd PMOS W=10u L=240n AS=9p PS=21.8u AD=9p PD=21.8u $ $x=3800 $y=1800 $w=400 $h=600
```

```
MPMOS_2 Output N_2 Vdd Vdd PMOS W=3u L=240n AS=2.7p PS=7.8u AD=2.7p PD=7.8u $ $x=6200 $y=1800
$w=400 $h=600
.ends

.subckt Cell9 A B C Carry Sum Gnd Vdd
CC1 A N_3  1p $ $x=3000 $y=7100 $w=600 $h=400
CC2 B N_3  1p $ $x=3000 $y=6000 $w=600 $h=400
CC3 C N_3  1p $ $x=3000 $y=4600 $w=600 $h=400
CC4 B N_5  1p $ $x=2300 $y=2900 $w=600 $h=400
CC5 C N_5  1p $ $x=2300 $y=1800 $w=600 $h=400
MNMOS_1 N_4 N_3 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=4600 $y=6100
$w=400 $h=600
MNMOS_2 Carry N_4 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=6100
$y=6100 $w=400 $h=600
MNMOS_3 N_9 B N_1 0 NMOS W=360n L=250n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=7200 $y=2200
$w=400 $h=600
MNMOS_4 N_1 C Gnd 0 NMOS W=360n L=250n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=7200 $y=1300
$w=400 $h=600
MNMOS_5 N_6 N_5 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3900 $y=1900
$w=400 $h=600
MNMOS_6 N_9 N_6 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5400 $y=1900
$w=400 $h=600
MNMOS_7 Sum N_9 N_2 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=8700 $y=1900
$w=400 $h=600
MNMOS_8 N_2 A Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=8700 $y=1000
$w=400 $h=600
MNMOS_9 N_8 A Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=6800 $y=9000
$w=400 $h=600
MPMOS_1 N_4 N_3 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=4600
$y=7200 $w=400 $h=600
MPMOS_2 Carry N_4 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=6100
$y=7200 $w=400 $h=600
MPMOS_3 N_6 N_5 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3900
$y=3000 $w=400 $h=600
MPMOS_4 N_9 N_6 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5400
$y=3000 $w=400 $h=600
MPMOS_5 Sum N_9 N_7 Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=8700
$y=3000 $w=400 $h=600
MPMOS_6 N_7 N_8 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=8700
$y=4600 $w=400 $h=600
MPMOS_7 N_8 A Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=6800
$y=10100 $w=400 $h=600
MPMOS_8 Sum A N_9 Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=10300
$y=2800 $w=400 $h=600
.ends

.subckt Cell10 A B Out S Gnd Vdd
MNMOS_1 A N_2 N_4 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1400 $y=-900
$w=600 $h=400 $r=270
MNMOS_2 B N_1 N_4 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1400 $y=-3200
$w=600 $h=400 $r=270
MNMOS_3 N_2 S Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=-3300 $y=-1900
$w=400 $h=600
MNMOS_4 Out N_3 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=8500 $y=-1900
$w=400 $h=600
```

```
MNMOS_11 N_3 N_4 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5200 $y=-
1900 $w=400 $h=600
MPMOS_1 A N_1 N_4 Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1400 $y=300
$w=600 $h=400 $r=90
MPMOS_2 B N_2 N_4 Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1400 $y=-
1700 $w=600 $h=400 $r=90
MPMOS_3 N_2 S Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=-3300 $y=-
1000 $w=400 $h=600
MPMOS_4 Out N_3 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=8500 $y=-
700 $w=400 $h=600
MPMOS_5 N_3 N_4 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=5200 $y=-
700 $w=400 $h=600
.ends

.subckt Cell12 A B C Out Gnd Vdd
CC1 A N_3  1p $ $x=-100 $y=3200 $w=600 $h=400
CC2 B N_3  1p $ $x=-100 $y=2100 $w=600 $h=400
CC3 C N_3  1p $ $x=-100 $y=600 $w=600 $h=400
MNMOS_1 N_4 N_3 Gnd 0 NMOS W=500n L=240n AS=450f PS=2.8u AD=450f PD=2.8u $ $x=1500 $y=2200
$w=400 $h=600
MNMOS_2 Out N_4 Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3000 $y=2200
$w=400 $h=600
MNMOS_3 Out A N_1 0 NMOS W=2.5u L=250n AS=2.25p PS=6.8u AD=2.25p PD=6.8u $ $x=4500 $y=2500
$w=400 $h=600
MNMOS_4 N_1 B N_2 0 NMOS W=2.5u L=250n AS=2.25p PS=6.8u AD=2.25p PD=6.8u $ $x=4500 $y=1600
$w=400 $h=600
MNMOS_5 N_2 C Gnd 0 NMOS W=2.5u L=250n AS=2.25p PS=6.8u AD=2.25p PD=6.8u $ $x=4500 $y=700
$w=400 $h=600
MPMOS_1 N_4 N_3 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=1500
$y=3300 $w=400 $h=600
MPMOS_2 Out N_4 Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3000
$y=3300 $w=400 $h=600
.ends

.subckt Cell13 In Out Gnd Vdd
MNMOS_1 Out In Gnd 0 NMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3200 $y=600
$w=400 $h=600
MPMOS_1 Out In Vdd Vdd PMOS W=360n L=240n AS=324f PS=2.52u AD=324f PD=2.52u $ $x=3200 $y=1700
$w=400 $h=600
.ends


***** Top Level *****
XCell8_1 B C D A Y2 Gnd Vdd Cell8 $ $x=5300 $y=-3700 $w=1800 $h=1400
XCell9_1 B D C N_1 N_2 Gnd Vdd Cell9 $ $x=600 $y=200 $w=1800 $h=1200
XCell10_1 N_2 N_4 N_5 A Gnd Vdd Cell10 $ $x=5100 $y=300 $w=1800 $h=1200
XCell10_2 N_1 N_3 N_6 A Gnd Vdd Cell10 $ $x=5200 $y=-1600 $w=1800 $h=1200
XCell12_1 B C D N_3 Gnd Vdd Cell12 $ $x=600 $y=-2000 $w=1800 $h=1200
XCell13_1 N_2 N_4 Gnd Vdd Cell13 $ $x=3000 $y=-400 $w=1800 $h=1000
XCell13_2 N_5 N_7 Gnd Vdd Cell13 $ $x=7300 $y=400 $w=1800 $h=1000
XCell13_3 N_6 N_8 Gnd Vdd Cell13 $ $x=7400 $y=-1500 $w=1800 $h=1000
XCell13_4 N_7 Y0 Gnd Vdd Cell13 $ $x=9700 $y=400 $w=1800 $h=1000
XCell13_5 N_8 Y1 Gnd Vdd Cell13 $ $x=9800 $y=-1500 $w=1800 $h=1000
VV5 Vdd Gnd  DC 2.5 $ $x=-7800 $y=0 $w=400 $h=600
VV1 C Gnd  BIT({0011001100110011} ON=2.5 ) $ $x=-2600 $y=-1300 $w=400 $h=600
VV2 D Gnd  BIT({0101010101010101} ON=2.5 ) $ $x=-2600 $y=-3100 $w=400 $h=600
```

```
VV3 B Gnd  BIT({0000111100001111} ON=2.5 ) $ $x=-2600 $y=100 $w=400 $h=600
VV4 A Gnd  BIT({0000000011111111} ON=2.5 ) $ $x=2500 $y=-4900 $w=400 $h=600
.PRINT   V(A) $ $x=1250 $y=-4750 $w=1500 $h=300 $m
.PRINT   V(B) $ $x=-3850 $y=250 $w=1500 $h=300 $m
.PRINT   V(C) $ $x=-3850 $y=-1150 $w=1500 $h=300 $m
.PRINT   V(D) $ $x=-3850 $y=-2950 $w=1500 $h=300 $m
.PRINT   V(Y0) $ $x=11750 $y=350 $w=1500 $h=300
.PRINT   V(Y1) $ $x=11750 $y=-1550 $w=1500 $h=300
.PRINT   V(Y2) $ $x=7250 $y=-3550 $w=1500 $h=300

********* Simulation Settings - Analysis Section *********
.tran 0.1n 160n

********* Simulation Settings - Additional SPICE Commands *********

.end
```

CHAPTER: 5

5.1 CONCLUSION

So far I have done 3 main modules in my project. They are capacitive voltage divider threshold logic, CMOS output wired logic. The capacitor in the capacitive voltage divider threshold logic is used because I can fabricate same value of capacitor within a very small area. The only substitution to this capacitor approach is to use a resistor. But for high value resistor I will be needing large area and that is the most disadvantage of using resistor. On the other hand the capacitor provides high speed at an expense of a little higher power consumption.

Now let's talk about CMOS output wired logic. This is superior to capacitor voltage divider threshold logic because its uses nonlinear device like CMOS and the PMOS and NMOS inside hence it acts as high resistance and low resistance interchangeably with time. As a result it operates at higher speed with lower power consumption than capacitive voltage divider threshold logic. The main advantage of this circuit is it has no loading effect because the input is totally isolated from the output side of the circuit.

If we want to design a Minnick Counter in a traditional pattern then no of MOS is increased. In the new design pattern MOS count quit low in number to improve the efficiency of its function. The newly design Minnick Counter has lots of advantages. The first one is it takes less occupied space. So power consumption is very low in the newly design Minnick Counter and the third advantage which it offers cost efficiency or low cost production. The newly designed Minnick Counter makes no delay.

5.2 FUTURE SCOPE:

Minnick Counter seamlessly performs in many ways. It can be utilized for error detection. On the other hand it can be used to compress data.

The most unique function it offers binary multiplications. It can easily perform well in the realm of big data multiplication between large n bit.

The Minnick Counter has many unique features which will enrich the future of VLSI.

REFERENCE

[1] A Threshold Logic Full Adder Based on Resonant Tunneling Transistors, published at: ESSCIRC'98

[2] R. Shalem, E. John, and L. K. John, "A novel low-power energy recovery full adder cell," in *Proc. Great Lakes Symp. VLSI*, Feb. 1999, pp. 380–383

[3] H. T. Bui, Y. Wang, and Y. Jiang, *"Design and Analysis of 10- Transistor Full Adders Using XOR-XNOR Gates,"* IEEE Trans. Circuits and Syst. II, Analog Digit. Signal Process., vol 49, no. 1, pp. 25-30, Jan. 2002.

[4] K. Navi, M. Maeen, V. Foroutan, S. Timarchi, and O. Kavei, *"A Novel Low Power Full-Adder Cell for Low Voltage,"* Integration the VLSI Journal, 2009

[5] S. Veeramachaneni, M. B. Sirinivas, *"New Improved 1-Bit Full Adder Cells",*CCECE/CGEI, Canada, 2008

[6] P.K. Sinha Roy,'' Test & realization of linearly separable switching functions'', Int. J. Control, 1970, Vol.11, No.5, 873-89

[7] T. Shibata and T. Ohmi, "An intelligent MOS transistor featuring gate-level weighted sum and threshold operations," in *IEDM, Technical Digest*, New York, NY, USA, Dec 1991, IEEE.

[8] H. O¨ zdemir, A. Kepkep, B. Pamir, Y. Leblebici, and U. C¸ iliniro˘glu, "A capacitive threshold-logic gate," *IEEE JSSC*, vol. 31, no. 8, pp. 1141–1149, August 1996.

[9] P. Celinski, J. F. L´opez, S. Al-Sarawi, and D. Abbott, "Low power, high speed, charge recycling CMOS threshold logic gate," *IEE Electronics Letters*, vol. 37, no. 17, pp. 1067–1069, August 2001.

[10] Mili Sarkar Sunit Das Upasana Roy Chowdhury Dibyajyoti Das " Design of Sequential Circuits using Single Electron Encoded Logic"IEMCON 2012.

[11] Mili Sarkar, Shilpi Raj,Prasenjit Sengupta "Design of Sequential circuits using Threshold Logic"S. Muroga, *Threshold Logic and Its Applications*. New York: Wiley,197

[12]'Bit-sliced median filter design based on majority gate'-C.L.Lee, C.W.Jen Dec. 1995. K. Elissa, "Title of paper if known," unpublished.

[13] VLSI Implementation of Threshold Logic-A Comprehensive Survey by Valeriu Beiu,Jose M Quintana,Maria J Avedillo,IEEE Transaction on Neural Networks,Vol.14,No.5 September 2003

[14] β-Driven Threshold Elements by Victor I Varshavsky

[15] Shibata T., Ohmi T. \Neuron MOS binary-logic integrated circuits: Part 1, Design fundamentals and soft-hardware logic circuit implemen-tation." IEEE Trans. Electron Devices, Vol.40,No.5, pp.974{979 (1993).

[16] Ohmi T., Shibata T., Kotani K. Four-Terminal Device Concept for Intelligence Soft Computing on Silicon Integrated Circuits." Proc. ofIIZUKA'96, pp.49{58.

[17] Bellabes N. et al. \Ratioed voter circuit for testing and in VLSI processing arrays." Proc. ISCAS'92(1992).

[18] Lee Ch.L., Jeh Ch.-W. \CMOS threshold gates and networks for order statistic filtering." IEEE Trans. on Circuits and Systems, Vol.41, No.6,pp.453{456 (1994).

[19] Bellabes N. et al. \Ratioed voter circuit for testing and in VLSI processing arrays." IEEE Trans. on Circuits and Systems, Vol.43, No.2, pp.143{152(1996).

[20] Weste N., Eshraghian K. \Principles of CMOS VLSI Design." Addison-Wesley (1985).

[21] Ohmi T. \ULSI reliability through ultraclean processing." Proc. of the IEEE, Vol.81, No.5,pp.716{729 (1993).

[22] Varshavsky et al. \H-latch", USSR AuthorshipCertificate No.1562964 cl.H 03 K 3/3

[23] M.J. Avedillo, J.M. Quintana, A. Rueda, and E. Jim´enez, "Low-power CMOS threshold-logic gate," *IEE Electronics Letters*, vol. 31, no. 25, pp. 2157–2159,

[24] J. Fernandez Ramos, J. A. Hidalgo Lopez, M. J. Martin, J. C. Tejero, and A. Gago, "A threshold logic gate based on clocked coupled inverters," *International Journal of Electronics*, vol. 84, no. 4, pp. 371–382, 2001.

[25] Y Taur , D.A. Buchanan, W. Chen , D. Frank, K. Ismail, H. Wann,S. Wind, and H. Wong. CMOS Scaling into the Nanometre Regime. Proceeding of the IEEE, Vol. 85(No.4):pp.486-504, 1997. 106
[26] IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 14, NO. 5, SEPTEMBER 2003 "VLSI Implementations of Threshold Logic—A Comprehensive Survey"
[27] Hazard-free edge-triggered D flipflop based on threshold gates,J.M.Quintana ,M.J. Avedillo and A Rueda
[28] 'Design of Combinational and Sequential Circuits using Threshold Logic'Mili Sarkar,Subhadeep Nag
[29] 'A Capacitive Threshold Logic Gate' by Hakan Ozdemir,Yusuf Leblebici,Asin Kepkep,Banu Pamir
[30] 'Compact and Power Efficient MOS-NDR Muller C Elements' by Juan Nunez,Maria J Avedillo,and, Jose M.Quintana
[31] 'Fast and Area Efficient Multi-Input Muller C Element based on MOS NDR' by Juan Nunez,Maria J Avedillo
[32] S. McCulloch and W. Pitts, "A Logical Calculus of the Ideas Imminent in NervousActivity", *Bulletin of Mathematical Biophysics*, 5, 1943, pp.115-133.
[33] C. Mead, *Analog VLSI and Neural Systems*.Addison-Wesley, 1989.
[34] T. Shibata, T. Ohmi, "Neuron MOS BinarylogicIntegrated Circuits: Part 1, DesignFundamentals and Soft-hardwareLogic CircuitImplementation", *IEEE Trans. Electron Devices*,Vol.40, No.5, 1993, pp. 974-979.
[35]T. Ohmi, T. Shibata, K. Kotani, "Four-Terminal Device Concept for Intelligence SoftComputing on Silicon Integrated Circuits".*Proc. of IIZUKA'96*, 1996, pp. 49-59.
[36] S.M. Fakhraie, K.C. Smith, *VLSI-CompatibleImplementations for Artificial Neural Networks*.Kluwer, Boston-Dordrecht-London, 1997.
[37 Montalvo, R. Gyurcsik and J. Paulos,"Toward a General-Purpose Analog VLSI Neural Network with On-Chip Learning", *IEEE Transactions on Neural Networks*, Vol.8, No.2,
March 1997, pp.413-423.
[38] V. Varshavsky, "Beta-Driven Threshold Elements", *Proceedings of the 8-th Great Lakes Symposium on VLSI*, IEEE Computer Society, Feb. 19-21, 1998, pp.52-58.
[39] V. Varshavsky, "Threshold Element and a Design Method for Elements", filed to Japan's Patent Office, Jan.30, 1998, the application number is JPA H10-54079.
[40] V. Varshavsky, "Simple CMOS Learnable Threshold Element", *International ICSC/IFAC Symposium on Neural Computation*, Vienna,Austria, Sept.23-25, 1998.
[41 V. Varshavsky, "CMOS Artificial Neuron on the Base of Beta-Driven Threshold Element",*IEEE International Conference on Systems, Man and Cybernetics*, San Diego, CA, October 11-14, 1998, pp.1857-1861.
[42] V. Varshavsky, "Synapse, Threshold Circuit and Neuron Circuit", filed to Japan's Patent Office on Aug. 7,1998, the application number is JPA-H10-224994.
[43] V. Varshavsky, "Threshold Element", filed to Japan's Patent Office on Aug. 12, 1998, the application number is JPA-H10-228398.
[44] V. Varshavsky and V. Marakhovsky, "Beta-CMOS implementation of artificial neuron",SPIE's 13th Annual International Symposium on Aerospace/Defense Sensing, Simulation, and Controls. Applications and Science ofComputational Intelligence II, Orlando, Florida, April 5-8, 1999, pp.210-221.
[45] V. Varshavsky and V. Marakhovsky, "Beta-CMOS Artificial Neuron and Implementability Limits", Lecture Notes in Computer Science 1607, Engineering Applications of Bio-Inspired Artificial Neural Networks.
[46] Jose Mira, Juan V. Sanchez-Andves(Eds.).Proceedings of International Work-Conference on Artificial & Natural Neural Networks (IWANN'99)}, Spain, June 2-4, Springer,Vol.11, 1999, pp. 117-128.

[47] V. Varshavsky and V. Marakhovsky, “The Simple Neuron CMOS Implementation Learnable to Logical Threshold Functions”,*Proceedings of International Workshop on SoftComputing in Industry'99 (IWSCI'99)*, June 16-18, Muroran,Hokkaido, Japan, IEEE, 1999, pp.463-468. 107
[48] V. Varshavsky and V. Marakhovsky, “Implementability Restrictions on the Beta-CMOS Artificial Neuron”,*Proceedings of the 6th International Conference on Electronics,Circuits and Systems (ICECS'99)*, September 5-8, 1999, Pafos, Cyprus, IEEE, 1999, pp. 401-405.
[49] V. Varshavsky and V. Marakhovsky,“Learning Experiments with CMOS ArtificialNeuron”, *Lecture Notes in Computer Science 1625, Computational Intelligence Theory and Applications, ed. by Bernard Reusch.Proceedings of the 6th Fuzzy Days International Conference on Computational Intelligence*,Dortmund, Germany, May 25-27, Springer,1999, pp. 706-707.
[50] A. Montalvo, R. Gyurcsik, and J. Paulos,“Toward a General-Purpose Analog VLSI Neural Network with On-chip Learning”, *IEEE Transactions on Neural Networks*, Vol.8, No.2, March 1997, pp. 413-423.

……………………………………..