A 0.449psec RMS Jitter All Digital Phase-Locked Loop

By

Ayman Nabil Mohamed

Basma Mourad Mohamed

Ehab Mahmoud Helmy

Hany Mohamed Amin

Ingy Abdelhamid Mohamed

Under the Supervision of

Dr. Hassan Mostafa

A Graduation Project Report Submitted to
the Faculty of Engineering at Cairo University
in Partial Fulfillment of the Requirements for the
Degree of
Bachelor of Science
in
Electronics and Communications Engineering

Faculty of Engineering, Cairo University

Giza, Egypt

July 2014

Table of Contents

List of Tables

List of Figures

List of Symbols and Abbreviations

AAPLL	All Analog Phase Locked Loop
ADPLL	All Digital Phase Locked Loop
ADC	Analog to Digital Converter
BBPD	Bang-Bag Phase Detector
BBPFD	Bang-Bang Phase and Frequency Detector
CP	Charge Pump
DCD	Duty Cycle Distortion
DCO	Digital Controlled Oscillator
DCV	Digitally Controlled Varactor
DFF	D flip flop
DJ	Deterministic Jitter
DLF	Digital loop filter
DSM	Delta sigma modulator
DRC	Design Rule Check
FBAR	Film Bulk Acoustic wave Resonator
FIR	Finite Impulse Response
FOM	Figure Of Merit
HDL	Hard ware Description Language
HLS	High Level Synthesis
IIR	Infinite Impulse Response
ISI	Inter-Symbol Interference
LPF	Low Pass Filter
LVS	Layout Vs. Schematic
MUX	Multiplexer
PD	Phase Detector
PFD	Phase and Frequency Detector
PLL	Phase Locked Loop
PVT	Process, Voltage and Temperature
RMS	Root Mean Square
RJ	Random Jitter
RTL	Register Transfer Logic

SSPLL	Sub-Sampling Phase-Locked Loop
TDC	Time-to-Digital Converter
VCO	Voltage Controlled Oscillator
VLSI	Very Large Scale Integrated Circuit

Acknowledgments

We would like to express our deepest gratitude for Faculty of Engineering, Communication department Professors and teaching assistants for the continuous support for our 4 years of study. We would like to express our deepest appreciation for Dr. Hassan Mostafa for the guidance, patience, useful knowledge and the time he spent with us working on this graduation project. Special thanks go to our colleagues for all the support and the help they offered during our study period.

A thank you to our families for the environment they created for us in order to excel in our studies and to achieve excessive successes. Above all, utmost appreciation and gratitude to the Almighty Allah for giving us the strength and the ability to work and produce this project. We are dedicating this work and all our efforts to the free souls fighting for human rights and dignity and standing tall for what they believe, for our colleagues behind bars and for the souls that left us while defending their rights.

We are also dedicating this for the soul of our colleague and best friend Omar Ossama, who died a year earlier who would have been expected to graduate with us these days.

Abstract

Phase Locked Loop (PLL) has many applications such that frequency tracking, clock generation and frequency synthesis. Analog PLL has the advantages of a very good performance and high frequency range. The main challenge is the high power consumption, large area, and scalability.

All digital PLL (ADPLL) is a replacement for the Analog PLL that has advantages such as the low power consumption, small area and scalability across different technology nodes. Still the performance of the Analog PLL is superior to the ADPLL in terms of frequency range, jitter and lock time.

An all-digital phase locked loop (ADPLL) is presented to achieve both low phase noise and low power simultaneously. Targeted specifications to be reached: 2psec peak-to-peak jitter with RMS value of 0.5psec covering frequency range from 415MHz to 1GHz, 1v operating voltage and the total area is $0.1mm^2$.

The proposed ADPLL is fabricated in 65nm CMOS technology, its power consumption is 80mW for a supply voltage of 1V. The measured phase noise of the DCO is -140dBc/Hz at an offset frequency of 1MHz. The integrated root-mean-square jitter is 0.449ps and the peak-to-peak jitter is 2.7ps. Lock time is around 70ns.

This ADPLL uses a bangbang phase and frequency detector to maintain the phase locking without time-to-digital converter, and the dividers are disabled. The DCO is an LC oscillator which produces jitter lower than the regular ring oscillator.

Chapter 1: Introduction

1.1 What is PLL?

Phase locked loop (PLL) has been widely used for clock recovery, noise and jitter suppression in communications, clock synchronization in memory interface and high-performance microprocessors, and frequency synthesis for instrumentation and RF receivers. Our basic challenges are power consumption, area, and scalability.
Analog PLLs are widely used, and show high performance characteristics in terms of jitter and frequency range. However, the high power consumption and large area have always been disadvantages for this type of PLLs. [1]

The last decade has shown a lot of interest in replacing the analog PLLs with an All Digital PLL (ADPLL). The main advantages driving this new trend are the fast time to market, and the small design effort required to migrate between different technology nodes.

The simple diagram of PLL in the figure below, its function is to keep the reference frequency and the DCO frequency in phase by locking on it. The voltage controlled Oscillator is used to generate an output clock with frequency proportional to the input actuation voltage VA. This output clock is fed back through a frequency divider to get the divided frequency. A Phase Detector (PD) is used to compare the phase of the divided clock with the phase of the reference clock and generate a voltage proportional to the phase error. The loop filter is used to stabilize the system and achieve the desired response. This control loop makes the divided clock have the same frequency and phase as the reference clock. Hence, the output clock will have a frequency that is N times the reference clock. [2]

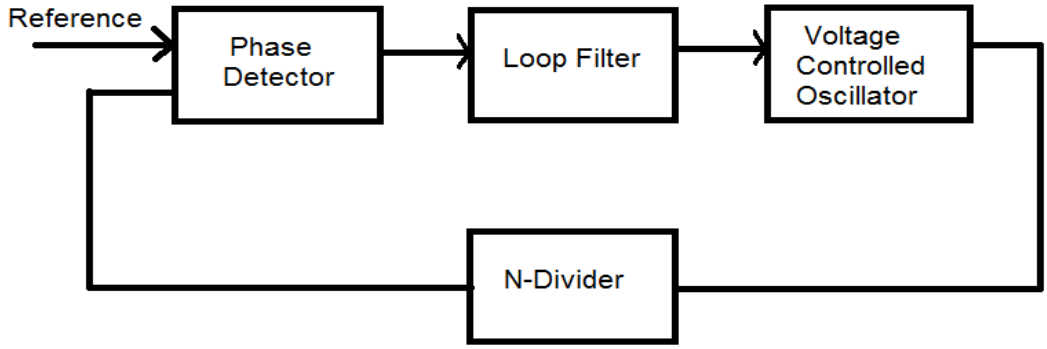


Figure 1: Simple Architecture of PLL

In this work, the target is to design an ADPLL that is suitable for an embedded system. The ADPLL is required to have extremely low jitter, small area and small lock time. The requirements on the power and frequency range are relaxed.

1.2 AAPLL vs. ADPLL

Advantages of ADPLL are low power, small area and scalability across different technology nodes also the disadvantages are frequency range, jitter and lock time, VCO in AAPLL is converted to DCO, which dominates the major performance measures of the ADPLL, such as power consumption and jitter.

Advantages of Analog PLLs are high performance characteristics in terms of jitter and frequency range and they are widely used. However, the high power consumption and large area have always been disadvantages for this type. In addition, when we go down into deep submicron, the design of these analog circuits becomes very sensitive and increases the design cycle time and time to market.

1.3 Issues faces PLL

From the problems faced us from using phase locked-loop are: Acquisition, Jitter and Frequency Range.

1.3.1 Acquisition

Acquisition is a nonlinear process. It is a phase acquisition since a PLL locks up with just a phase transient if the frequency difference is small and the loop can acquire lock without the cycle in PLLs. But if the reference frequency is initially far away from

the VCO free-running frequency, it is the frequency acquisition. A coarse and a fine bit by phase and frequency detector (PFD) and time-to-digital converter (TDC) when the frequency acquisition with no phase difference is done under the lock condition in PLLs. The accumulated bits and all zero bits continuously are injected into digitally controlled oscillator (DCO). The constant bits generates a constant frequency from DCO. [3]

1.3.2 Frequency Range

The reference frequency multiplied by N factor of the PLL determine the output frequencies of ADPLL, and N factor depends on the application's specification. The factors are categorized by the characteristics of ADPLLs. It may have mixed factors or programmable ones.

Since the ranges of output frequencies are dictated by the target application, the DCO produces the proper frequencies in ranges of the applications. The output frequencies should be considered with unpredictable effects on target applications such as temperature variation, supply voltage fluctuation.

1.3.3 Jitter

Jitter is defined as the short-term variations of a signal with respect to its ideal position in time. As shown in the figure, the original signal (unit interval) is expected to be in certain position (ideal edge location) but it appears shifted (edge location shifted), then the difference between the original signal and the shifted version is the jitter.

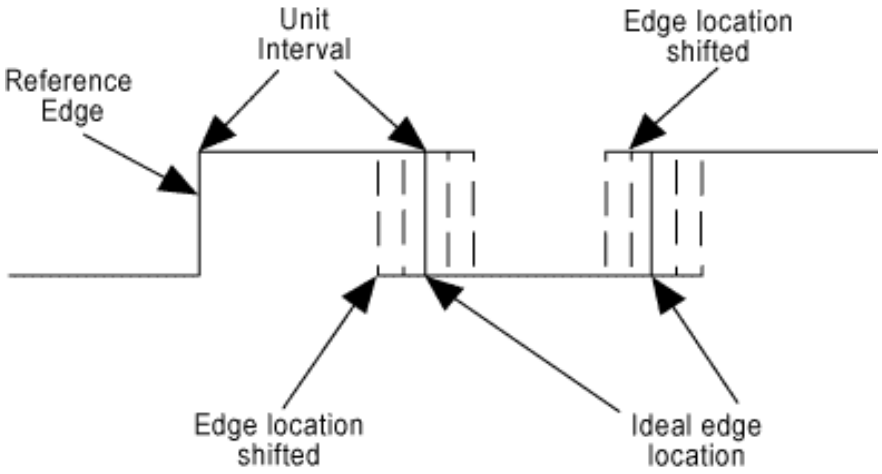


Figure 2: Jitter Effect

There are two kinds of jitter, Deterministic jitter (DJ) and Random jitter (RJ). The predictable component of jitter in the circuit is called deterministic jitter and the remaining components of jitter are called random jitter.

Deterministic Jitter: Jitter with non-Gaussian probability density function. It is always bounded in amplitude and has specific causes. Four kinds of deterministic jitter are identified: duty cycle distortion, data dependent, sinusoidal, and uncorrelated (to the data) bounded. DJ is characterized by its bounded, peak-to-peak value. [4]

To measure the deterministic jitter on a clock (or data) waveform you must trigger your oscilloscope at a rate commensurate with the source of the deterministic jitter. Deterministic jitter comes from many sources, including duty-cycle distortion (DCD) and intersymbol interference (ISI). [5]

Random Jitter: It is a Gaussian jitter that is unbounded which comes from thermal vibrations of semiconductor crystal structure which causes mobility to vary depending on instantaneous temperature of material, also from material boundaries have less than perfect valence electron mapping, and Imperfections due to semi regular doping density through semiconductor substrate, well and transistor elements. [4]

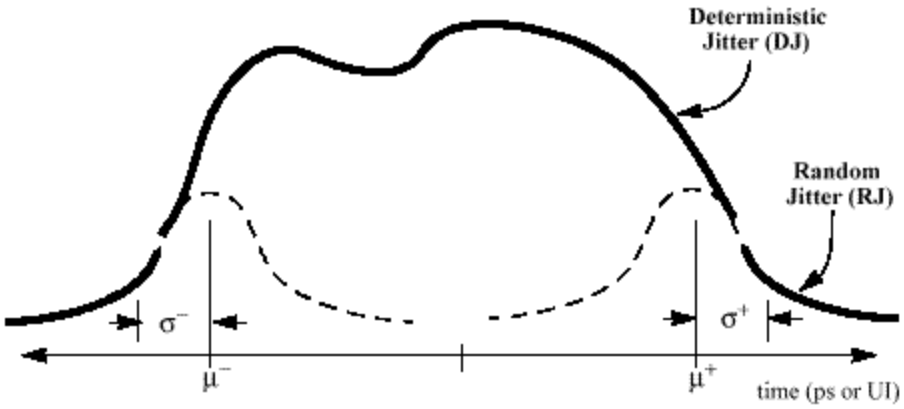


Figure 3: Deterministic and Random Jitter [4]

The main difference between random and deterministic jitter is the components, as the deterministic components have a lower ratio of peak value to standard deviation than do the random components. Generally, a certain amount of deterministic jitter does not hurt your system nearly as much as a similar quantity of random jitter.

After we have discussed the meaning of phase locked loop (PLL) and mentioned the difference between analog phase-locked loop and digital phase-locked loop, it was clear that analog is better from the jitter point of view, which is the core of our project. Hence, there are specifications which we target to reach in our work to say that digital phase-locked loop is better than one. These specifications are: first, to get peak-to-peak jitter lower than 2psec, second the root-mean-square of jitter to be lower than 0.5psec, third the frequency range must be between 400MHZ and 1GHZ, fourth the power consumption to be less than 10mW, fifth and finally to get an area less than $0.1mm^2$.

To achieve such specification we did searched and worked on different kind of papers as going to be shown in Chapter 3.

In addition, we are going to show basic block diagrams of PLL and their types in Chapter 2. In Chapter 4, blocks of the proposed architecture are discussed from its internal structure. The whole system is connected in Chapter 5 and the results are shown before layout and after it. Finally, we conclude our work in Chapter 6.

Chapter 2: Basic Blocks of PLL

In this chapter, we are going to discuss the main blocks of the PLL and its different structures.

2.1 Phase Detector (PD)

It has two input signals: the reference input signal and the feedback from the DCO.

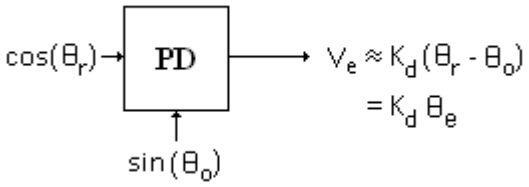
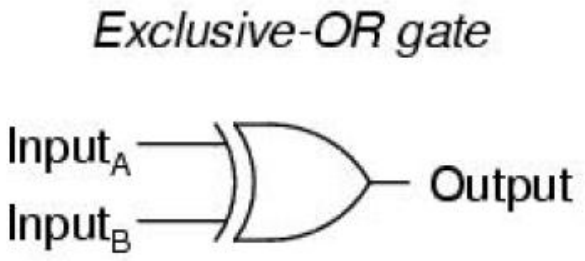


Figure 4: Phase Detector [6]

An ideal phase detector (PD) produces an output signal whose DC value is linearly proportional to the difference between the phases of the two periodic inputs. The output pulse will have a width which is equal to the time difference between consecutive zero crossings of the two inputs.
Phase detectors consist of many types in the two main categories of analog and digital environment, depending on the kind of application. Most of the analog phase detectors are just analog multipliers: their output voltage is proportional to the multiplication of the two input voltages. And their main advantage is that they can be used in a large frequency range. In the other hand the digital phase detectors consists of two categories of linear phase detectors and bangbang phase detectors.
These two categories are detailed now.

2.1.1 Linear Phase Detectors

The first implementation of this type of detectors is an XOR digital gate; whose inputs are the reference signal and the DCO output.



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

Figure 5: Left XOR gate, right Truth table [6]

The XOR gate is perfectly linear; its output voltage is exactly proportional to the phase error. However, it would have a problem, if the phase error is greater than 180^0 that the characteristic loses its linearity and the output voltage is not proportional to this error anymore as shown below. [6]

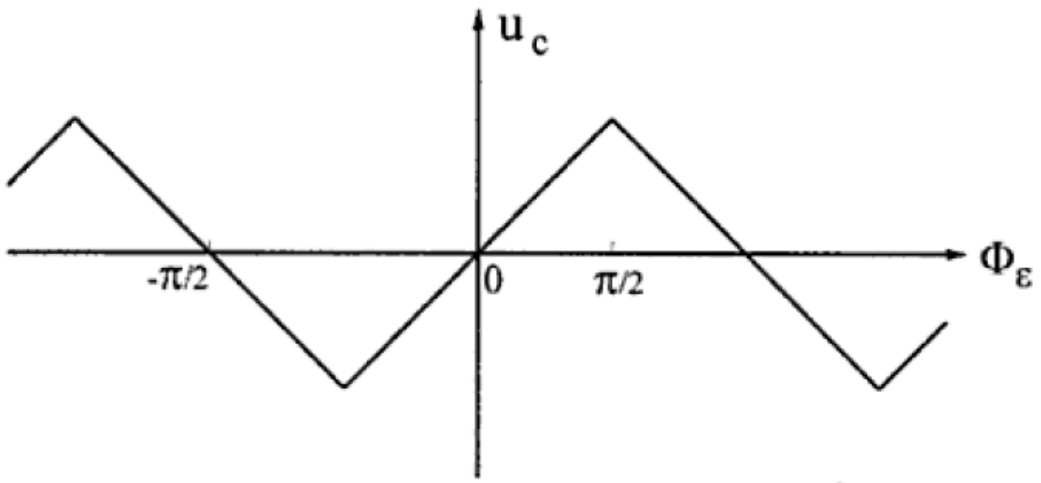


Figure 6: XOR phase detector characteristics [6]

The second type is phase and frequency detector (PFD), it is shown in figure 7 with its timing diagram.
It is linear and its output is always proportional to the phase difference between the two input signals and have another advantage that it is used also as a frequency detector. [6]

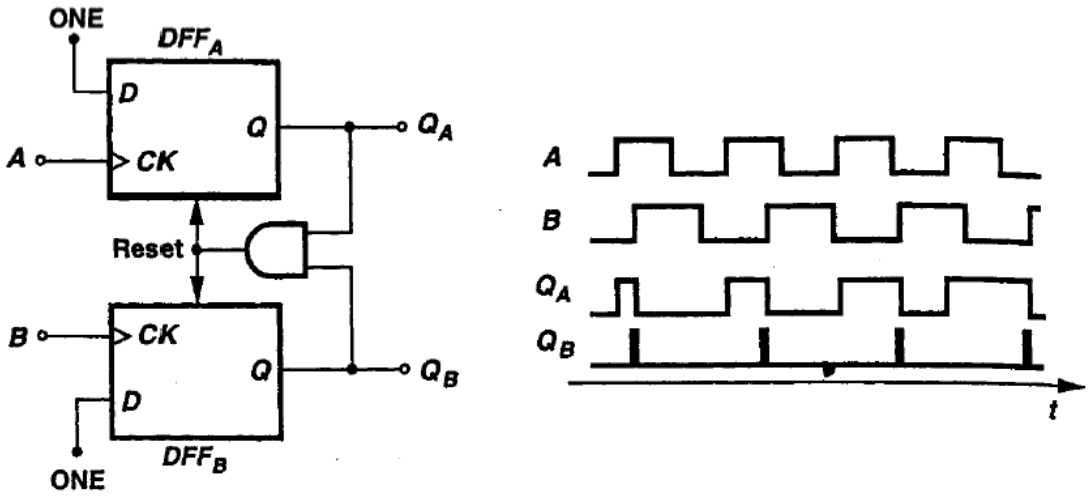


Figure 7: Left: PFD, Right: Timing diagram [6]

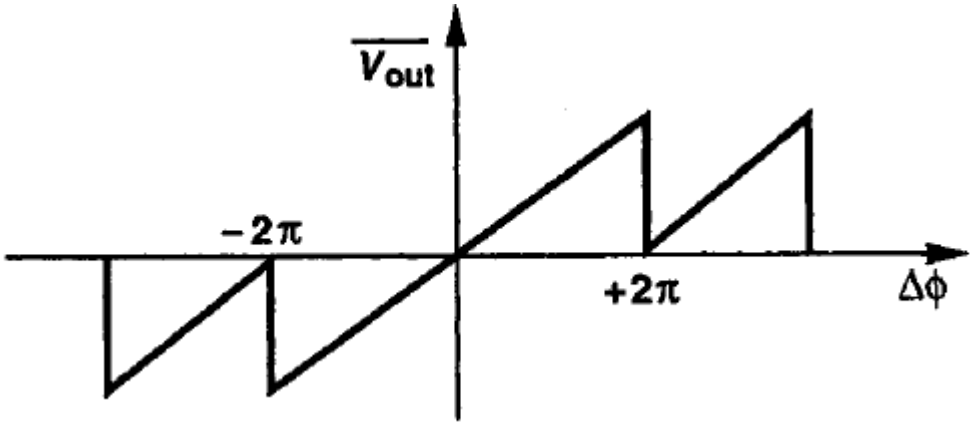


Figure 8: PFD characteristics [6]

Third type is Hogge phase detector, which consists of two D flip-flops and two XOR gates as below.

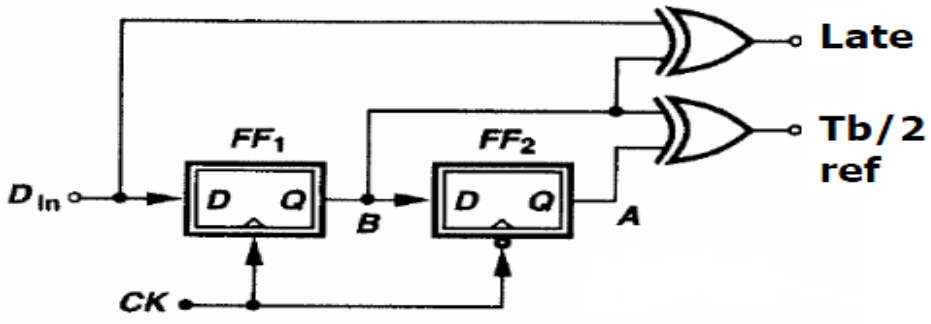


Figure 9: Hogge phase detector [7]

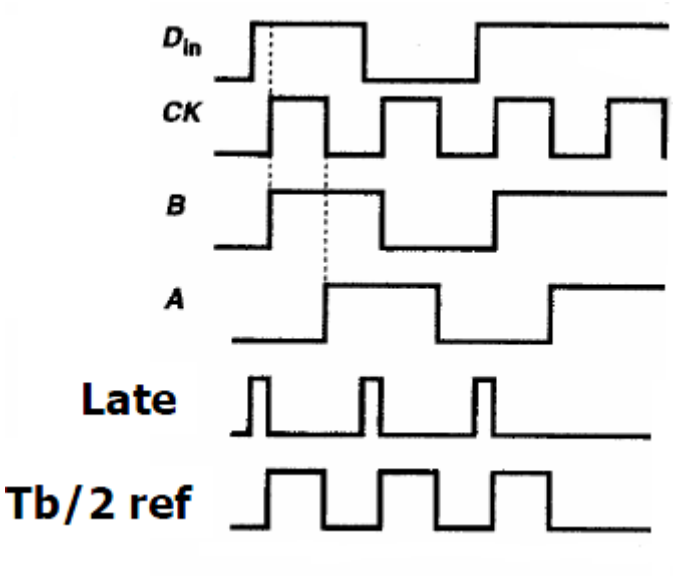


Figure 10: Timing Diagram [7]

The late signal produces a signal whose pulse width is proportional to the phase difference between the incoming data and the sampling clock. A Tb/2 reference signal is produced with a Tb/2 delay.

If the clock is sampling early, the late signal will be shorter than Tb/2 and vice-versa.

2.1.2 Bangbang Phase Detectors

We only have one type, which is Alexander Bangbang phase detector.

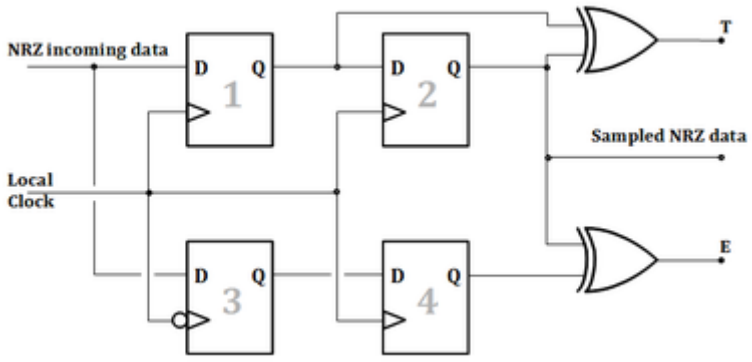


Figure 11: Alexander Bangbang phase detector [8]

Non-linear binary Alexander phase detector only provides information on the phase error not on magnitude. Its characteristics are not linear as shown below.

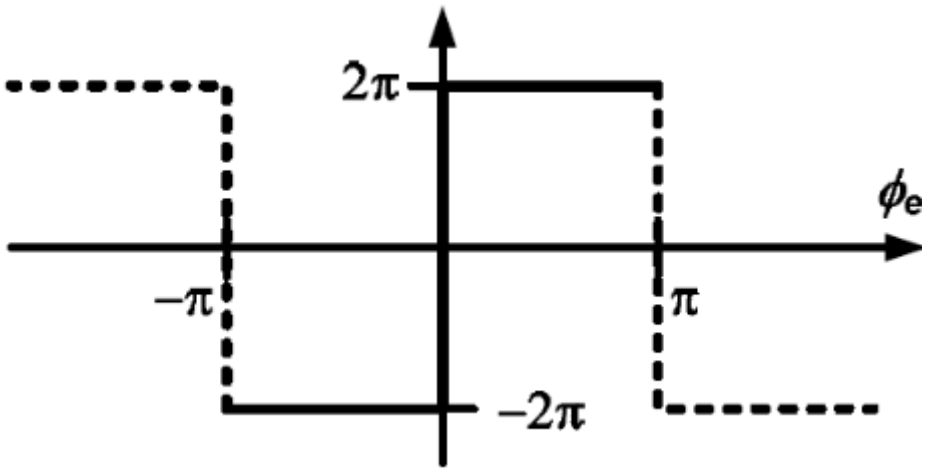


Figure 12: Alexander phase detector characteristics [8]

After analysis of the categories of phase detectors, we can check this table.

Table 1: Comparison between Linear and Bangbang PDs

	Linear PD	BangBang PD
Advantages	Small output jitter	Less sensitive to data patterns
Disadvantages	Nonlinearity for non-uniform data	High output jitter

We made our choice to linear PD (PFD); as it has great advantage of small jitter, which is critical for ADPLL. In addition, detects frequency and phase.

2.2 Loop Filter

There are two types of filters, FIR and IIR filters. The design of both filters is not important, what really makes the main difference between them is the way where each filter is implemented.

2.2.1 FIR Filter

Because of the way FIR filters can be synthesized, virtually the response of any filter you can imagine can be implemented in an FIR structure as long as tap count is not an issue. For example, Butterworth and Chebyshev filters can be implemented in FIR, in order to get the desired response you need large number of taps. That is the reason why we generally use prototypes other than the S domain polynomials as prototypes

for FIR filters. As another example, if it is required to get a band pass filter with two pass bands, or a band pass filter with specific phase properties, FIR will be the best solution as it can do the job with no problem.

The beauty of FIR filters and which is the most important feature is that they can be implemented with integer math. As it is known, the most required features are low power, low cost, portable devices.

These types of processors work great and very common and available in the industry, but seldom have a floating point math core, while integer math is the standard as it is easy, cheap, fast, and low power. So, FIR is used to implement a digital filter.

Therefore, the characteristics of the filter not determines whether FIR or IIR will be used. [9]

2.2.2 IIR Filter

IIR filters are essentially restricted to the well-defined responses that can be achieved from the S domain polynomials such as the Butterworth. However, it is quite difficult to synthesize filter in an IIR structure as required. From the disadvantages of IIR filters, that it cannot be implemented in integer math; as the IIR coefficients cannot be scaled to integer values without having the filter's math calculations explode. For example, scaling the coefficients by 16, which will not create nearly enough significant digits for filtering, will cause the calculations to go exponential. [9]

2.3 DCO

A VCO, or DCO in digital PLLs, is one of the key components in a PLL, and it has a great impact on the PLL's overall performance. It has two types, LC Oscillator and Ring based DCO. These types differ in performance, power, area and applications. [10]

2.3.1 LC Oscillator

There are many advantages for LC VCO. For example, it has a superior phase noise performance. On the other side, LC VCO has a small tuning range, large layout area and possibly higher power, which consider as disadvantages.

The design of LC is shown in figure below, the frequency control for an LC VCO is through an NMOS based varactor, assuming that a diode varactor is unavailable in a standard digital CMOS process. [10]

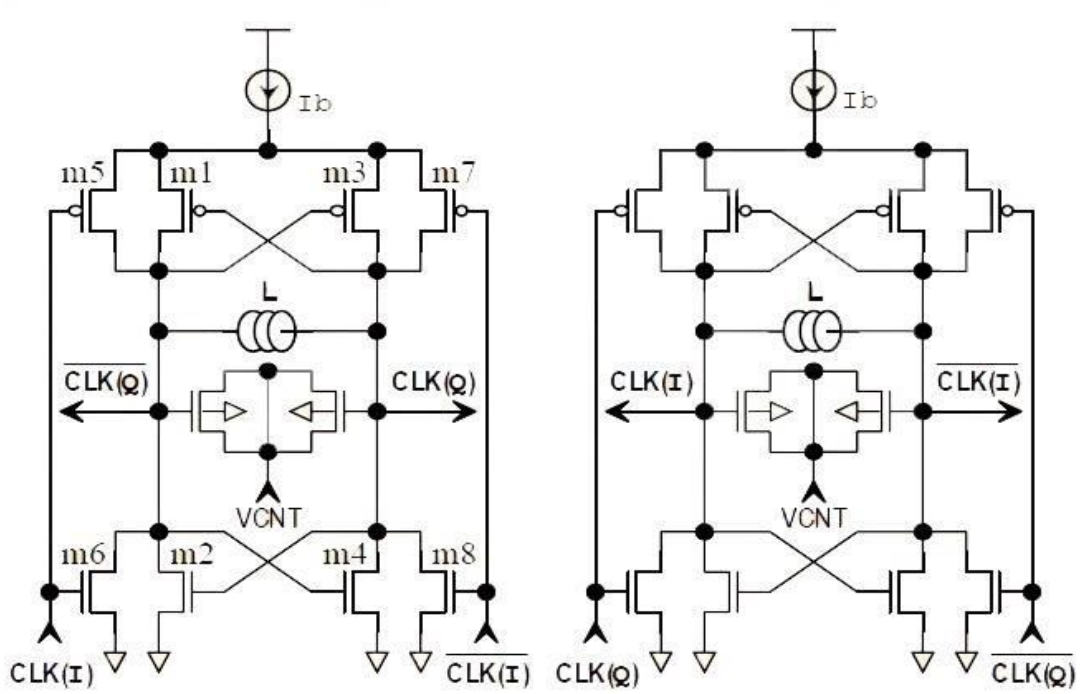


Figure 13: LC VCO [10]

2.3.2 Ring-based VCO

Ring based DCO is often considered to be the first choice; as it is good in simplicity and cost effectiveness. A Ring VCO uses a voltage controlled current sink in parallel with a constant current sink, which keeps the VCO running at a minimum required clock frequency.

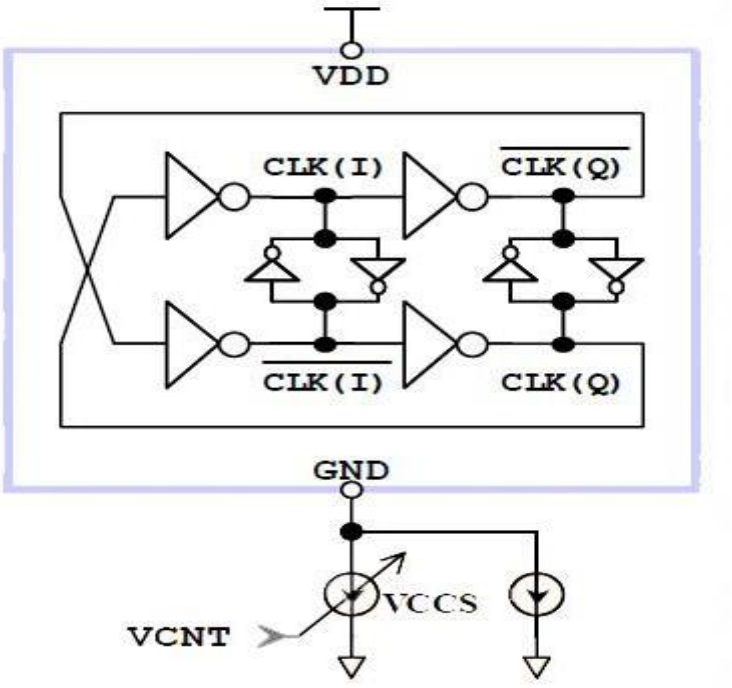


Figure 14: Ring VCO [10]

We have presented a comparison of LC and Ring VCO based PLL designs. The power, layout area, frequency operation and optimization of loop bandwidth to minimize phase noise are all considered in order to obtain a fair comparison. The results show that a Ring VCO based PLL design can meet the performance requirements of a certain high-speed clock synthesizer application if the PLL loop bandwidth is properly optimized. Also, a Ring VCO based PLL has the advantages of larger tuning range and smaller layout area compared with an LC VCO based PLL. Of course, an LC VCO based PLL design exhibits a superior phase noise performance but this advantage is mitigated to some extent by short channel effects. [10]

Table 2: Trade-off between Ring and LC VCO based PLL designs [10]

	Advantages	Disadvantages
Ring-based VCO	Common approach for digital chips. Many ways to control frequency. Multi-phase clock generation. Wide frequency tuning range.	High phase noise → widen loop bandwidth to reduce. High VCO gain → sensitive to disturbance. Poor stability at high frequency.
LC Oscillator	Common approach for RF design. Good stability. Low-term and period jitter filtering. Low long-term and period jitter. Low phase noise.	Large layout area → large area for inductor. Narrow tuning frequency range. Require a lot of characterization. Poor integration and more complicated design.

Chapter 3: ADPLL Architectures

In order to reach our specifications, we did many researches on many papers till we get these 4 papers to operate on.

3.1 First Paper

It is ISSCC conference paper titled "A 2.9-to-4.0GHz Fractional-N Digital PLL with Bang-Bang Phase Detector and 560fsrms Integrated Jitter at 4.5mW Power". Its design is as shown. Figure 4 shows the PLL block diagram, where the standard cell based digital blocks are filled in grey. The TDC is a single D flip flop, which in the following will be referred to as the bangbang phase detector (BBPD). Bangbang PDs are never used in ΔΣ fractional-N PLLs for wireless systems, because they act as hard limiters on the timing error between reference and divider output and their nonlinearity would be responsible for the huge undesirable spurs and noise at the output. In an attempt to resolve this issue, the synthesizer in includes the BBPD in a loop with the ΔΣ modulator, but this results into a type-I PLL with poor phase noise. Moreover, the coarse quantization of the time error provided by a BBPD does not allow canceling the ΔΣ noise at the TDC output.

In order to design a low noise bangbang fractional-N PLL, the divider has been implemented as the cascade of an integer-N divider and a 10b controllable delay stage. In this way, the deterministic quantization error of the ΔΣ modulator is reduced below random noise [11]

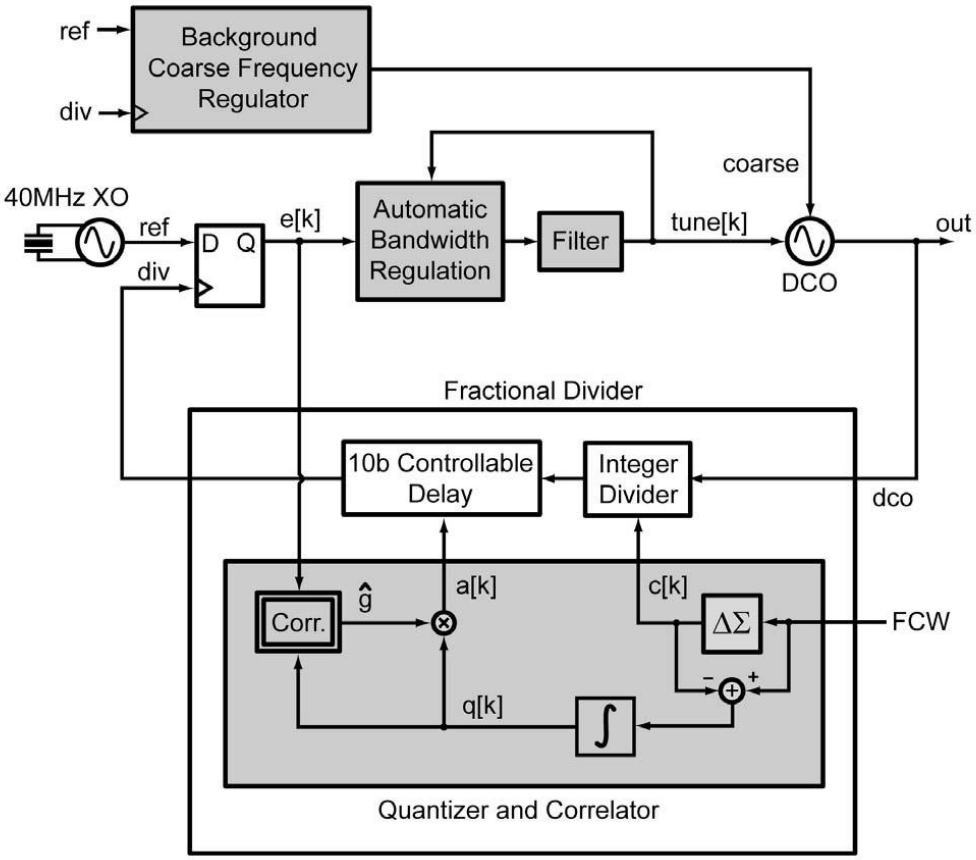


Figure 15: First Architecture of ADPLL [11]

To meet the stringent integral phase noise requirements make use of high resolution and high linearity time-to-digital converters (TDCs). On the other side the disadvantages of those high performance TDCs are the synthesizer design became complicated and dissipate large part of the power budget, which leads to poor jitter power compromise. This paper introduces a fractional-N PLL based on a 1b TDC, achieving jitter of 560fs RMS from 3kHz to 30MHz at 4.5mW power consumption, even in the worst case of fractional spur falling within the PLL bandwidth. The circuit synthesizes frequencies between 2.92 and 4.05GHz with 70Hz resolution. [11]

However, it did not satisfy our needs, as it does not contain enough information.

3.2 Second Paper

It is titled: "A 2.2GHz Sub-Sampling PLL with 0.16psrms Jitter and -125dBc/Hz In-band Phase Noise at 700µW Loop-Components Power". Its hierarchy is as follows.

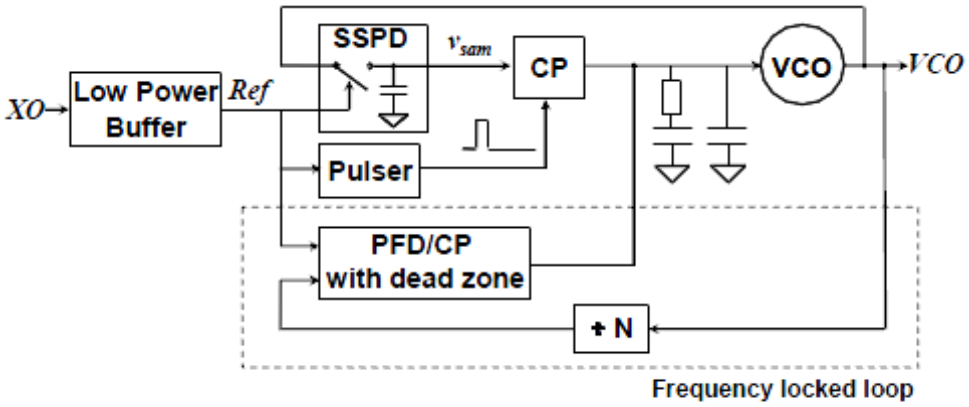


Figure 16: Second Architecture of ADPLL [12]

A dividerless PLL exploits a phase detector that directly samples the VCO with a reference clock. No VCO sampling buffer is used while dummy samplers keep the VCO spur smaller than -56dBc. A modified inverter with low short circuit current acts as a power efficient reference clock buffer. The 2.2GHz PLL in 0.18µm CMOS achieves -125dBc/Hz in band phase noise with only 700µW loop components power.

Clock multiplication PLLs with very low jitter have recently been proposed based on subsampling and injection locking. In a PLL, the VCO dominates the out off-band phase noise while the loop components dominate the in-band phase noise. The sub sampling (SS) PLL can achieve very low in-band phase noise due to many reasons, first the divider noise is eliminated; second the phase detector (PD) and charge pump (CP) noise is not multiplied by N^2.

This paper describes a new SSPLL design aiming to reduce the loop components power while maintaining its superior in-band phase noise performance. [12]

3.3 Third Paper

It is titled “A 1.5GHz 0.2psRMS Jitter 1.5mW Divider-less FBAR ADPLL in 65nm CMOS”.

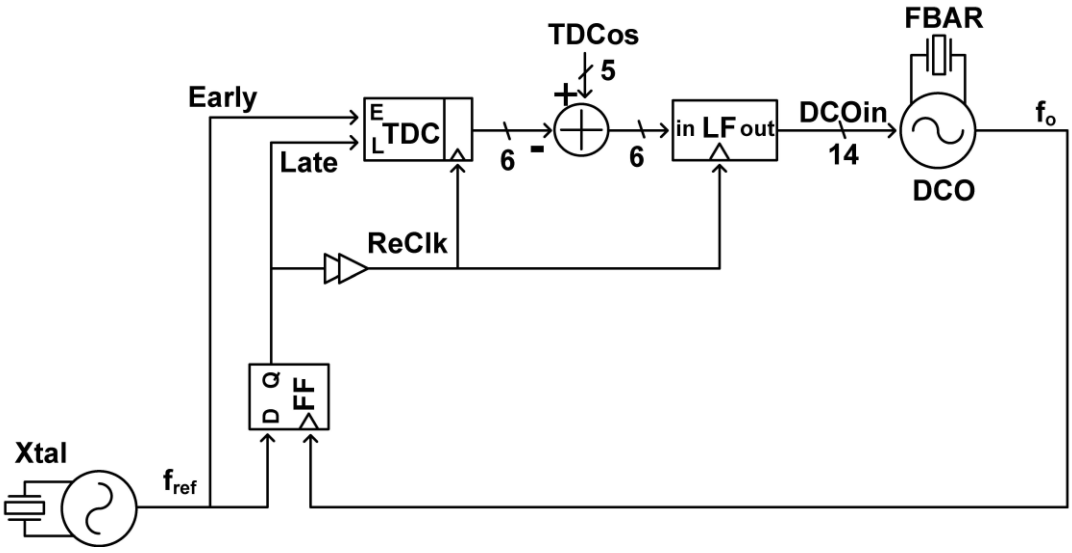


Figure 17: Third Architecture of ADPLL [13]

This paper presents a low power, low jitter, PVT stable film-bulk acoustic wave resonator (FBAR) based all digital phase-locked loop (ADPLL) in a 65nm CMOS process. A power-efficient integer-N ADPLL architecture is introduced.
This architecture is dividerless, which means there is not a divider block in the feedback (unity gain) and it can be seen in the figure above.
From this paper’s title, we find that its RMS jitter 0.19ps in frequency range of 10kHz to 40MHz at 1.5GHz carrier frequency with power 1.5mW. This low power high performance FBAR ADPLL can be used in low power radios, high performance ADCs, and high-speed data links. [13]
As shown in the figure above, it consists of three main blocks, TDC, FIR Filter and FBAR DCO.

3.3.1 TDC Block

We searched for different types of TDC to get the most convenient one to match our specifications, and we found five architectures as follows:

3.3.1.1 Digital Delay Line TDC

As shown in Figure 6, its function is to measure the time interval between two events indicated by the signal edges: start and stop. The earlier pulse, start propagates in a delay line which consists of a series of buffers. The buffer outputs are the inputs of a series of D flip-flops.

The rising edge of the second pulse, stop, latches the status of the delay line. The D flipflop outputs form a thermometer code showing the distance between start and stop rising edges. The time resolution is limited to the smallest buffer delay obtained in the technology which is used to implement the delay line. [14]

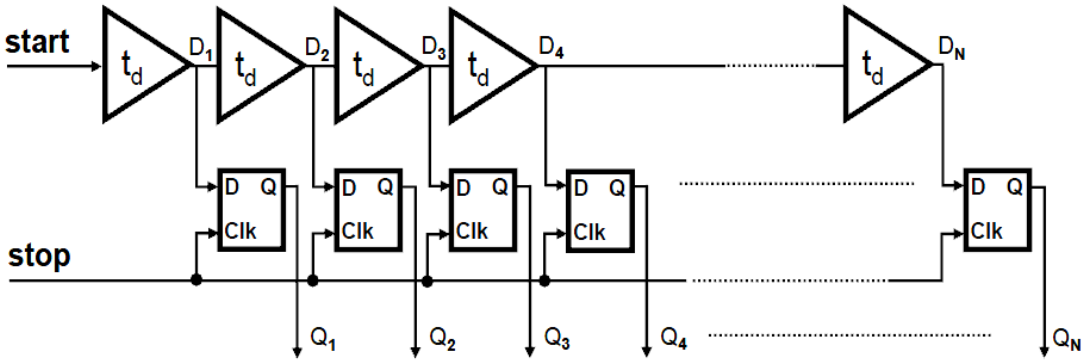


Figure 18: Digital Delay Line TDC [14]

3.3.1.2 Inverter-based delay line TDC

Since the buffer is composed of two inverters, this is an inverter-based TDC which can potentially double the buffer-based TDC resolution rather than the first architecture which its unit delay is limited to a buffer delay. [14]

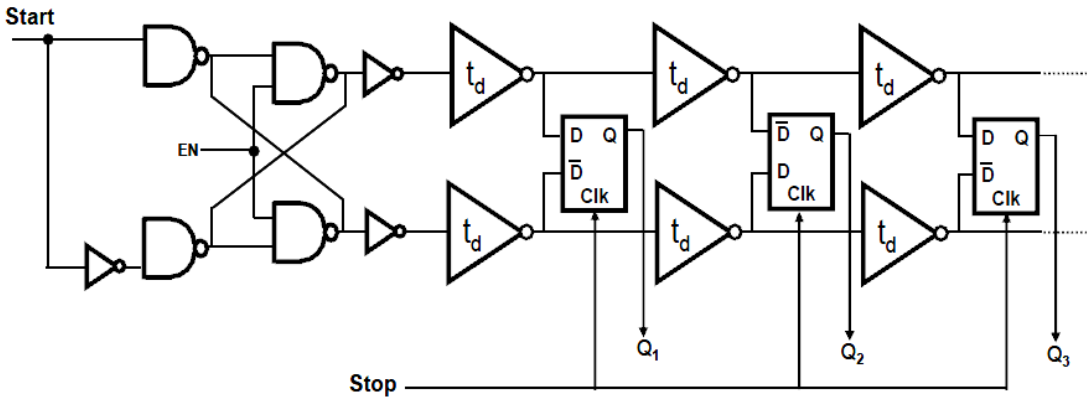


Figure 19: Inverter-based delay line TDC [14]

3.3.1.3 Vernier Oscillator TDC

The Vernier Oscillator time-to-digital conversion method is based on two startable oscillators running at two different frequencies. The difference in the frequencies defines the time resolution. Start and stop signals trigger the oscillators. Since the period of the *stop* oscillator is shorter, the phase of the *stop* oscillator catches up with phase of the start oscillator. In addition, it counted by the counter. [14]

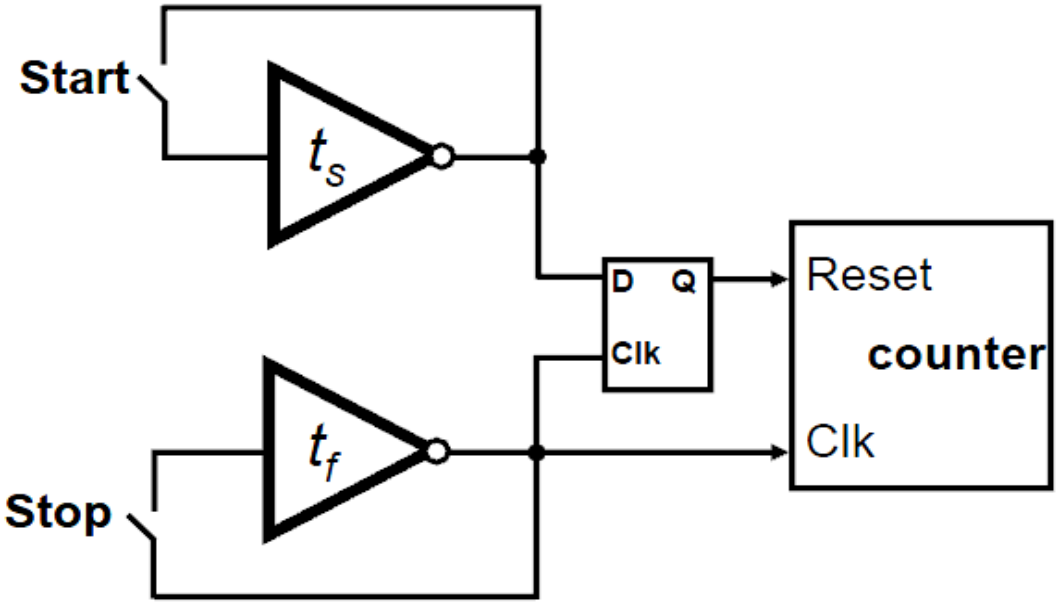


Figure 20: Vernier Oscillator TDC [14]

3.3.1.4 Vernier delay line TDC

The resolution in previous architectures is limited to a gate delay either buffer or inverter-based, which is also limited by the speed of the technology in use. We can get over this by a delay line consisting of two parallel elements, with different delays as a Vernier delay line as shown in Figure 10. There are two parallel delay chains. Our goal is to measure the time interval between start and stop. The first delay of the buffer is in the upper chain (t1) and is slightly greater than the second delay of the buffer is in the lower chain (t2). The start and stop travel through the delay chains until they become aligned. The position 'n' in the delay line at which stop catches up with the start signal, shows the time distance between start and stop as shown in figure. [14]

$$d = n.t_R \quad , \quad t_R = t_1 - t_2 \qquad (1)$$

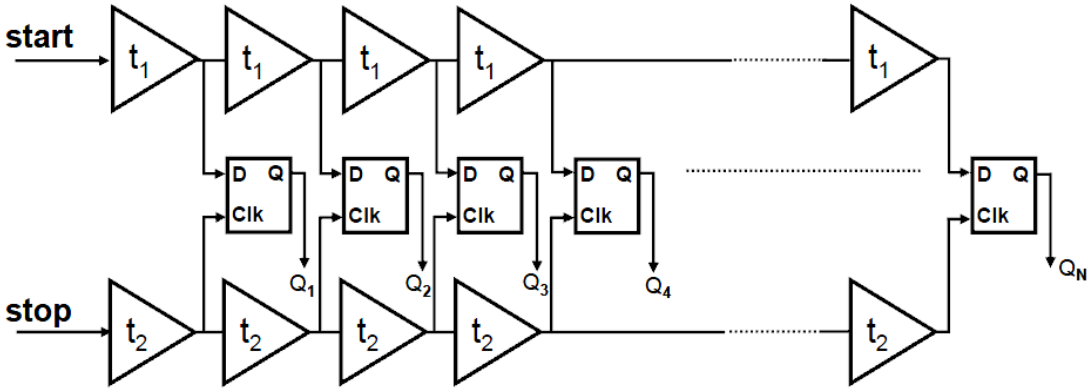


Figure 21: Vernier Delay Line TDC [14]

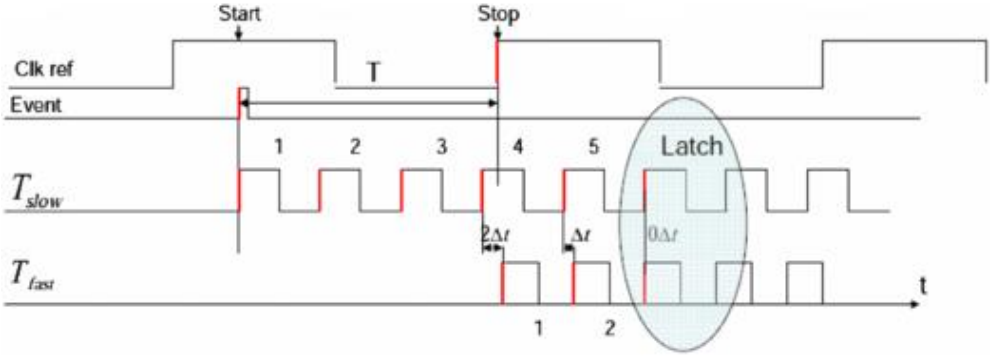


Figure 22: Vernier Delay Line TDC timing diagram [15]

3.3.1.5 Hybrid TDC

Long time interval, the delay line must be very long which degrades the resolution rather than short time intervals in the delay line TDCs discussed so far that have high resolution. However, in some applications, there is a need to simultaneously measure a wide time range with a high resolution. To obtain this goal, a counter is used as a coarse quantizer and a delay line TDC is employed as a fine quantizer resolving the residual error of the counter measurement. [14]

From these different architectures, Vernier Delay Line TDC is the best to obtain the same timing diagram of the design below.

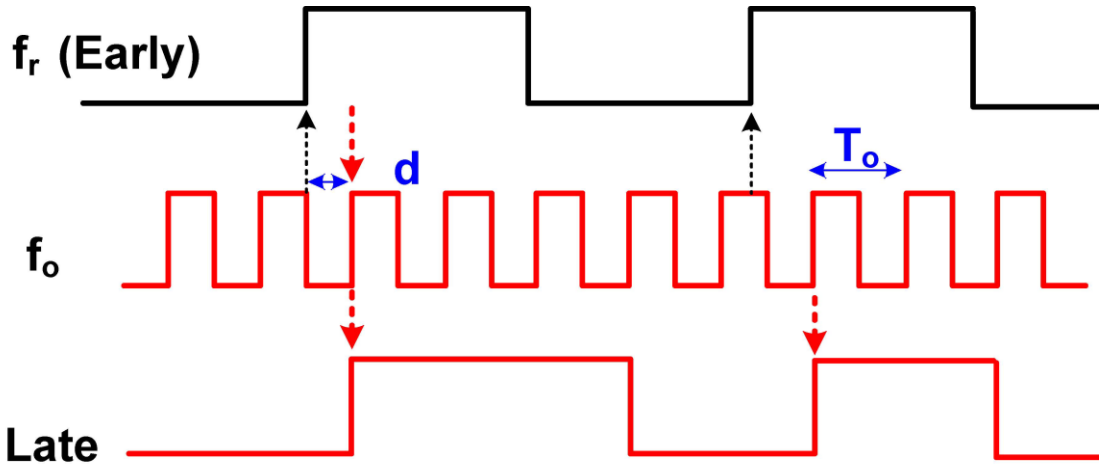


Figure 12: Re-timing and re-sampling of FBAR DCO output with an FF [13]

3.3.2 FBAR DCO

We use DCO in this design with ring oscillator so as the accumulative error would be reduced as well as the jitter. As the advantage of FBAR DCO is producing output frequency which is smooth with very low error. However, the problem here that FBAR cannot be made using CMOS technology, but it has been made by HP technology and the solution was to bring it off-chip, which is not applicable especially for fabrication.

3.4 Forth Paper

This is the paper we got our design from which is titled “A 4.8-GHz Dividerless Subharmonically Injection-Locked All-Digital PLL With a FOM of −252.5 dB”. This ADPLL is presented to achieve both low power and low phase noise simultaneously. To achieve phase locking without a time-to-digital converter this architecture uses a bang-bang phase detector also the power can be reduced by disabling the dividers. In addition, a subharmonically injection-locked technique is used to achieve a low phase noise.

It consumes power of 3.661mW for a supply voltage of 1.1V. The measured phase noise is equal to −122.33dBc/Hz at an offset frequency of 1MHz. RMS jitter is 123.4fs for the offset frequency from 1kHz to 100MHz. The calculated figure of merit (FOM) is equal to −252.5dB. [16]

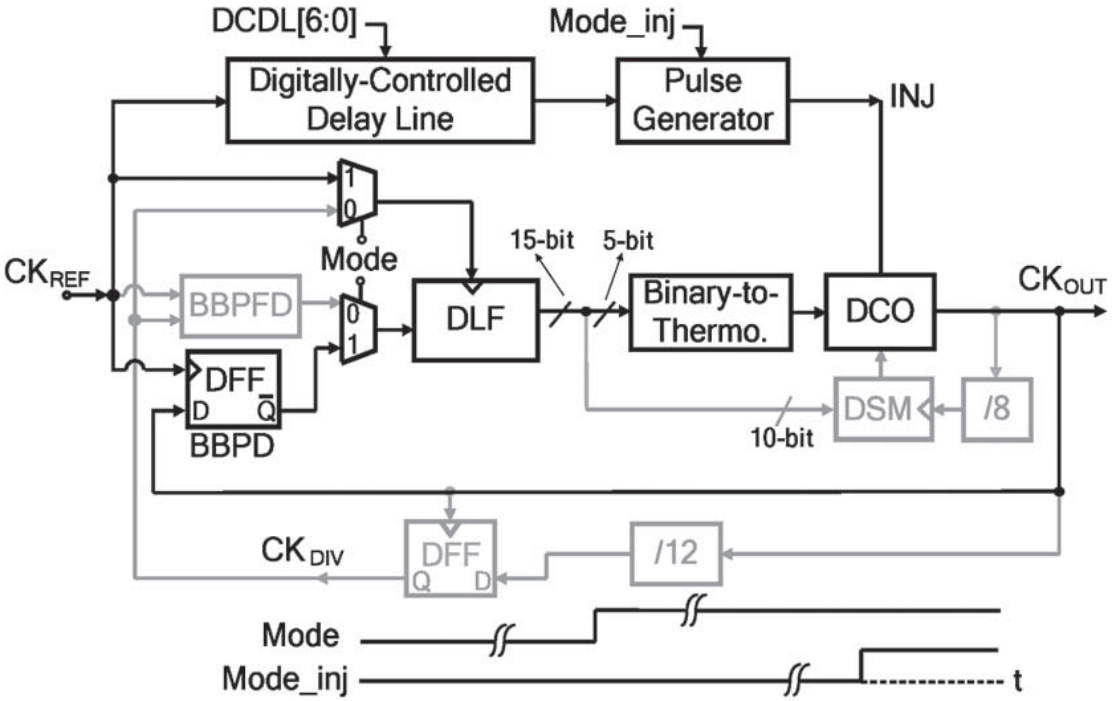


Figure 13: Forth Architecture of ADPLL [16]

The imperfections in advanced CMOS technology are leakage current, poor transconductance of analog devices, area consuming passive devices, and low supply voltage. These imperfections are solved using ADPLLS. The in-band phase noise of an ADPLL is highly dependent on the timing resolution of a TDC. For an ADPLL with a low input frequency, the required area of a high-resolution TDC will be large. A bangbang phase frequency detector (BBPFD) is widely used to eliminate a large high resolution TDC, but the limiting cycle jitter of a BBPFD and the jitter of a DCO are proportional and inversely proportional to the loop bandwidth, respectively. However, under these process variations it is difficult to have an optimal loop bandwidth. DCO with a delta sigma modulator (DSM) is widely used to have the fine frequency resolution and the low phase noise, while the frequency of a DCO is increased, the embedded DSM and the dividers will consume the power significantly. A dividerless subharmonically injection locked ADPLL is presented to achieve a low power and noise clock generator.

This paper is a conventional ADPLL using an LC DCO and a BBPFD works first. This BBPFD is needed first for frequency acquisition second for phase acquisition. By using the BBPFD and a retimed D flip flop (DFF) following the divider, the timing error between CKREF and CKOUT is reduced to an intrinsic delay of a DFF, whereas the conventional ADPLL is locked. This delay is further reduced by using a

bangbang phase detector (BBPD) to tolerate the process and supply variations. The BBPFD cannot be reused since it is connected to the retimed DFF. Thus, a BBPD is needed. In addition, the dividers and a DSM can be turned off to save the power by using this BBPD. Since the timing error between CKREF and CKOUT is small enough, the subharmonically injection locked technique can be adopted to lower the phase noise further. To conclude, this system works in 2 phases: PFD works, PFD stops and PD works and after some time DCDL works too. [16]
We are showing the details of this paper and its blocks in the next chapters.

Chapter 4: Blocks of the Proposed ADPLL

As we mentioned in the last section that the conventional architecture works as two PLLs in one circuit, as PFD works first trying to lock on the phase and frequency of the reference clock. Then after stabilizing with a certain value of jitter, a pulse of multiplexer stops the PFD and selects the PD to work as a second PLL to refine the jitter by locking on the phase difference. After a while the injection pulse makes the upper path of DCDL and Pulse Generator works producing a delayed pulse by a certain value which is upper than the half of the period of the reference clock.

In this chapter, we show our modification on the blocks of the chosen design and its internal structure schematically and by Cadence.

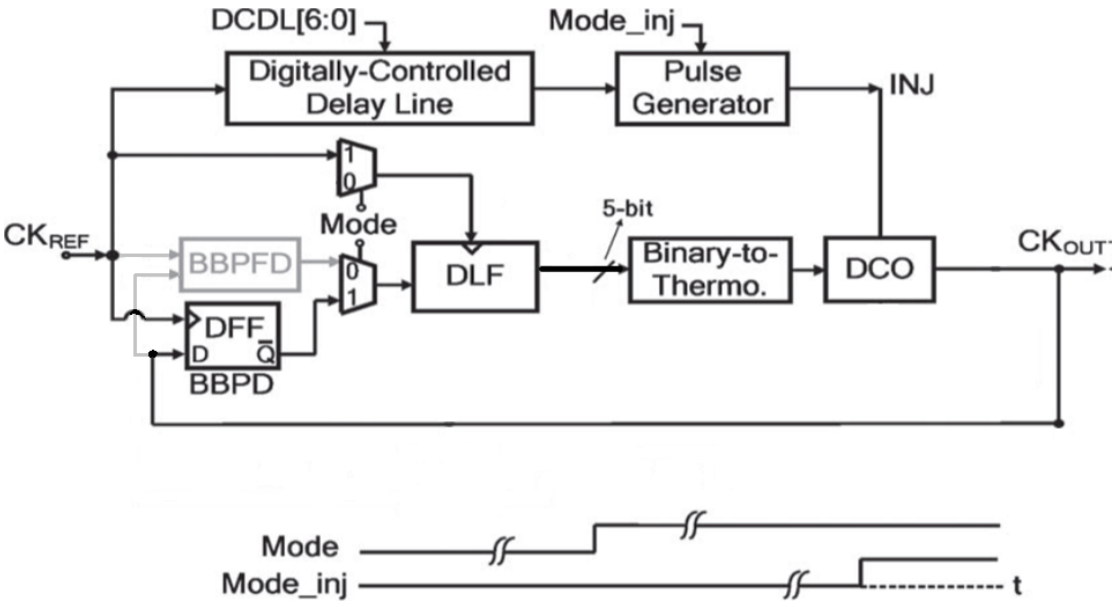


Figure 23: Proposed Architecture of ADPLL

4.1 DCO

It consists of LC oscillator and two arrays each array has 32 elements. LC oscillator generate frequency f = $\frac{1}{2\pi\sqrt{LC}}$ Hz. Therefore, to change the oscillator output frequency; there are two ways to generate different frequencies. The first method is by changing the inductance due to different digital inputs. And the second method is by changing the capacitance due to different digital inputs. The second method is the one used in our design. There are 32 elements in each array. Each element consists of one capacitor and

NMOS transistor in series. NMOS transistor used as a switch to change the oscillator capacitance to change the output frequency. The oscillator input is 32 bits generated from binary to thermometer decoder. To decrease the output frequency, increase the oscillator capacitance by increasing the number of transistors that turned on. There was a problem in this element design as well as that the total capacitance of each element not the capacitor value only but also as well the equivalent capacitance, the capacitor value and the transistor parasitics are in parallel so the dominant capacitance is the parasitic capacitance of the transistor which is very small and there are not large range of oscillation also it needs very large inductor to achieve the frequency range needed.

To solve this problem, the size of the transistor must be large enough to increase the parasitic capacitance of the transistor and it reach the capacitance value. Then the equivalent capacitance is the capacitance value divided by two. LC oscillator generate sinusoidal signal so the output must be square wave so to achieve it, the oscillator must reach saturation at VDD and GND. To decrease the lock time and make the LC oscillator reach saturation it need large power to generate a square wave with small power and fast locking time, therefore comparator is used to achieve this.

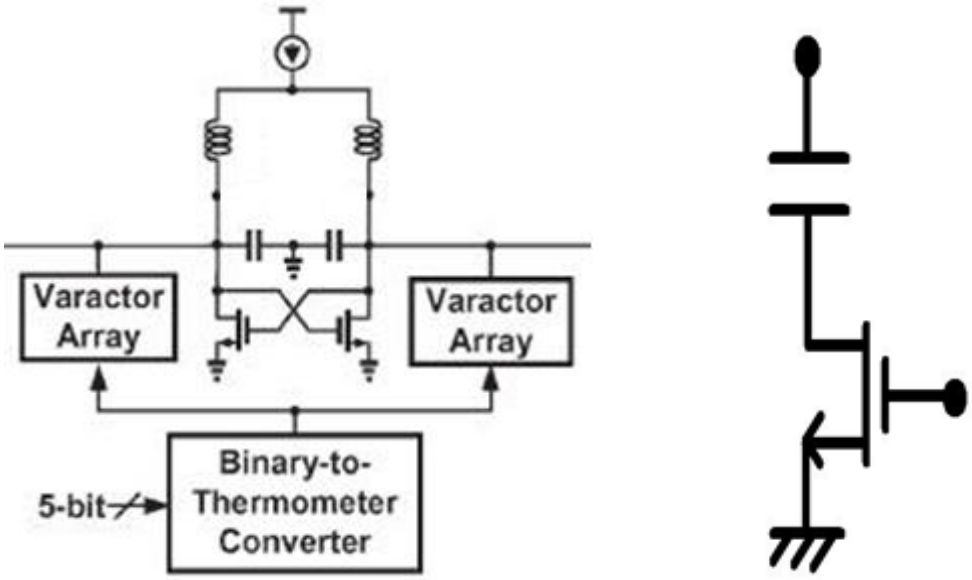


Figure 24: Left: LC Oscillator, Right: Array Element

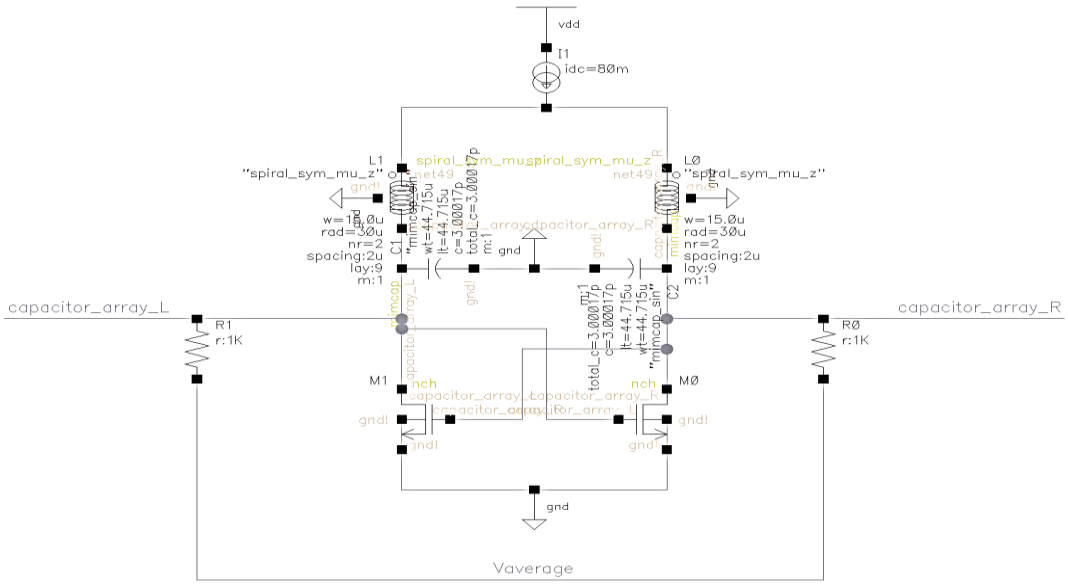


Figure 25: LC by Cadence

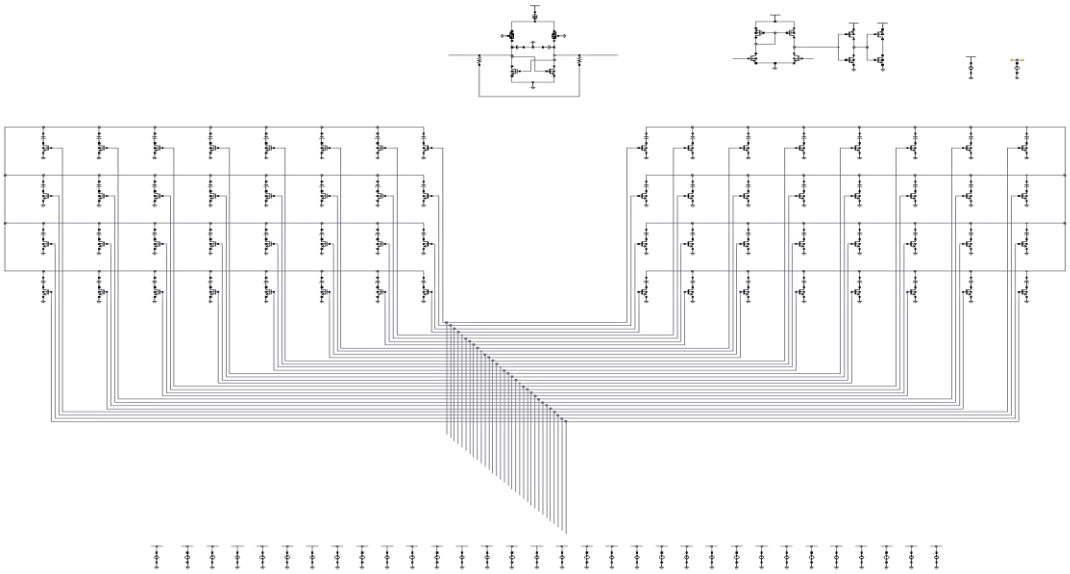


Figure 26: Oscillator Transistor level

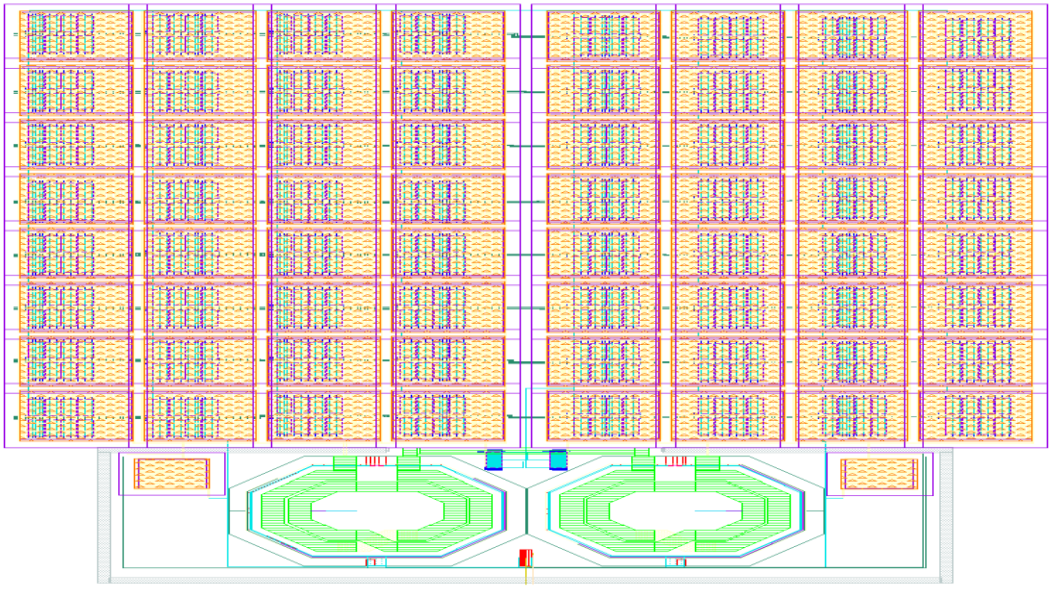


Figure 27: DCO Layout

Therefore, to test the function of the LC oscillator combined with the comparator, we enter the input (11111000000000000000000000000000) then we obtain an output frequency of 636.9MHz, as shown.

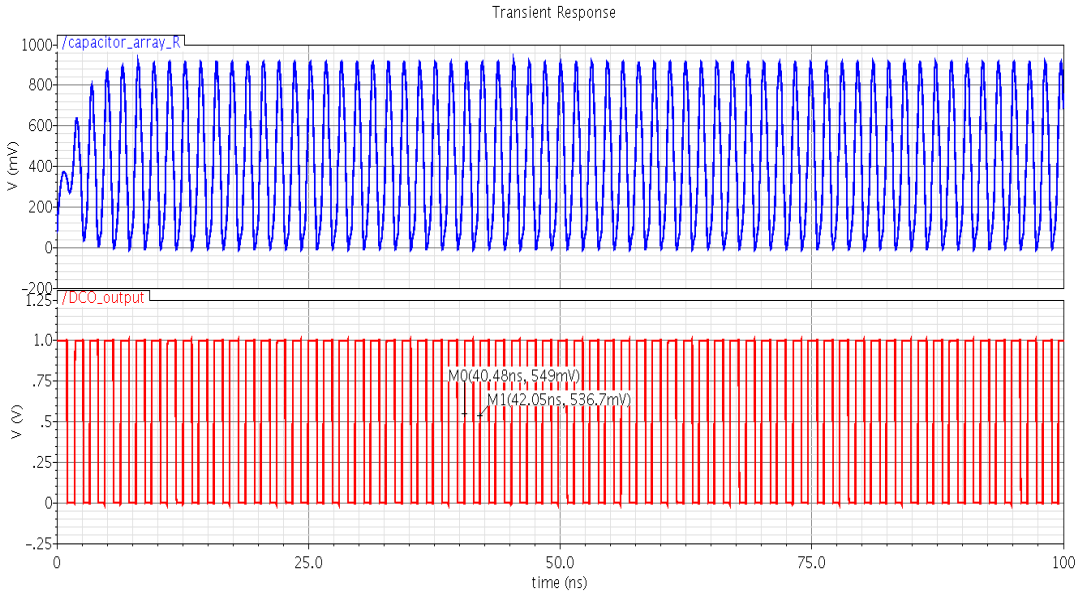


Figure 28: Output Frequency of LC Oscillator at 636.9MHz

Then we input a different one of (11111111110000000000000000000000) to get a different frequency, and the output frequency is 581.4MHz, as shown.

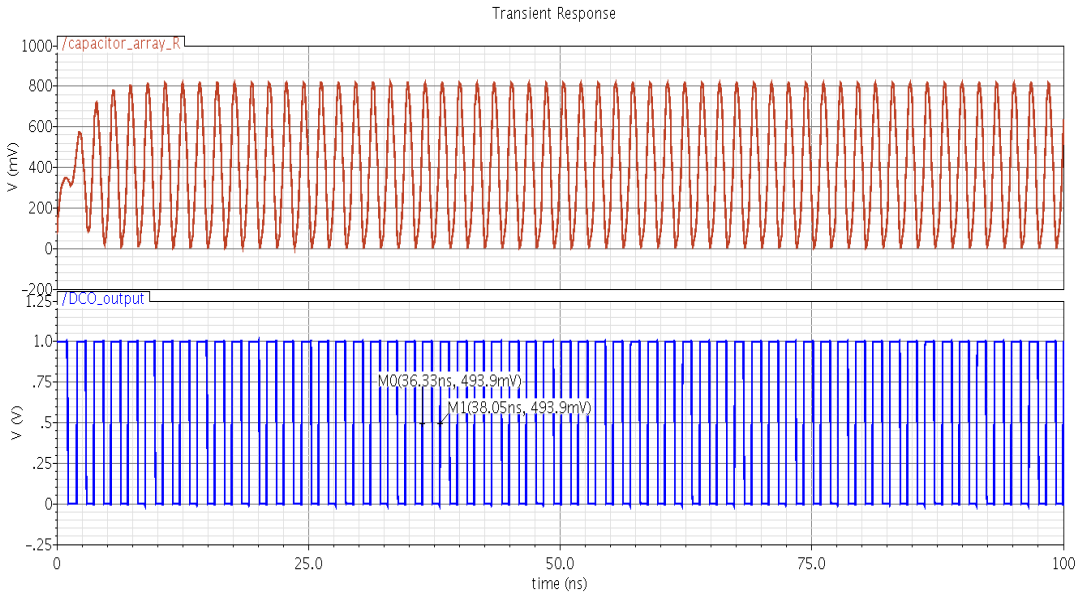


Figure 29: Output Frequency of LC Oscillator at 581.4MHz

Thus, the LC oscillator and comparator are generating the frequencies as needed.

4.2 The Comparator

Used to generate square wave from sinusoidal wave, we also use it to reduce power. It consists of differential pair where its output connected to a buffer.

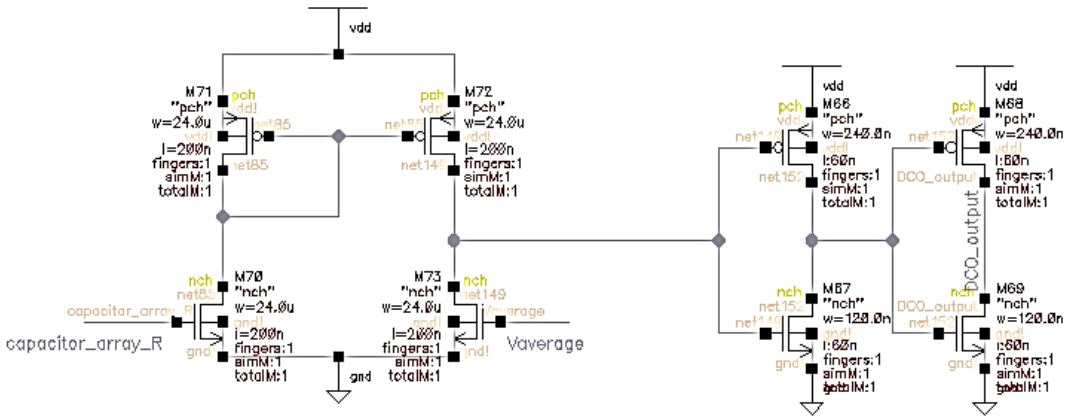


Figure 30: Comparatot Schematic

4.3 Binary to Thermometer Decoder

Firstly, we made it using Verilog code to check the function of the completely system then we made it in gate level. It consist of five input bits and 31 output bits where the bit number 32 always connected to VDD. It consists of three arrays: NAND array, inverter array and the main array that generates the output. The main array consists of 496 NMOS and 496 PMOS transistors used as switches for VDD if the transistor is PMOS and for GND if the transistor is NMOS. Each NMOS transistor's source connected to the GND and each PMOS transistor's source connected to the VDD. The main array has 32 rows each row is the minterm of each input. The NAND array generates the minterms of the output and the output connected with the input of the main array to turn only one row at a time. The third array is the inverter array, which generates the complement of the input and the output of the NAND array.

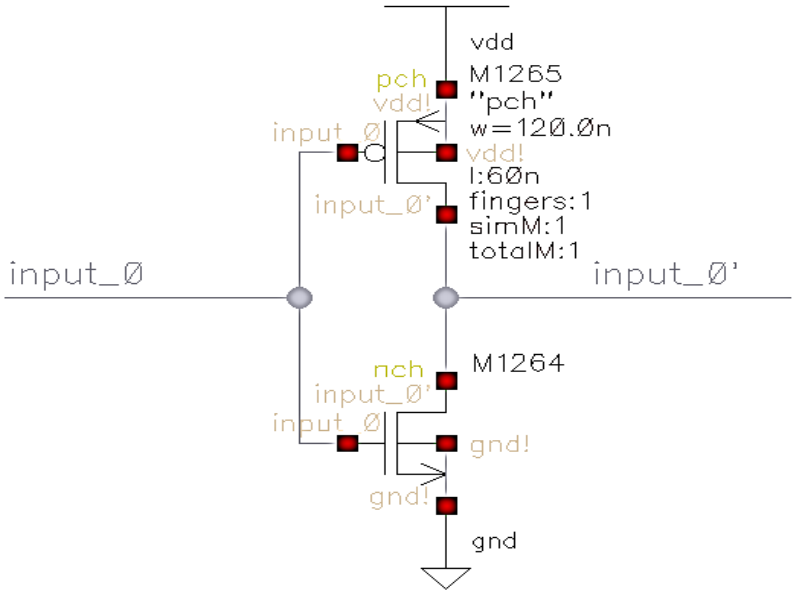


Figure 31: Inverter

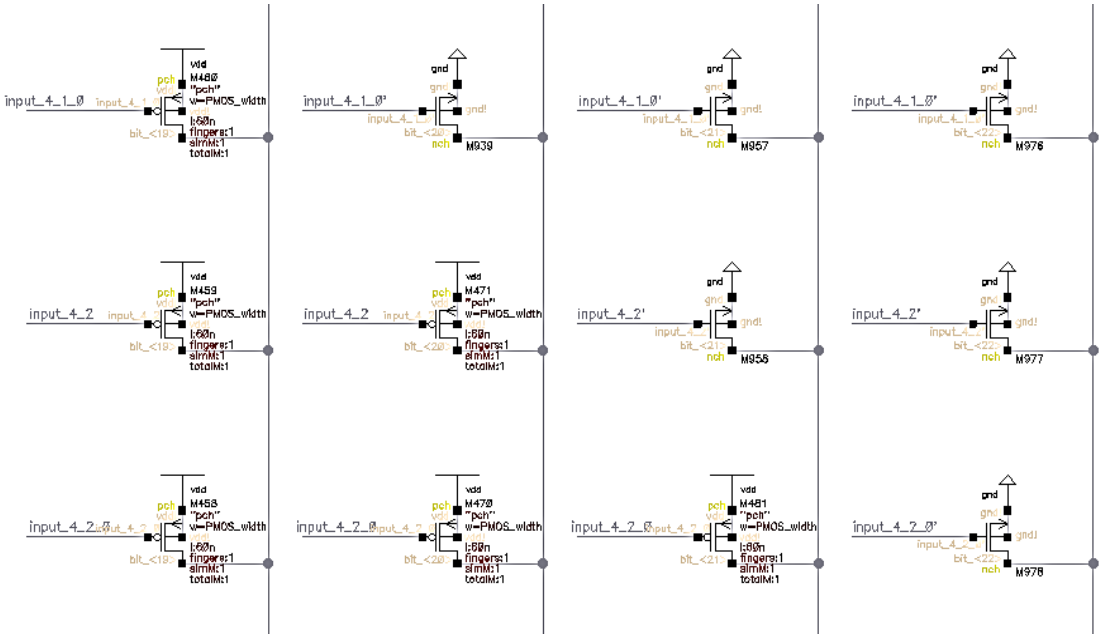


Figure 32: Decoder

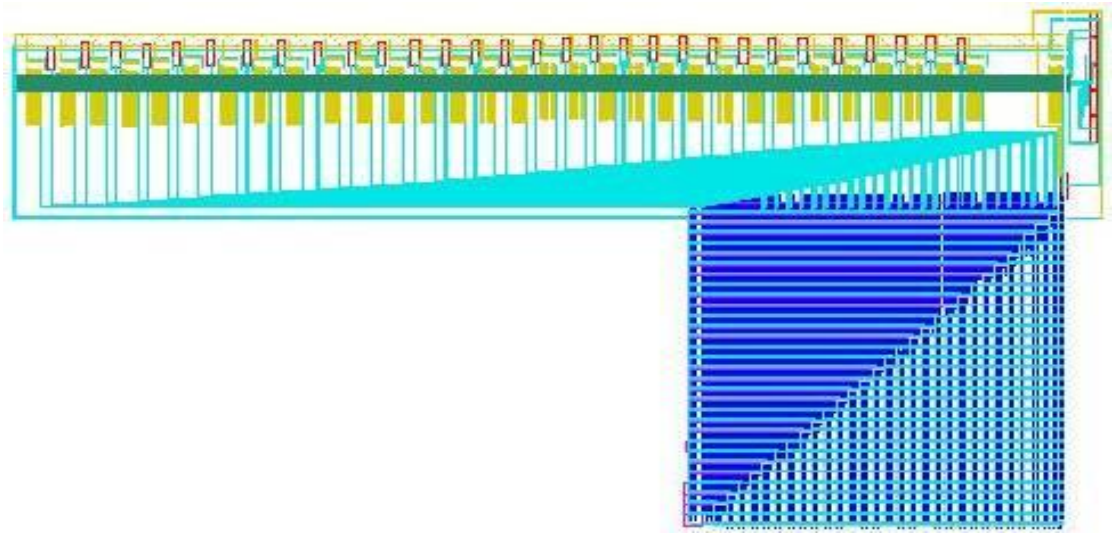


Figure 33: Binary to Thermometer Decoder Layout

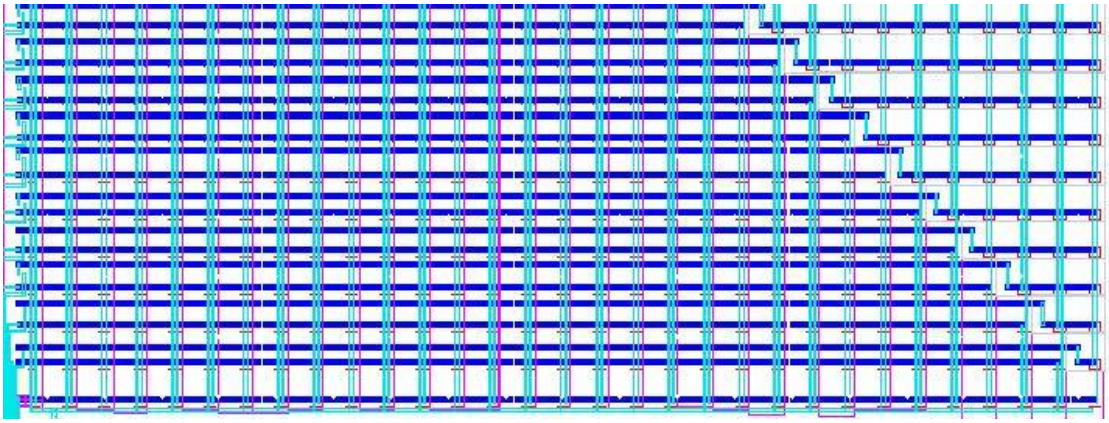


Figure 34: Zoom in version of Layout

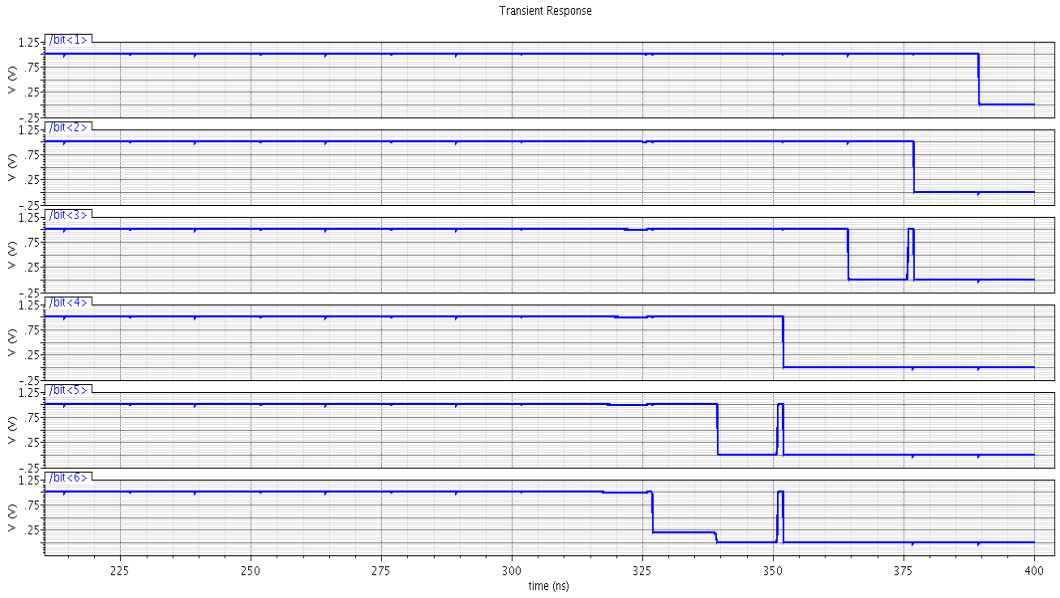


Figure 36: Timing Diagram of Binary to Thermometer Decoder

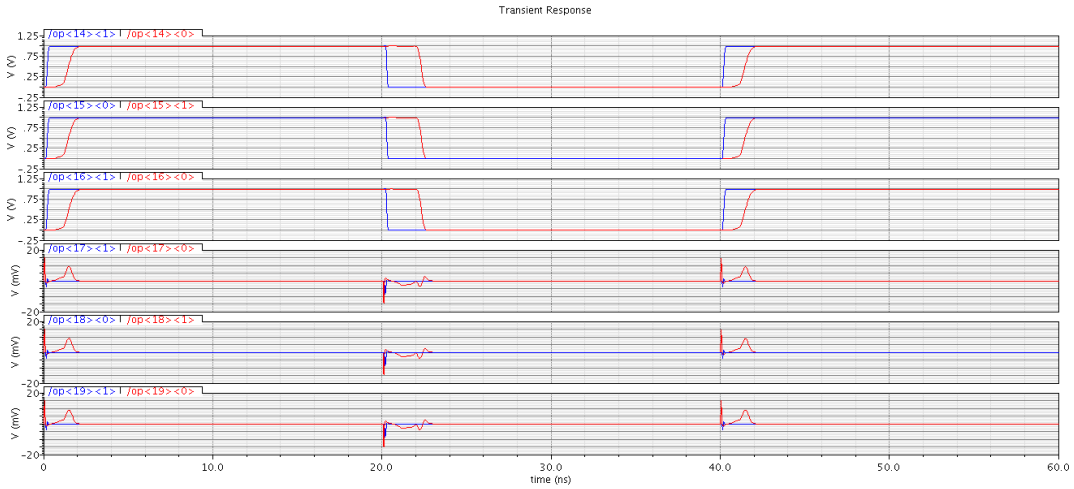


Figure 35: Simulation results before and after layout

In the above figure, we compare between the simulation results of Binary to Thermometer Decoder before layout and after it.

4.4 Phase and Frequency Detector (PFD)

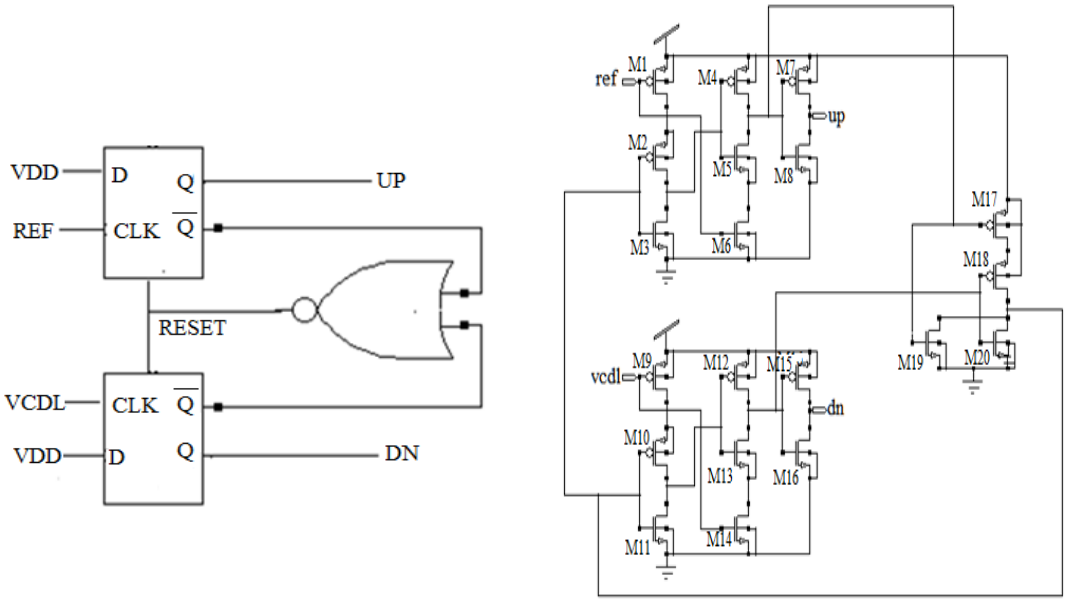


Figure 37: Left: Schematic of PFD, Right: Transistor Level [17]

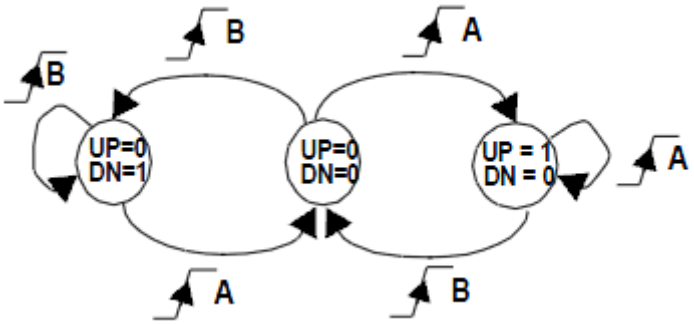


Figure 38: State Transition Diagram [18]

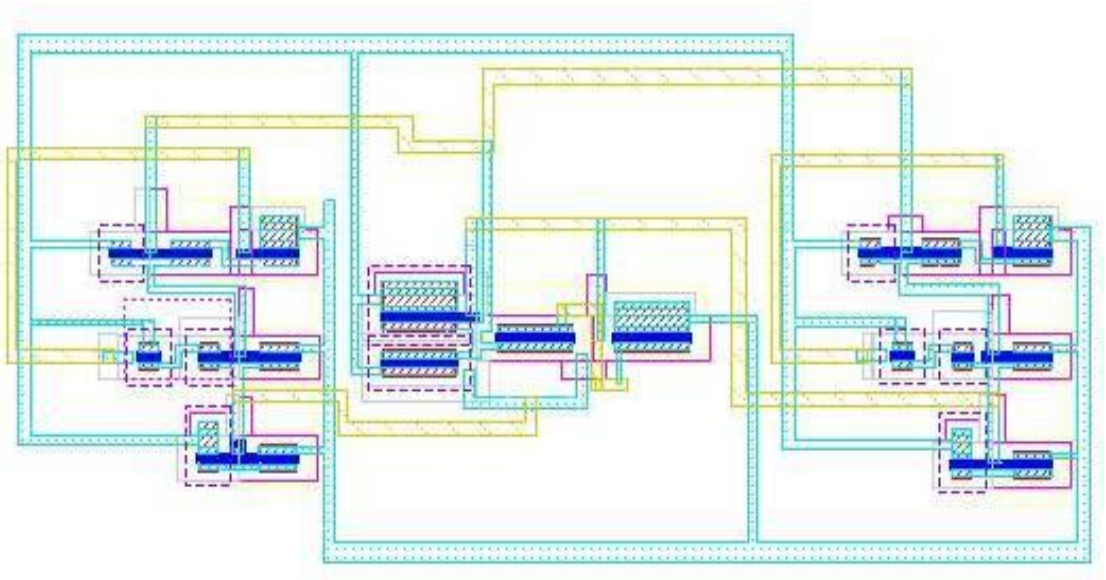


Figure 39: PFD Layout

The PFD is a state machine with 3 states. Where both UP and DN outputs are assumed to be initially low. When input A leads B, the UP output is asserted on the rising edge of input A. While the UP signal remain in this state until a low to high transition occurs on input B. At that time, the DN output is asserted, which cause both flip flops to reset through the asynchronous reset signal. Notice that a short pulse proportional to the phase error is generated on the DN signal, and that there is a small pulse on the DN output, whose duration is equal to the delay through the NOR gate and register reset delay. The pulse width of the UP pulse is equal to the phase error between the twosignals. For the case when input B lags A the roles are reversed, and a pulse proportional to the phase error is generated on the DN output. If the loop is in lock, short pulses will be generated on the UP and DN outputs.

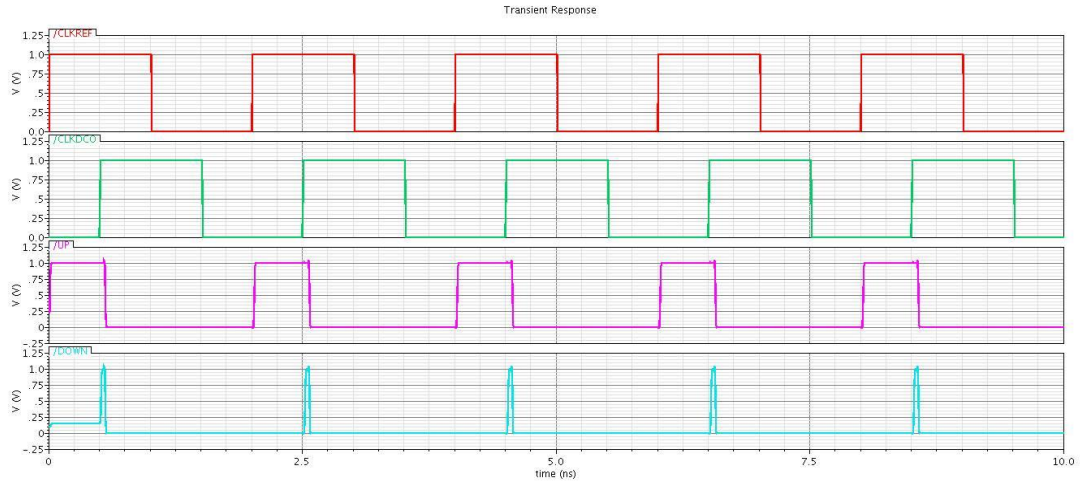


Figure 40: Phase Acquisition

This circuit also consider as a frequency detector, as it is measure the frequency error. There are two cases ,The first case when A has higher frequency than B, the PFD generates a lot more UP pulses where the average is proportional to the frequency difference, while the DN pulses average close to zero. The second case exactly the opposite when B has a frequency larger than A where many more pulses are generated on the DN output than the UP output.

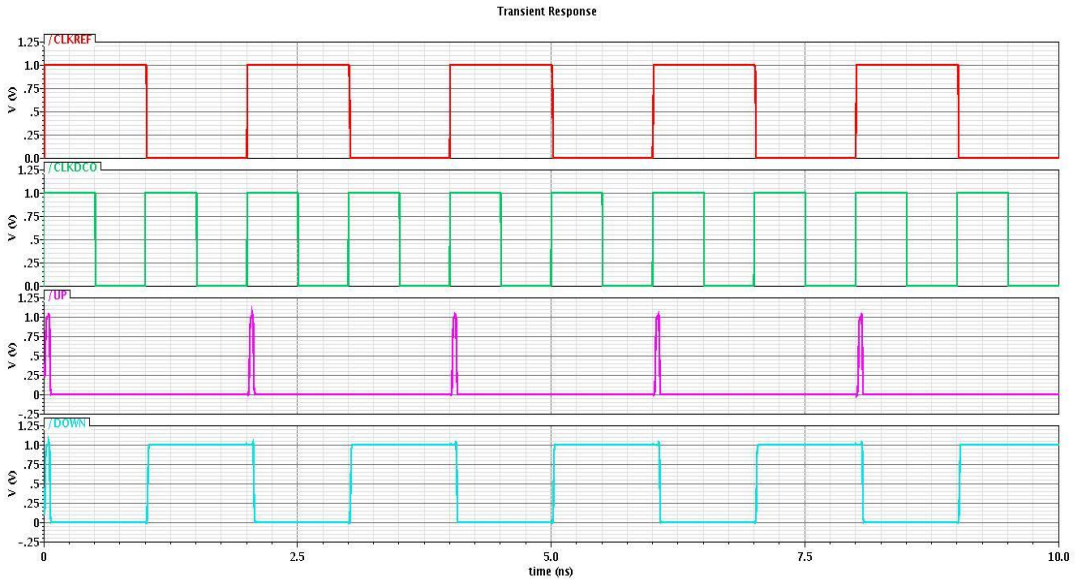


Figure 41: Frequency acquisition, the reference clock is 500MHz and DCO clock is 1GHz

The phase characteristics of the phase detector is shown in this figure. Notice that the linear range has been expanded to 4π.

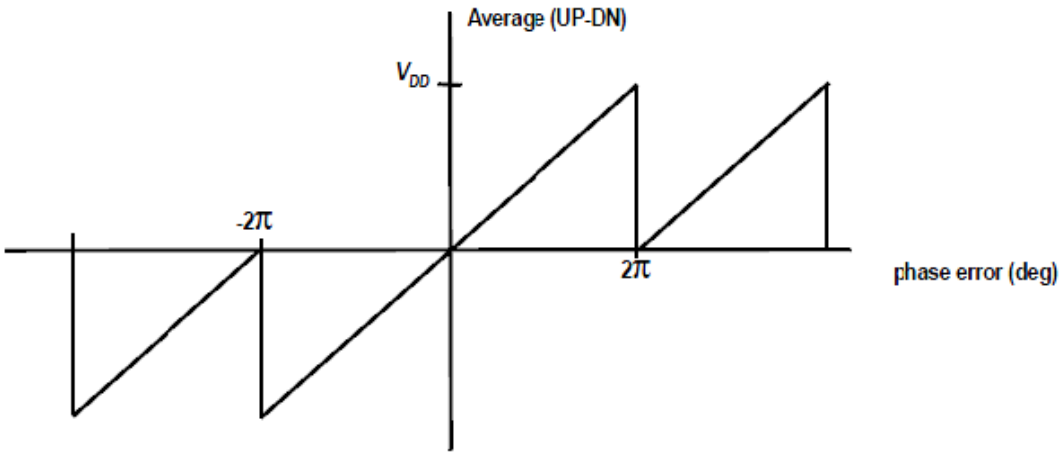


Figure 42: Phase Characteristics [18]

4.5 Bangbang Phase Detector (BBPD)

It has two inputs and one output. The two inputs are the reference clock and the output of the comparator, which is connected to the DCO. Its output is the phase difference between the two signals. Maximum frequency is 10GHz, and its resolution is 20 psec.

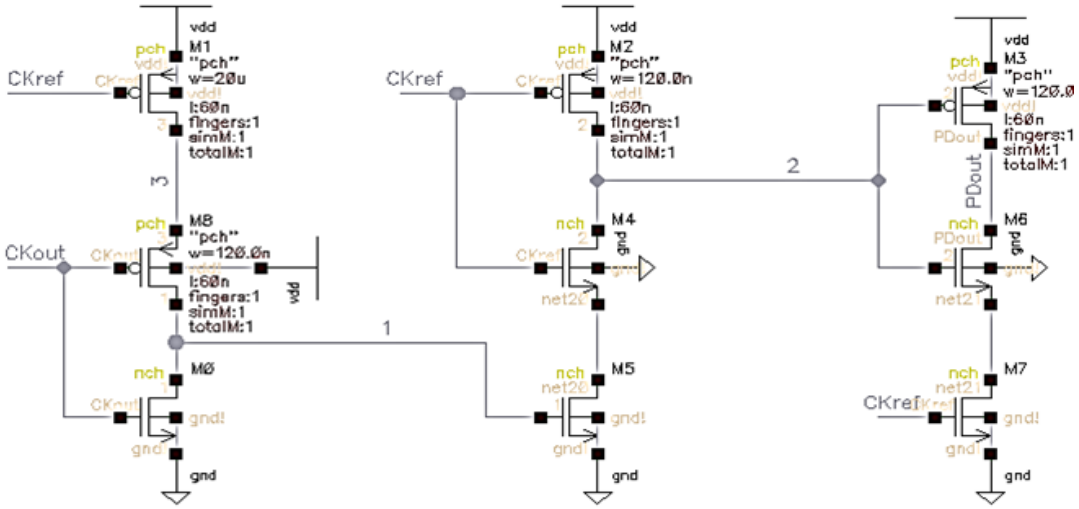


Figure 43: BBPD Schematic

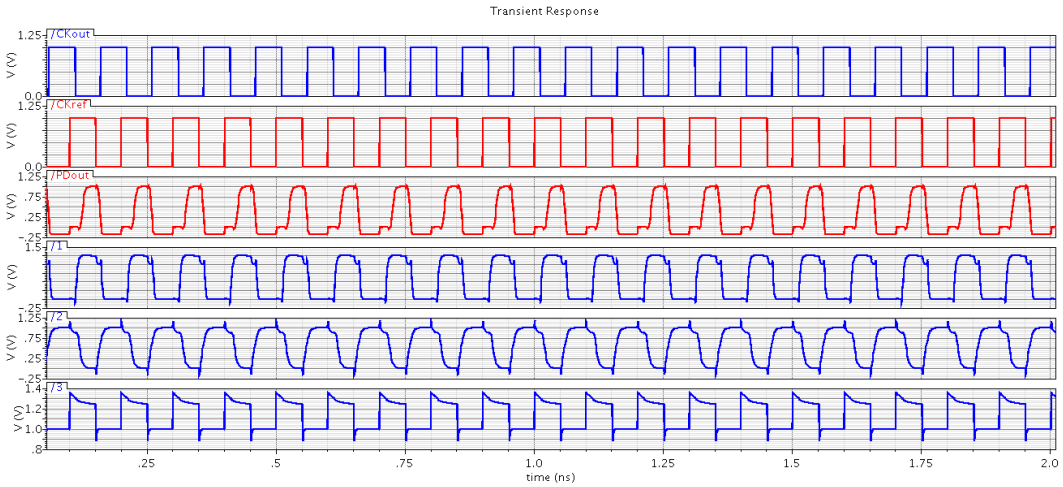


Figure 44: Timing diagram of BBPD

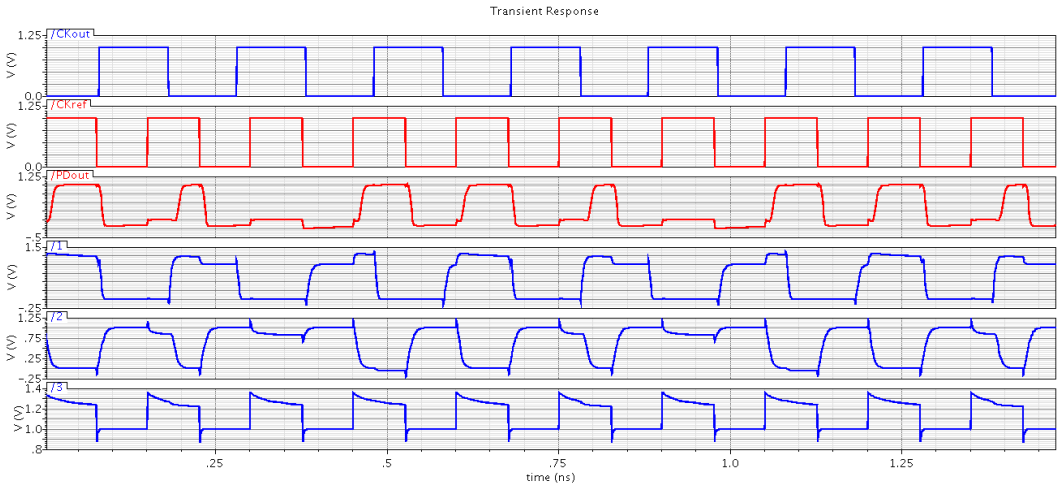


Figure 45: Zoom-in version of Fig. 37

4.6 Digital Loop Filter (DLF)

Firstly, we made it with a Verilog code to check the functionality of the whole system. Then we made it in gate level in which we choose a 5-bit synchronous binary up/down presettable counter.

The advantage of this counter is the preset to any desired starting value, which will improve the lock time of our ADPLL by selecting the starting value to be half of the range (10000). This is done either asynchronously or synchronously. This presetting operation is also known as loading; hence, its name is loadable counter. The figure below shows a 3-bit of the 5-bits asynchronously presettable synchronous up/down counter. [19]

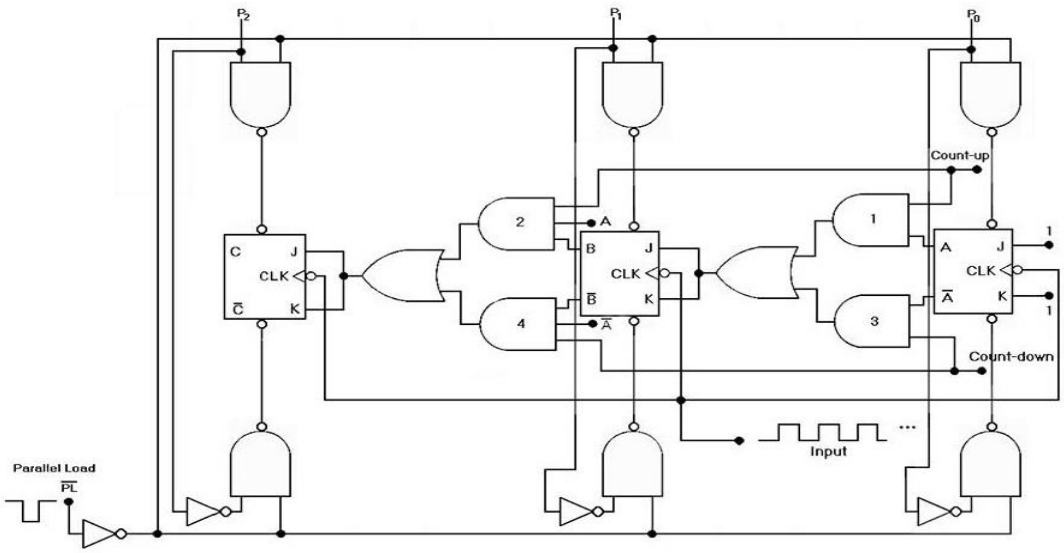


Figure 46: Up/Down Counter

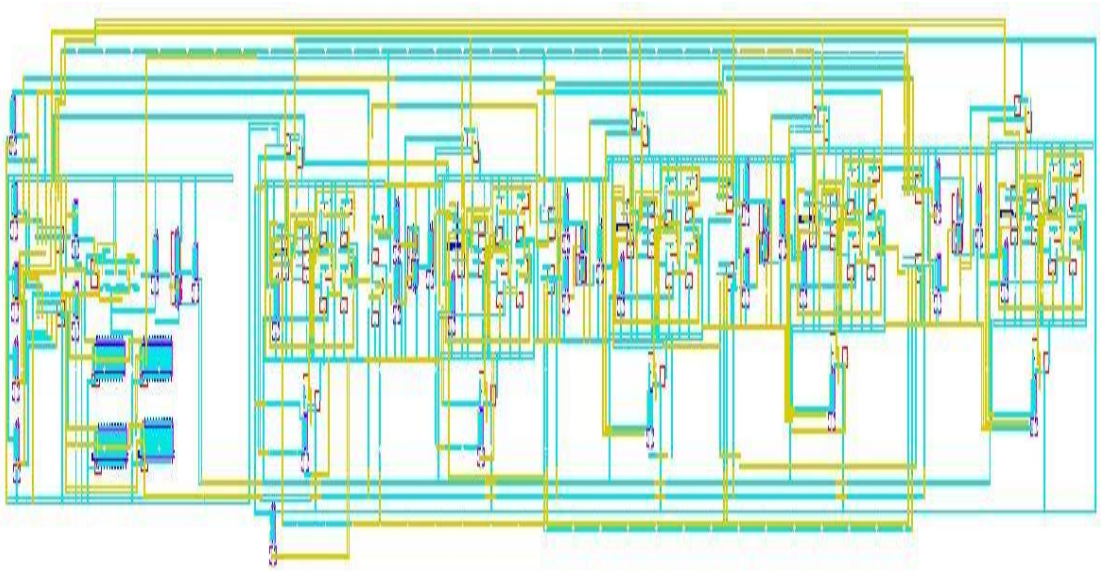


Figure 47: DLF Layout

By using the speed up technique, we overcome the problem of additive propagation delay. From the figure above, we can see that the third flip-flop gets its J-K input from the output of a 2-input AND gate and the fourth flip-flop gets its input from a 3-input AND gate and so on. Therefore, for our 5-bit counter, we are going to need:

- One 4-input AND gate,
- One 3-input AND gate,
- One 2-input AND gate,
- One 1-input AND gate,
- and one 0-input AND gate.

In the diagram above for the completely 5-bit counter, the asynchronous PRESET and CLEAR inputs are used to perform the asynchronous presetting. The counter is loaded by applying the desired binary number to the inputs P4, P3, P2, P1 and P0 and a LOW pulse is applied to the PARALLEL LOAD input, not (PL). This will asynchronously transfer P4, P3, P2, P1 and P0 into the flip-flops. This transfer occurs independently of the J, K, and CLK inputs. As long as not (PL) remains in the LOW state, the CLK input has no effect on the flip-flop. After not (PL) returns to high, the counter resumes counting, starting from the number that was loaded into the counter.

For the example above, say that P4 =1, P3 = 0, P2 = 0, P1 = 0, and P0 = 0. When (PL)is low, these inputs have no effect. The counter will perform normal count-up or downoperations if there are clock pulses. Now let us say that (PL) goes HIGH at Q2 = 0, Q1 = 1 and Q0 = 0. This will produce LOW states at the CLEAR input of Q0, Q1, Q2,Q3 and the PRESET inputs of Q4. This will make the counter go to state 10000 regardless of what is occurring at the CLK input. The counter will remain at state 101 until (PL) goes back to LOW. The counter will then continue counting from 10000.

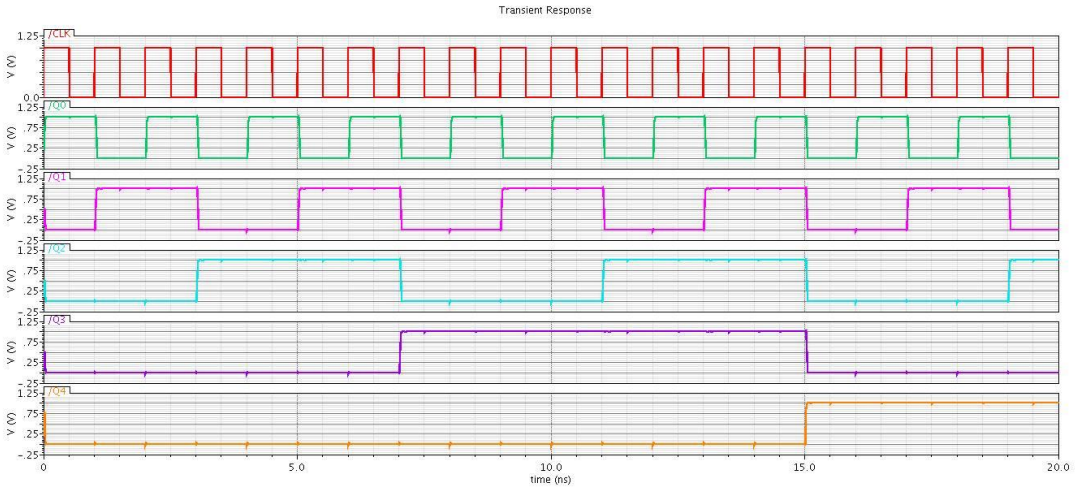


Figure 48: Timing diagram of up counting

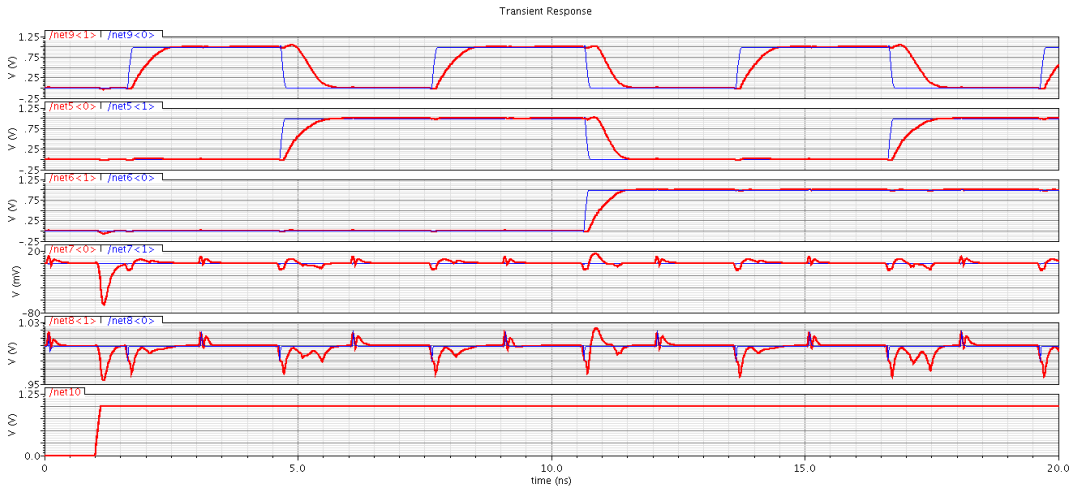


Figure 49: Simulation results before and after layout

Last figure represent the comparison between the simulation results of the Loop Filter performing counting up, before and after layout.

4.7 DCDL and Pulse Generator

There are many types of DCDL, like NAND-based and Mux DCDL. In our design, we chose Mux DCDL as shown.

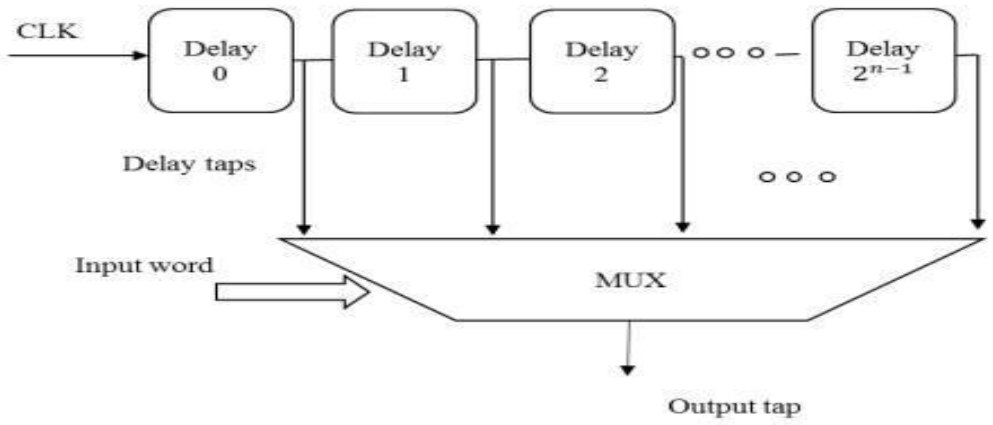


Figure 50: DCDL based on Mux [20]

It consists of 64 delay cells each of resolution 44ps to cover the range needed, which is from 2ns (half the period of the maximum frequency 1Ghz) to 4.8ns (half the period of the minimum frequency 415Mhz). We select the delay needed simply by 6-bit control word of a 64 Mux, but the problem here is that the Mux produces considerable delay by itself which did not harm us as the minimum delay needed is high. Therefore, we made a delay cell, which produced with the Mux delay of 2ns, as shown in the figure below. [20]

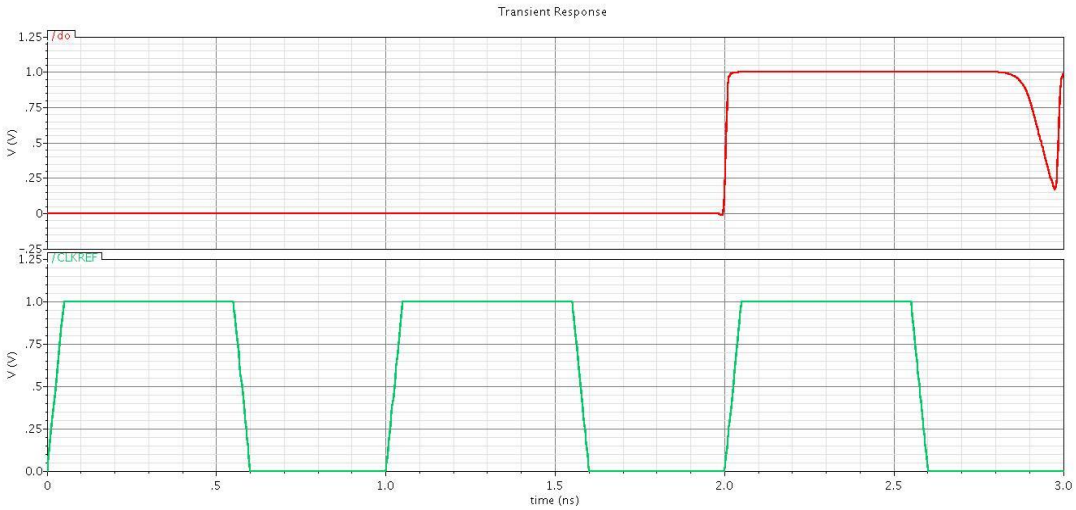


Figure 51: Timing diagram showing the minimum delay

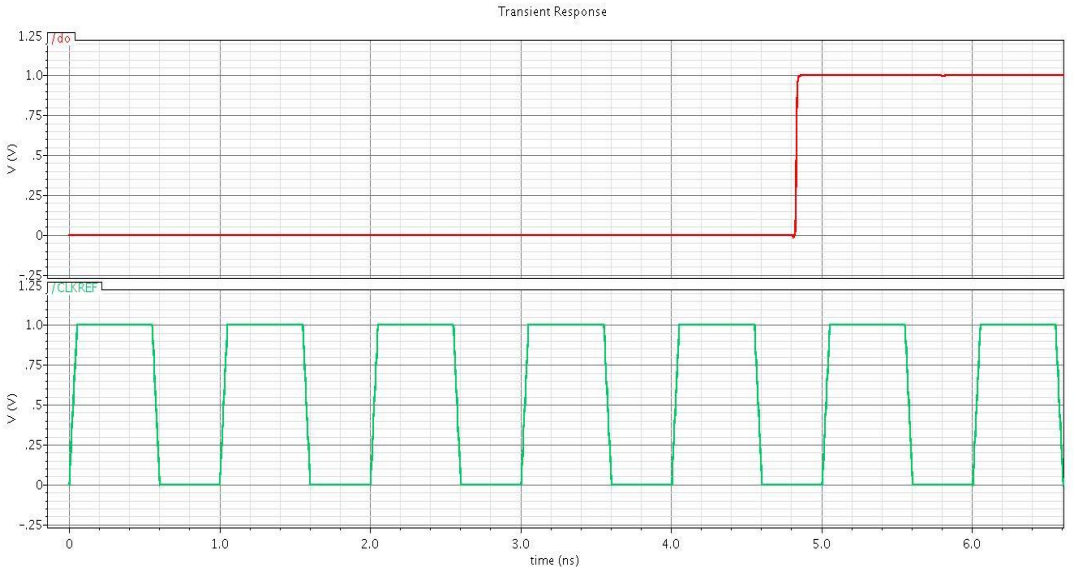


Figure 52: Timing diagram showing the maximum delay

The Pulse Generator produce a pulse after specific delay of the DCDL with width equals delay of the inverter. Its structure is as shown.

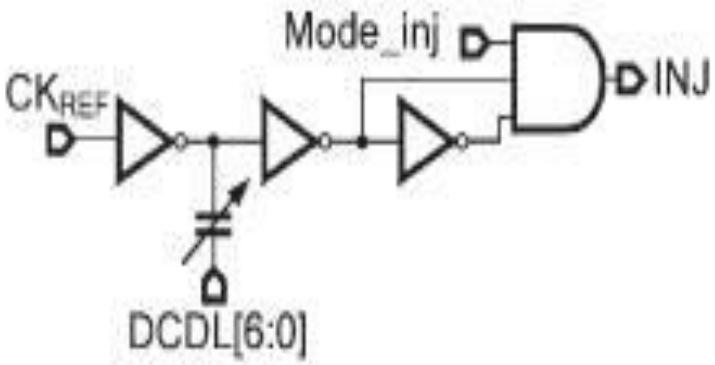


Figure 53: Pulse Generator [16]

Chapter 5: Simulation Results

The proposed ADPLL structure is designed and simulated using 65nm CMOS technology. Phase noise of the DCO is extremely reduced to -140dBc/Hz at offset frequency 1MHz, which is a remarkable achievement and greatly pushed at reaching our core target (jitter). This proposed ADPLL works at a frequency range from 415MHz to 1GHz, while the PLL tries to lock the reference frequency around 70ns. The power consumption is 80mW at 1v supply with area less than $0.1mm^2$. Our core target is 2ps peak-to-peak jitter with 0.5ps RMS jitter, which we have reached.
In this work, we moved from the upper level of VLSI (RTL) till we reached the lowest level (Layout) following full custom technique. But first for checking the functionality of our design, we made some of its blocks using Verilog and get these results.

5.1 Verilog Results

Here, we used binary to thermometer decoder and the filter only in Verilog to test the system at a reference frequency of 625MHz. And we got these results of frequency response.

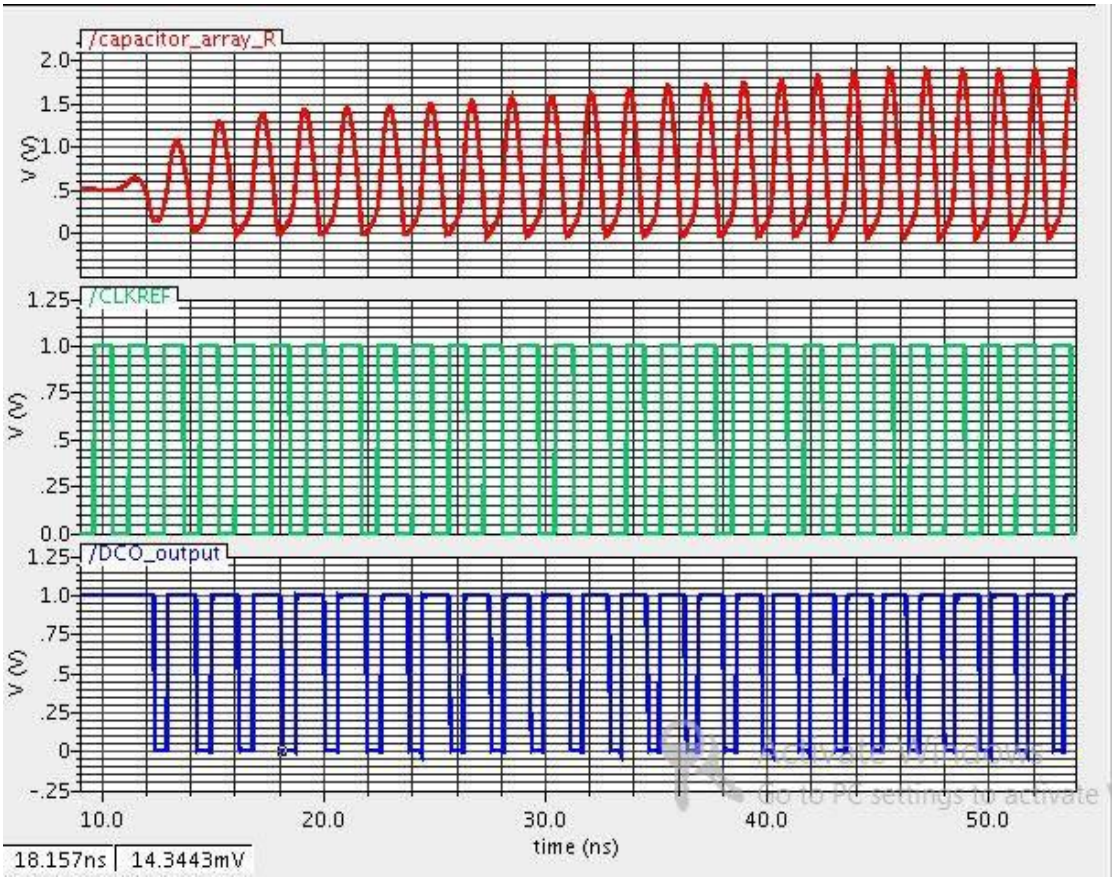


Figure 54: Frequency Plots

First plot shows the frequency generated from the LC oscillator, second plot shows the reference frequency at 625MHz. And the third plot shows the output frequency after the comparator.

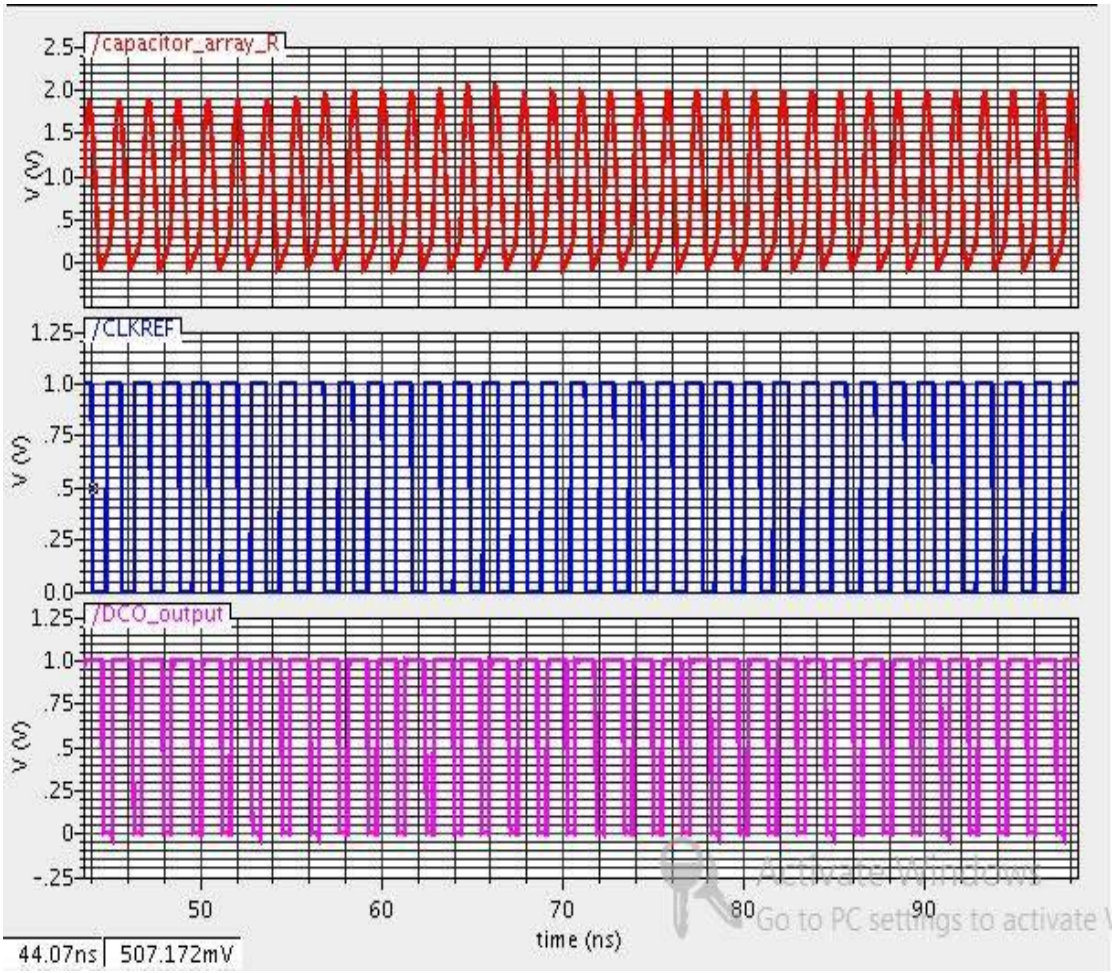


Figure 55: Same results of figure 40 but after 40ns

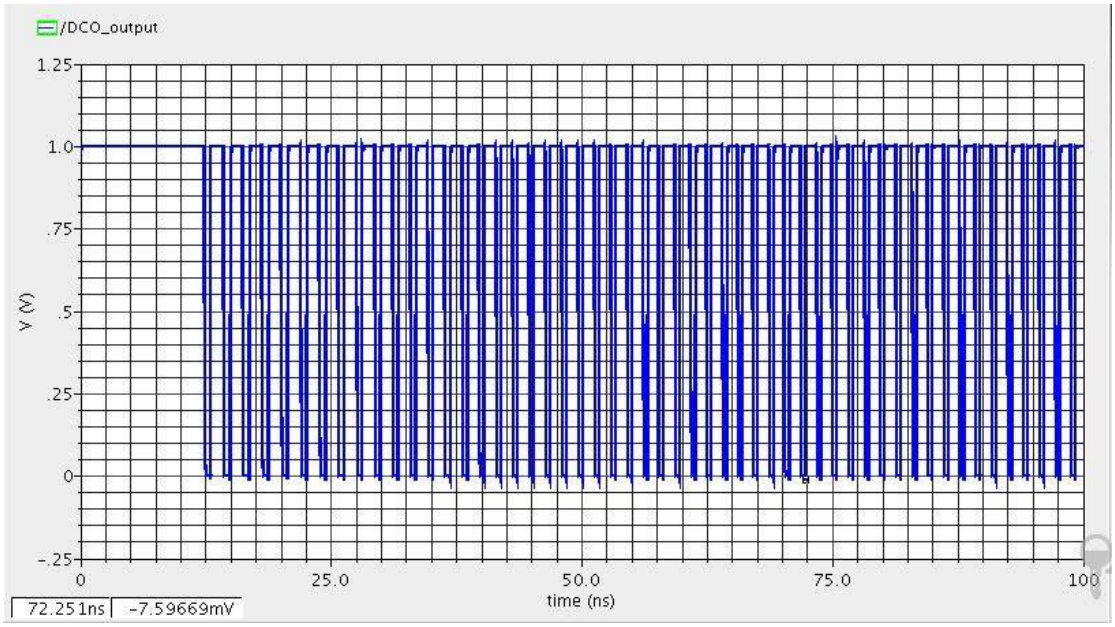


Figure 56: Output Frequency

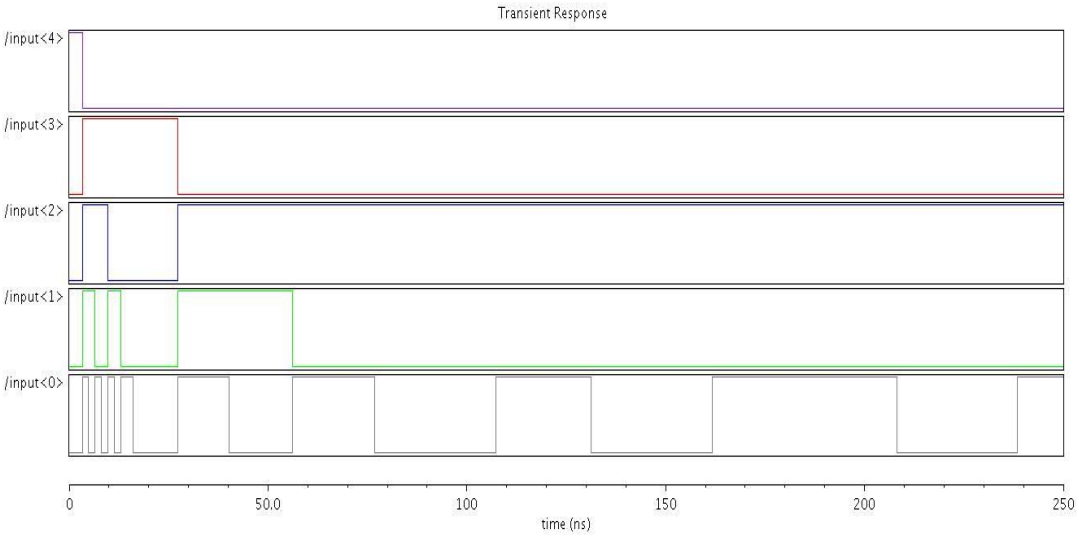


Figure 57: Filter output

The figure above, we see that the filter stopped at certain value, which is 01000. Which in frequency is around 625MHz. Therefore, the system is locking well.

5.2 Results of Full Custom System

The response of the frequency under time-domain before layout is presented in figures below.

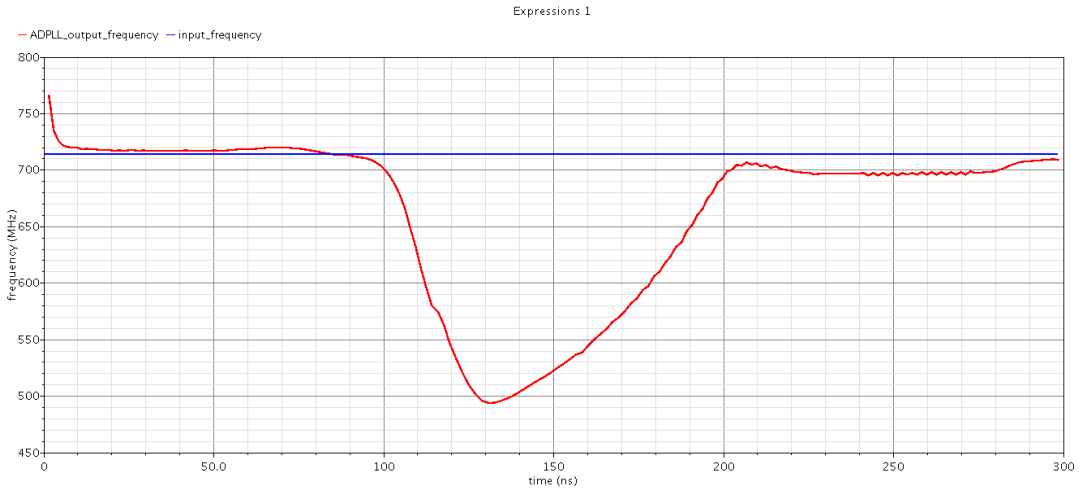


Figure 58: Locking around 715MHz

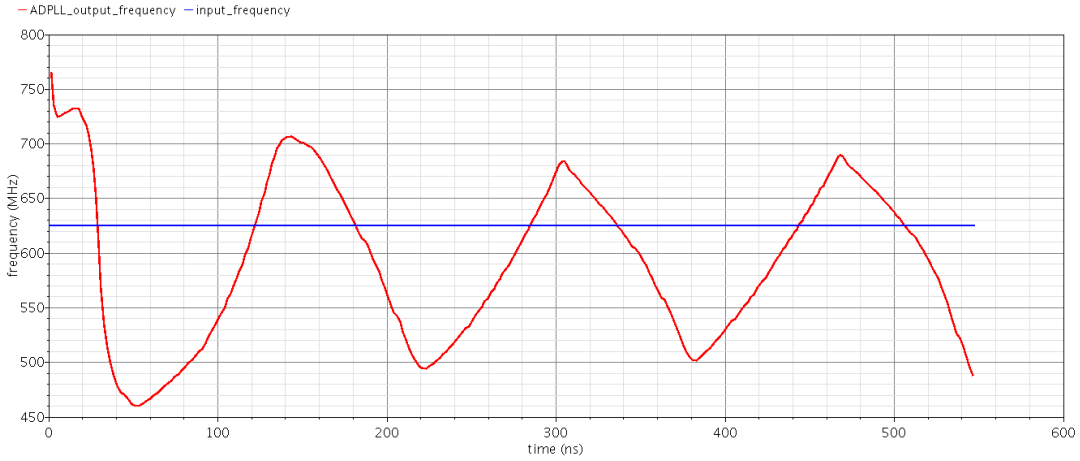


Figure 59: Locking around 625MHz

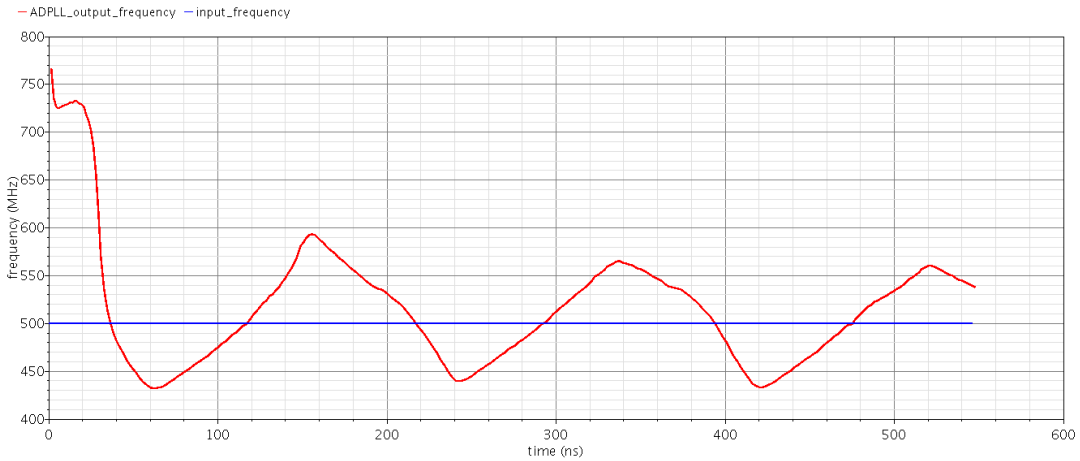


Figure 60: Locking around 500MHz

5.3 Phase Noise and Jitter in Oscillators

Phase noise and jitter are metrics that quantify the frequency stability of a periodic signal. The noise power spectrum *P* is defined by the phase noise around the fundamental frequency as shown in figure 48. In the ideal case at the fundamental output frequency *fo*, the spectrum is a Dirac-delta function. However, practically, due to device noise, the oscillator output spectrum exhibits noise power around the fundamental frequency.

Jitter metrics quantify the time domain uncertainty in the oscillator period. Ideally, edges of the oscillator signal occur at identical intervals in time as shown by the voltage waveform in Fig. 49. In practical circuits, the edges of the signal deviate in time by certain amount each cycle from this ideal position. [21]

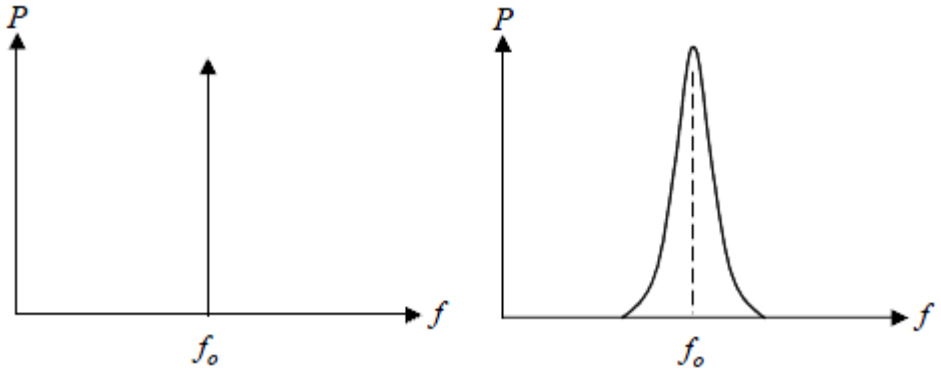


Figure 61: Left: Ideal Output Spectrum, Right: Output with phase noise [21]

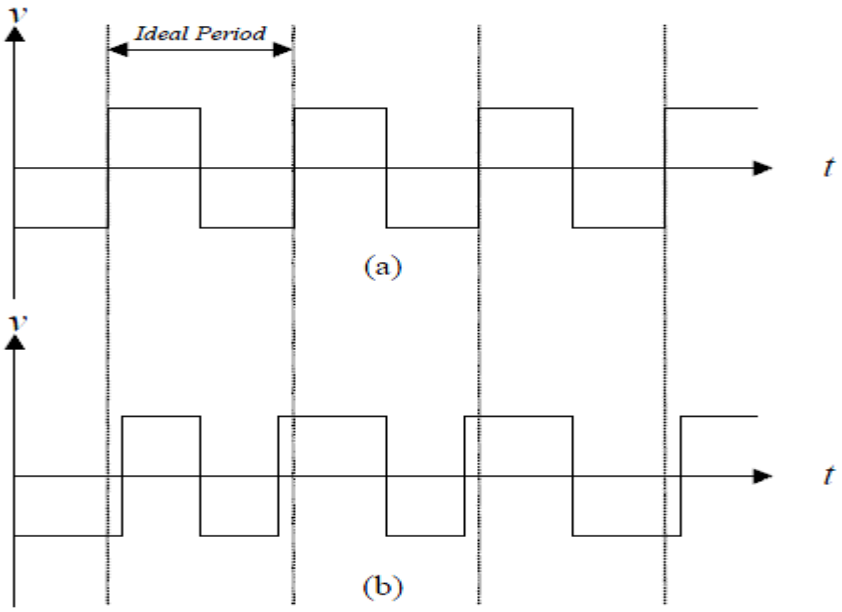


Figure 62: (a) Ideal Time domain output, (b) with jitter [21]

5.3.1 Phase Noise

Consider the ideal voltage output, *vo*(*t*), of an autonomous oscillator as a function of time, *t*. This signal can be expressed as,

$$vo(t) = Vo\cos(\omega ot) \qquad (2)$$

Where ωo is the fundamental radian frequency, and *Vo* is the nominal voltage amplitude, while the output of the same oscillator, under the influence of noise processes, can be described by,

$$vn(t) = (Vo + \varepsilon(t)) \cos(\omega ot + \varphi(t)) \quad (3)$$

Where $\varepsilon(t)$ and $\varphi(t)$ are in general, zero-mean stochastic processes. Typically, fluctuations in the amplitude of the oscillator signal are ignored as they can be eliminated with the introduction of a limiter. [21]

5.3.2 Jitter

The three most commonly employed timing jitter metrics for autonomous oscillators include long-term, period, and cycle-to-cycle jitter. All three metrics quantify the uncertainty in the oscillator period. Consider $vo(t)$ again and define the fundamental oscillation period as $T = 2\pi/\omega o$. Next, consider the noisy signal, $vn(t)$. Define the absolute instant in time of the k-th positive voltage transition of $vn(t)$ as tk and the period of the k-th cycle as Tk. The expected value of the discrete random sequence Tk is $E[Tk] = T$ and $Tk = tk+1 - tk$.

The long-term, or n-cycle, jitter is defined as,

$$J_n(k) = \sqrt{var(t_{k+n} - t_k)} \quad (4)$$

This metric measures the variation of the signal edges across $n$cycles. The period jitter, J, is simply a specialized case of the longterm jitter, where $n = 1$ as given by,

$$J_1(k) = \sqrt{var(t_{k+1} - t_k)} = \sqrt{var(T_k)} = J \quad (5)$$

and clearly J is simply the standard deviation of a single period.

Lastly, cycle-to-cycle jitter measures the variation between adjacent periods as given by, [21]

$$J_{cc}(k) = \sqrt{var(T_{k+1} - T_k)} \quad (6)$$

5.3.3 Relationship between Phase Noise and Jitter

The period jitter, J, can be expressed as,

$$J = \sqrt{aT} \quad (7)$$

Where a is the constant associated with the Lorentzian function. The single sideband phase noise spectral density can be related to the period jitter by the following expression:

$$J = \sqrt{2\frac{f^2}{f^3 o}\left(\frac{N_o}{P_o}\right)_{fm}} \quad (8)$$

This is the most important equation that links phase noise with jitter and hence we can calculate both of them. [21]

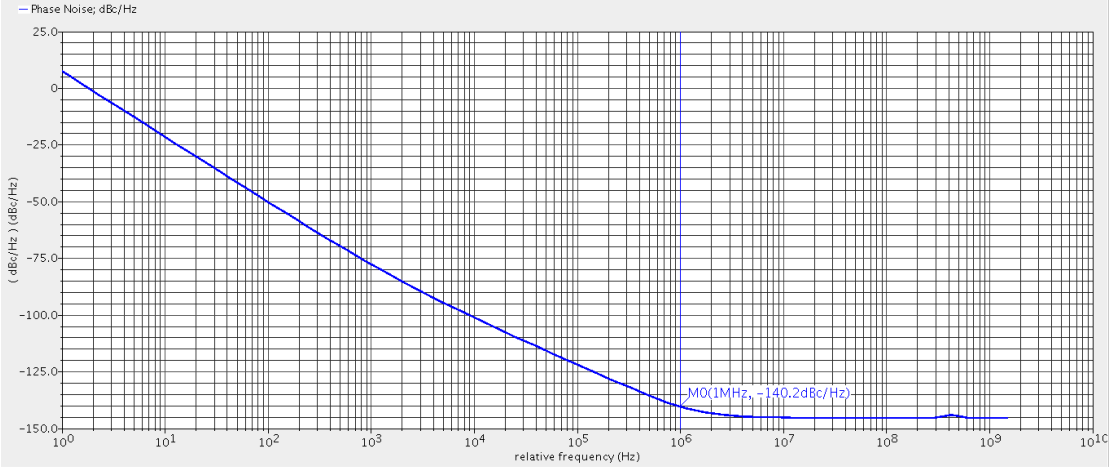


Figure 63: Phase Noise

5.4 Jitter Measurement

The jitter is measured by the method shown in the appendix-2.

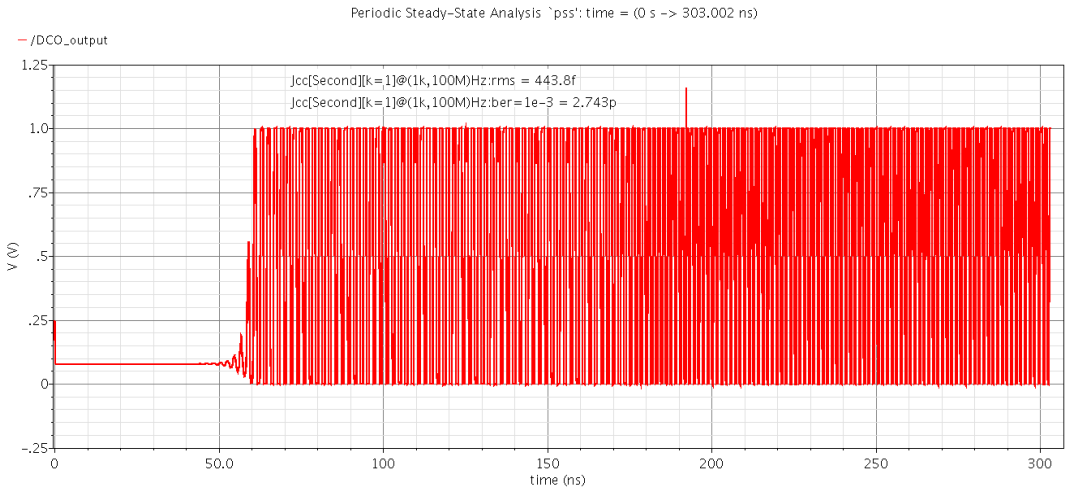


Figure 64: Output of ADPLL with jitter calculation

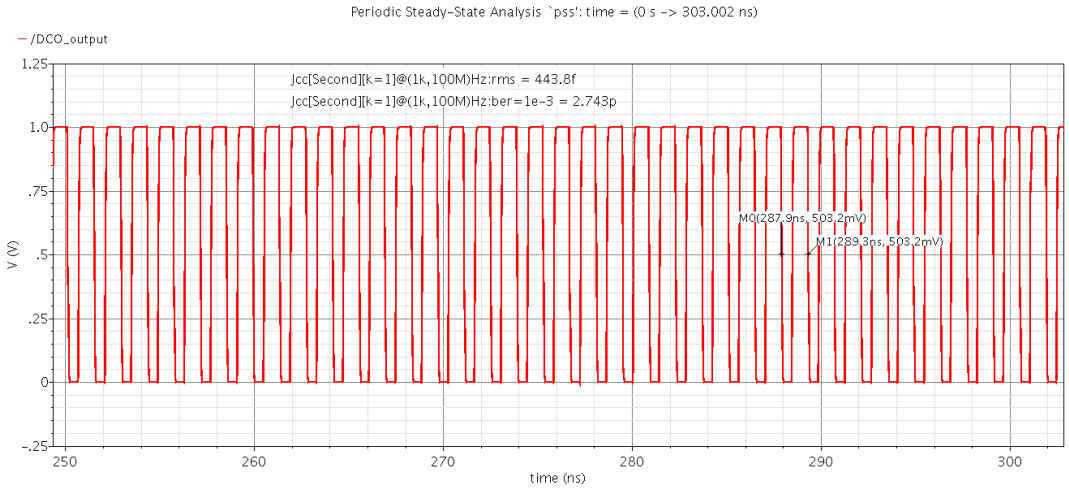


Figure 65: Zoom-in version of last figure

5.5 Measurement of ADPLL

The overall measurements for the proposed ADPLL are shown in the table below.

Table 3: Overall measurments for the propoesed ADPLL

DCO Frequency Range	415MHz to 1GHz
DCO Area	87109.0892 μm^2
LC Oscillator phase noise at 1MHz offset	-140.2dBc/Hz
RMS Jitter	0.449ps
Peak-to-peak Jitter	2.7ps
Power consumption	80mW

5.6 Post Layout Simulation

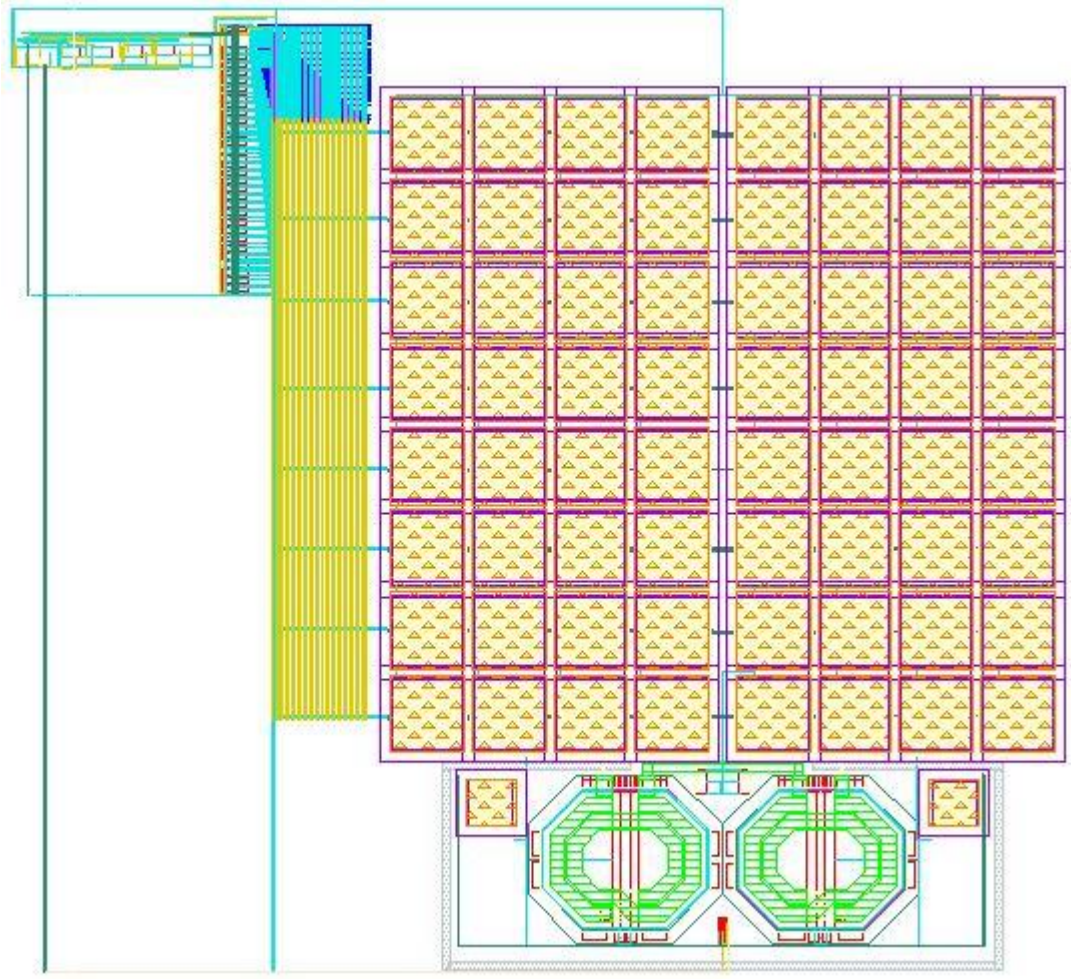


Figure 66: Layout of the whole system

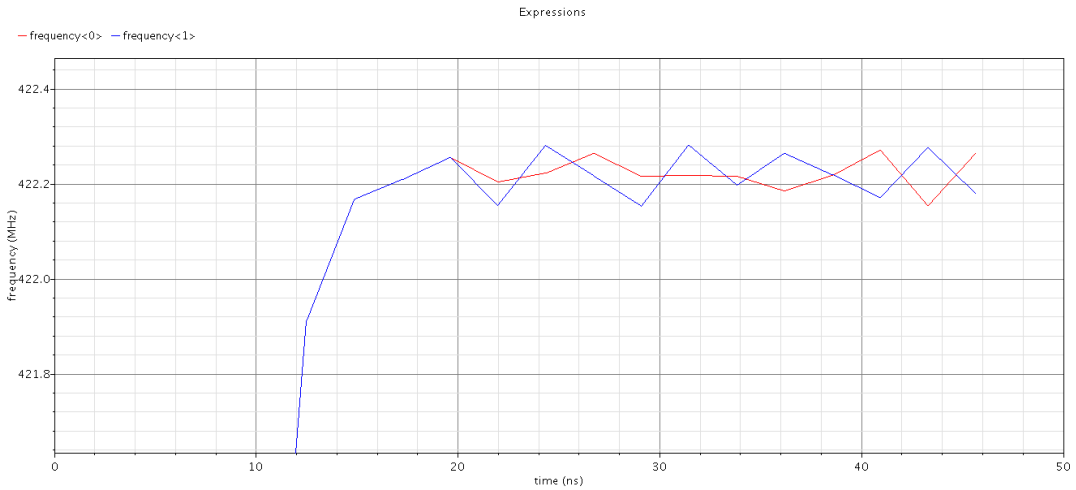


Figure 67: DCO output frequency before and after layout at 422.2MHz reference frequency

First figure shows the layout of the whole system, and the second one shows the output frequency of the DCO before and after layout when we entered a reference frequency of 422.2MHz. These results show us the parastics did not affect the locking much; because we improved them.

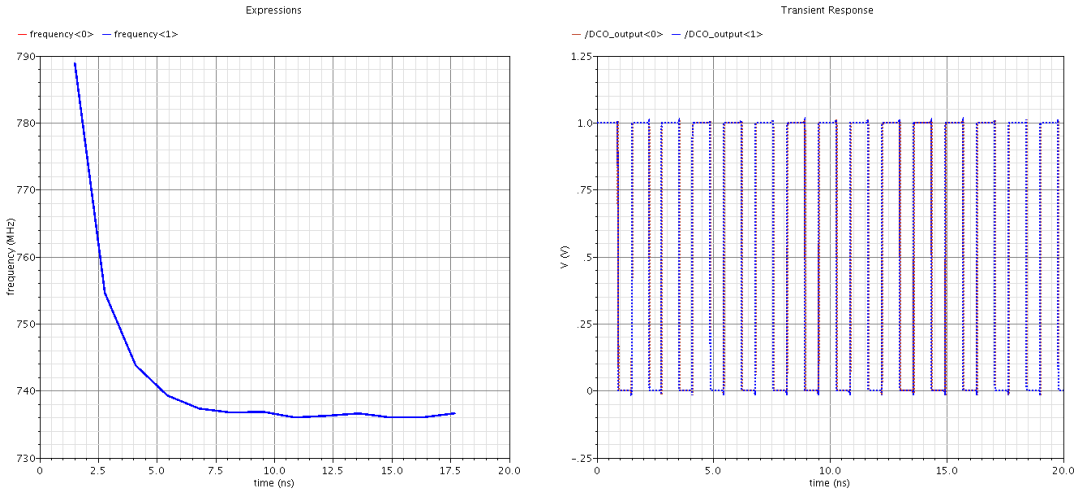


Figure 69: Left: Output frequency at 736MHz, Right: Output clock

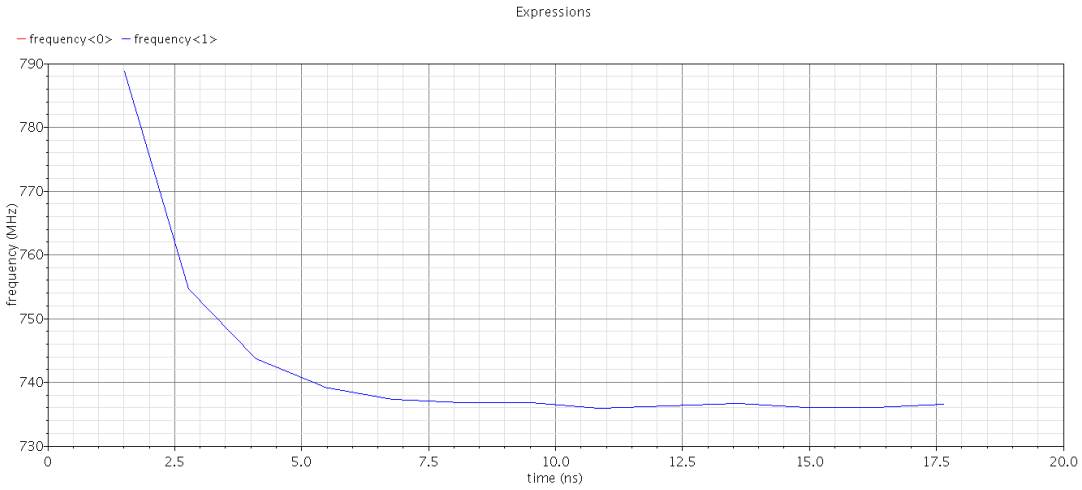


Figure 68: Zoom-in version of the ouptut frequency at 736MHz

Here, we changed the reference frequency to 736MHz and simulated the layout to get these results which shows us that the system is locking at 736MHz.

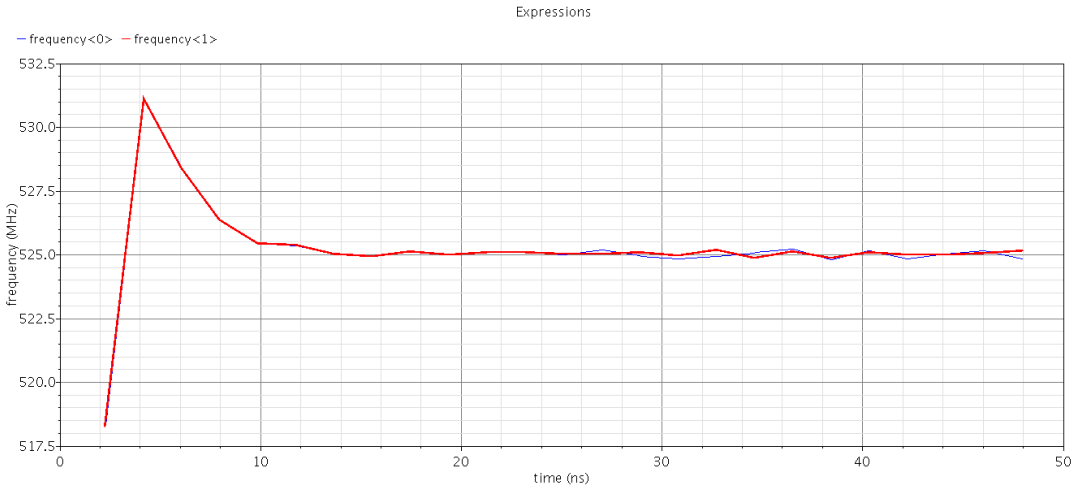


Figure 70: DCO output frequency before and after layout at 525MHz reference frequency

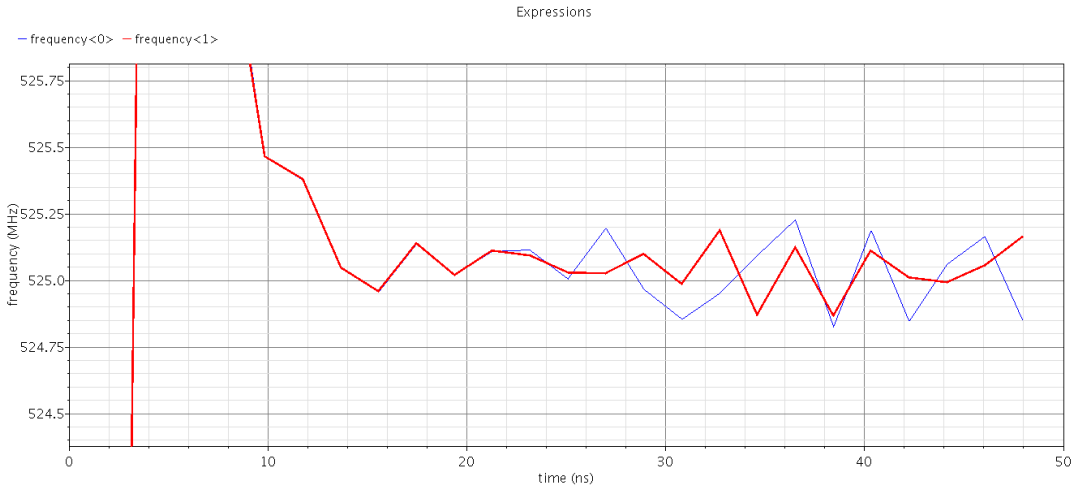


Figure 71: Zoom-in of last figure

After these results, we conclude that the system is working well before and after layout and the parasitics did not cause the system to be failed.

Chapter 6: Conclusion

All Digital Phase-Locked Loop, ADPLL, is presented here all over this work with different structures of the blocks consisting it. In the shown table, a simple comparison between analog PLL and digital one to show why we are moving nowadays to digital PLLs.

Table 4: Comparison between AAPLL and ADPLL

	AAPLL	ADPLL
Area	Large	Small
Power	High	Low
Jitter	Better	Worse
Ability of change in frequency	Continuous	Discrete
Scalability	Hard	Easy
Lock Time	Small	Large

The table shows that ADPLL is better than AAPLL in many fields except jitter, which is a big problem. Therefore, this work focuses on the jitter as a main core, with extremely reduced DCO phase noise and small area. The proposed ADPLL is implemented on 65nm CMOS technology with a 1V operating voltage operating from 415MHz to 1GHz. The lock time of the PLL is almost 70nsec and the total power is around 80mW with total area less than $0.1mm^2$. ADPLL has a peak-to-peak jitter of 2.7ps at 400MHz, and a 0.449ps RMS jitter.

In this work, we followed the Gajski chart, as followed.

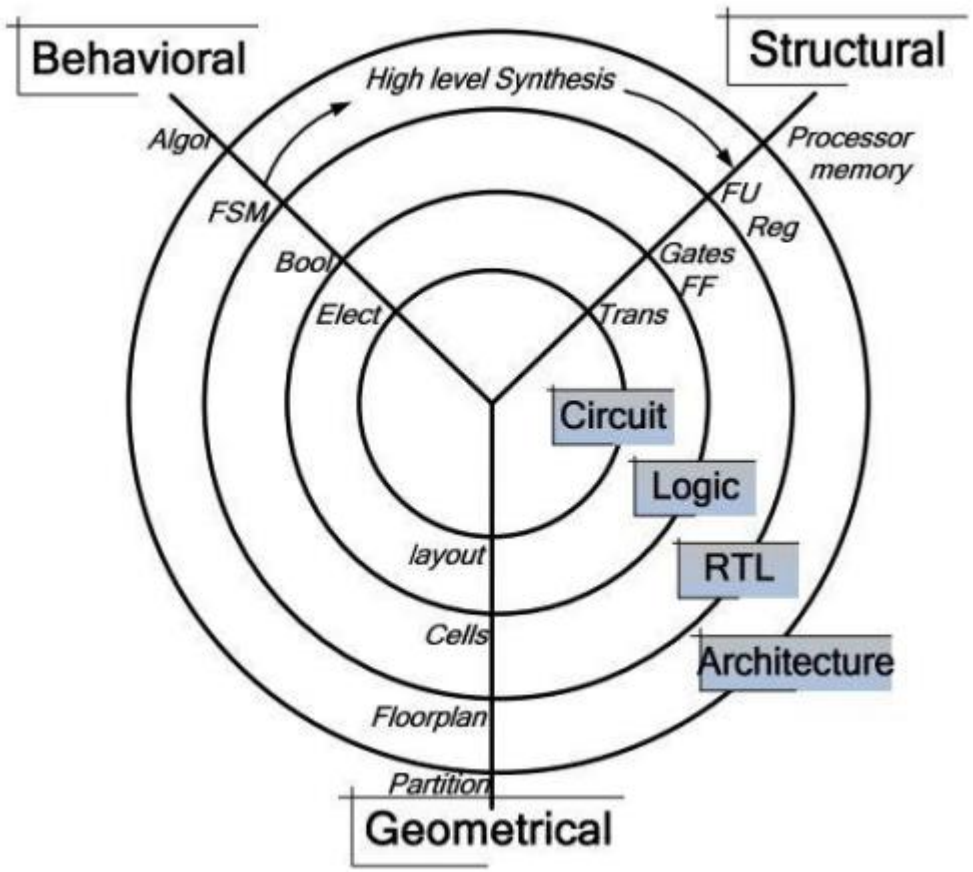


Figure 72: Gajski Chart

The proposed ADPLL can be modified as a future work to minimize the power and hence the system is enhanced.

References

[1] M. S. Kim, 0.18μm CMOS Low Power ADPLL with a Novel Local Passive Interpolation Time-to-Digital Converter Based on Tri-State Inverter, July 2012.

[2] A. Reda, H. Mohamed, K. Ebrahim, K. Mohamed, M. Sherif, All Digital Phase Locked Loop (ADPLL), July 2013.

[3] C.C. Chung and C.Y. Lee, An all-digital phase-locked loop for high-speed clock generation, Solid-State Circuits, IEEE Journal of, Vol. 38(No.2):Pages 347-351, Feb. 2003.

[4] C. Emmerich, Introduction to Jitter, October.

[5] H. Johnson, Random and Deterministic Jitter, EDN Magazine, June 2002. Also available at: https://www.sigcon.com/Pubs/edn/RandomJitter.htm

[6] Ana Armendáriz, Analysis and Design of PFDs in Cadence, 2010.

[7] B. Razavi, editor, Phase-Locking in High Performance Systems IEEE Press, 2003.

[8] S. C. LEE, Digital Circuits and Logic Design.

[9] Differences between FIR and IIR Filters. Also available at: http://iowahills.com/A8FirIirDifferences.html

[10] M. Hsieh, G. E. Sobelman, "Comparison of LC and Ring VCOs for PLLs in a 90 nm Digital CMOS Process".

[11] D. Tasca, M. Zanuso, G. Marzin, S. Levantino, C. Samori, A. L. Lacaita, "A 2.9-to-4.0GHz Fractional-N Digital PLL with Bang-Bang Phase Detector and 560fsrms Integrated Jitter at 4.5mW Power", ISSCC 2011 / SESSION 5 / PLLs / 5.1.

[12] X. Gao, E. Klumperink, G. Socci, M. Bohsali, B. Nauta, "A 2.2GHz Sub-Sampling PLL with 0.16psrms Jitter and -125dBc/Hz In-band Phase Noise at 700μW Loop-Components Power", 2010.

[13] J. R. Hu, R. C. Rubyy, B. P. Otis, "A 1.5GHz 0.2psRMS Jitter 1.5mW Divider-less FBAR ADPLL in 65nm CMOS", 2012.

[14] Shahrzad Naraghi, Time-Based Analog To Digital Converters, 2009.

[15] J. Marteau, J. d'Ars, D. Gibert, K. Jourde, S. Gardien, C. Girerd, J. C. Ianigro, “Implementation of sub-nanosecond time-to-digital convertor in field-programmable gate array: applications to time-of-flight analysis in muon radiography”, February, 2014.

[16] I-Ting Lee, K. H. Zeng, S. I. Liu, “A 4.8-GHz Dividerless Subharmonically Injection-Locked All-Digital PLL With a FOM of −252.5 dB”, September, 2013.

[17] R. Nandini, H. S. Raghav, B. P. Singh, “Design of Low Power Phase Frequency Detectors for Delay Locked Loop”, International Journal of Emerging Science and Engineering (IJESE), March, 2013.

[18] Rabaey, Digital Integrated Circuits A Design Perspective (2nd Ed).

[19] L. C. Wei, A. Long, Synchronous Counters-Final Report. Also available at: https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cwl3/report.html

[20] Omar Haridy, Synthesizable Delay Line Architectures For Digitally Controlled Voltage Regulators, July, 2012.

[21] M. S. McCorquodale, M. K. Ding, R. B. Brown, Study and Simulation of CMOS LC Oscillator Phase Noise and Jitter, 2003.

Appendix

1- Post Layout Simulations

First, we need to know, what is the meaning of post layout simulation?

It means that I have achieved required specifications on schematic level, so we need to make the layout of this circuit and run some physical verification tests to be sure that my design will approximately work after fabrication.

These physical tests are:

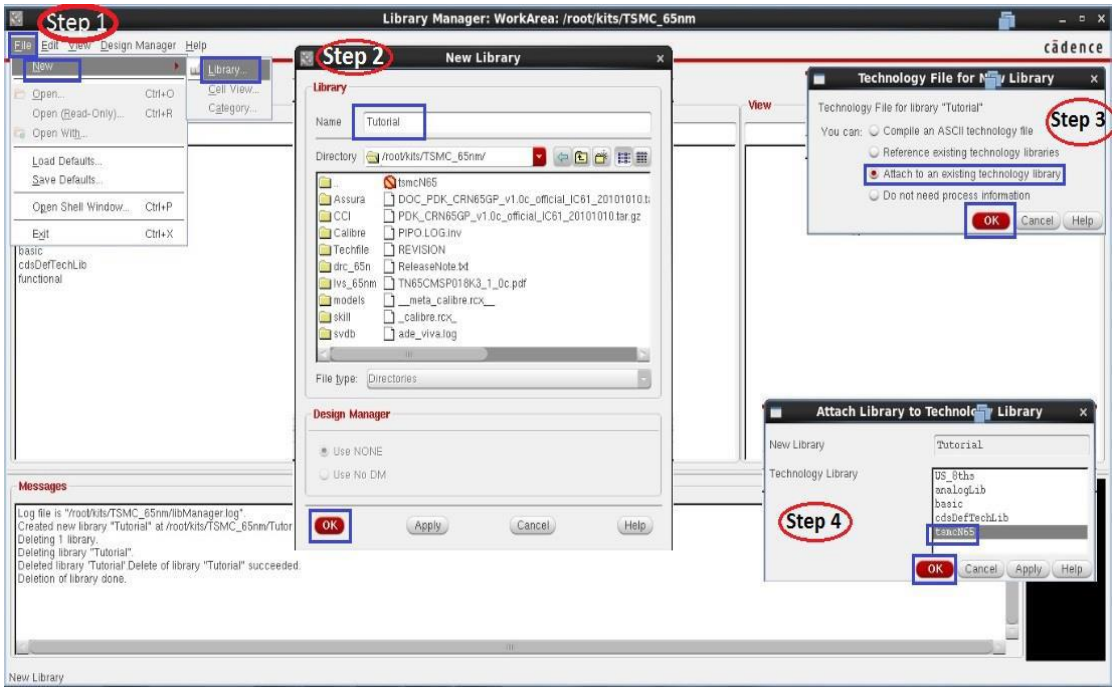
DRC: Design Rule Check on Layout

LVS: Layout vs. Schematic, to check the connections

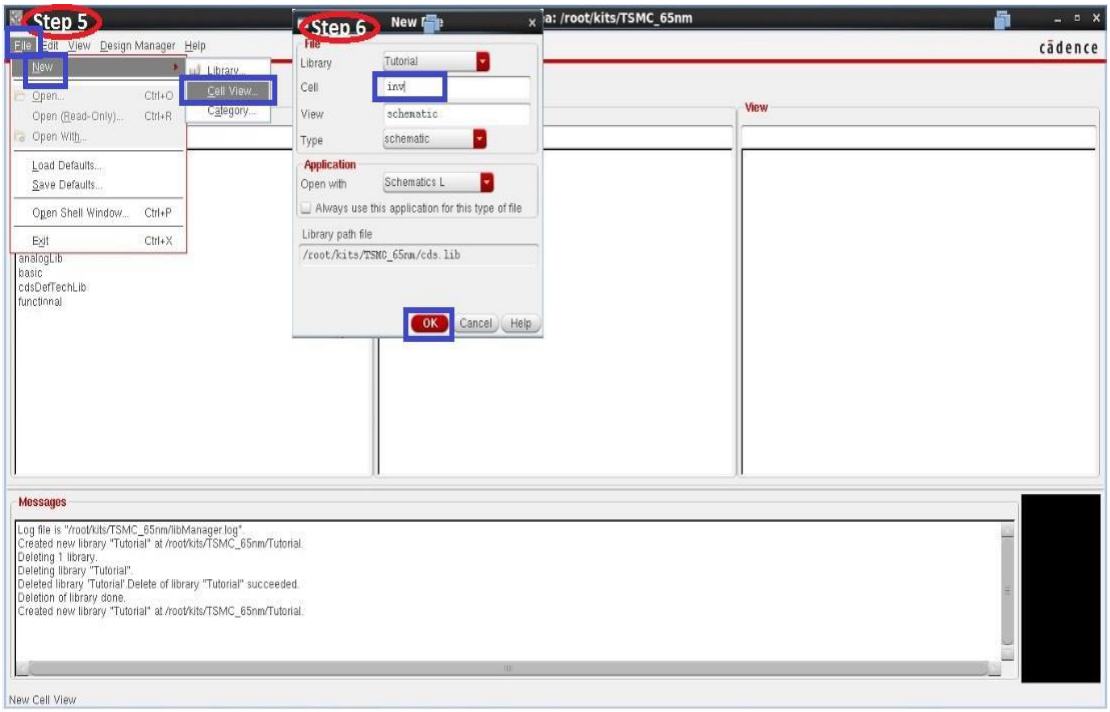
PEX: Parasitic Extraction for Resistor + Capacitors to get approximate result from real life.

Now, Let us begin with steps towards tape-out of an IC:

1- In order to create new library in cadence 6.1x:



2- In order to create new cell view to draw your own schematic:



3- Begin by drawing schematic:

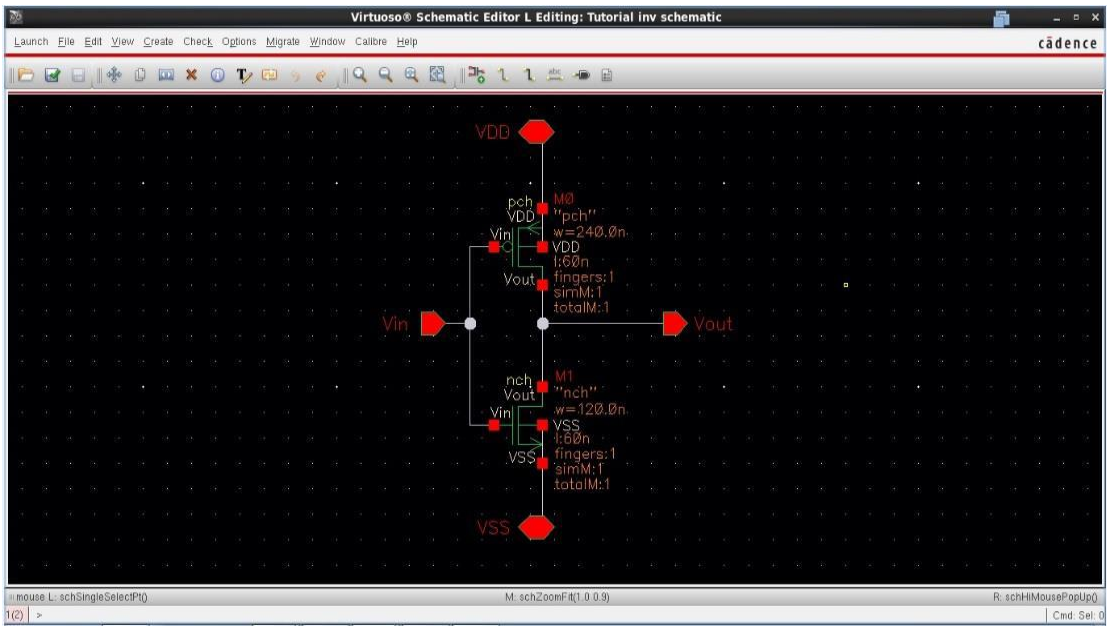
Inserting components → press ‘i’

Change component parameter → press ‘q’

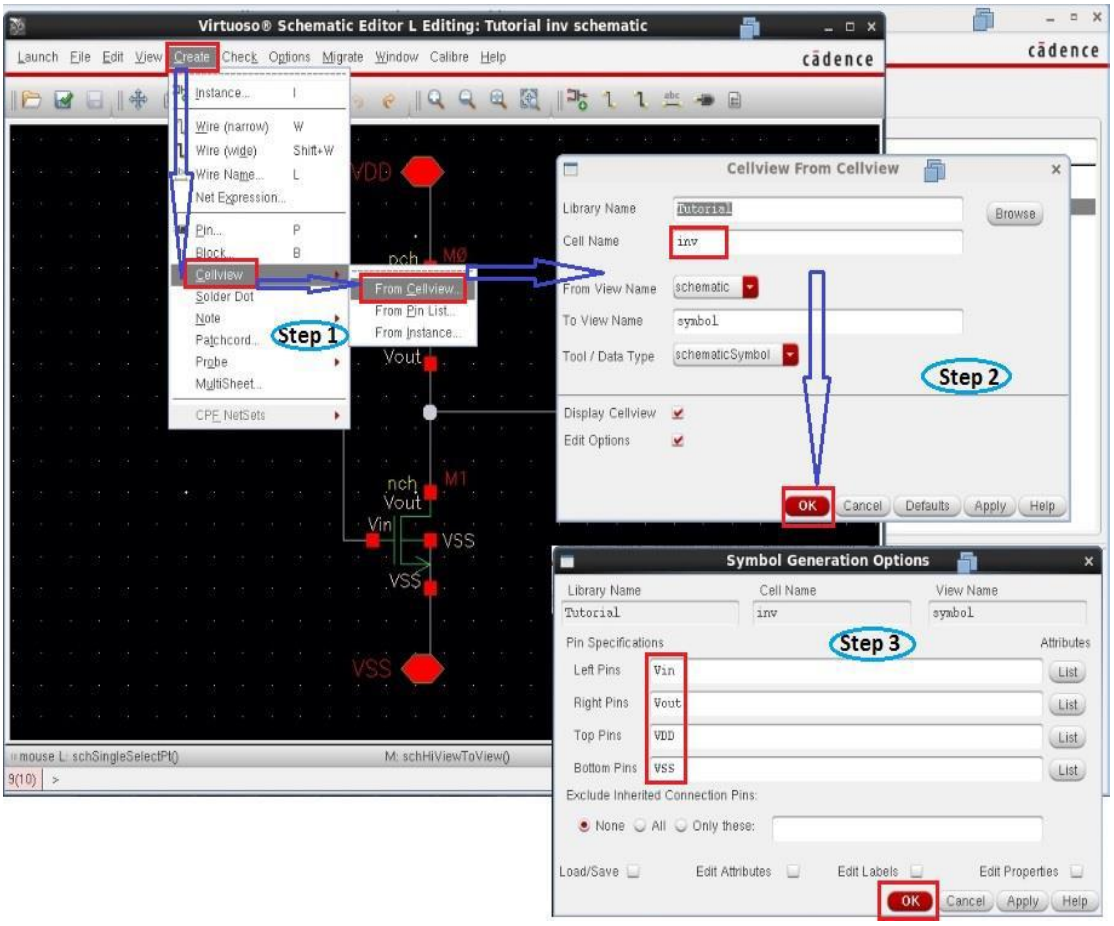
For wiring between different components → press ‘w’

For making I/O pins → press ‘p’

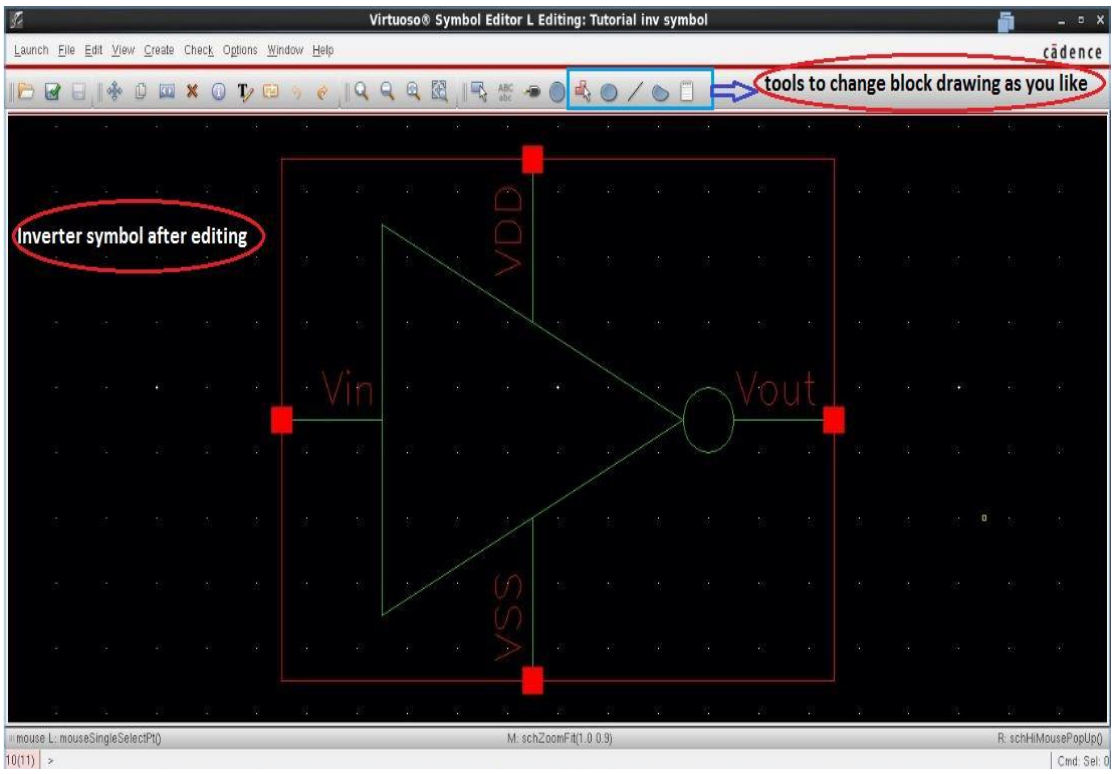
Here is an inverter shown as example:



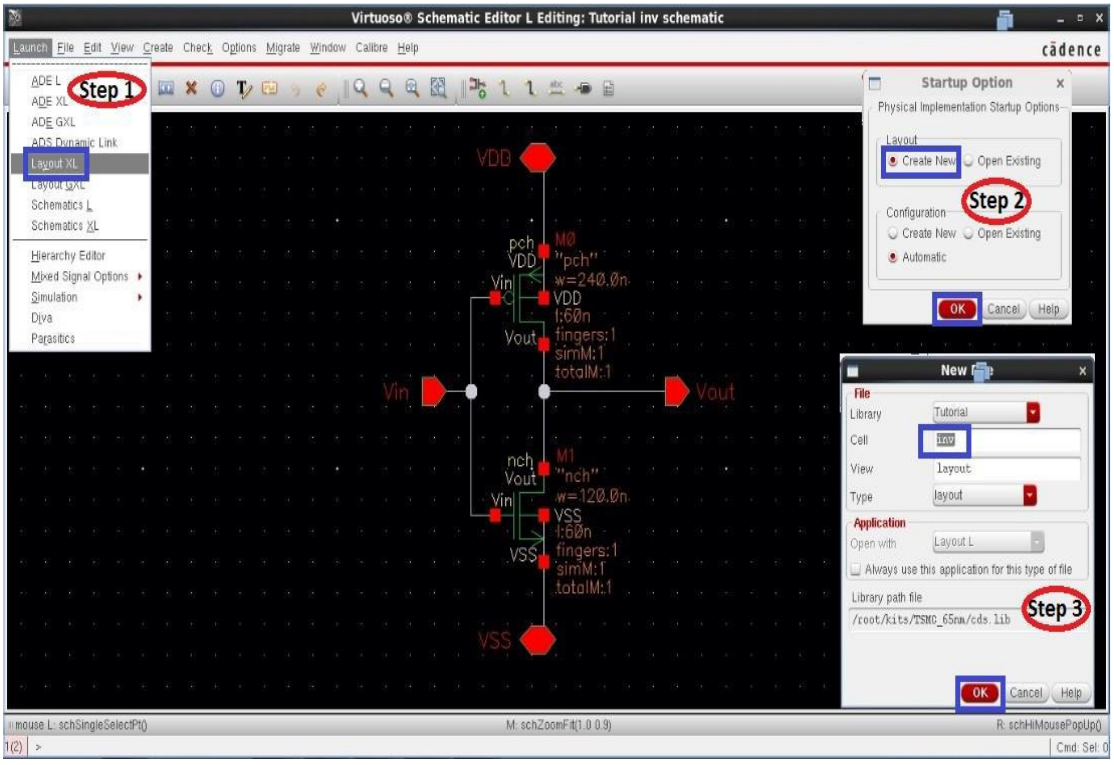
4- Create symbol from this schematic:



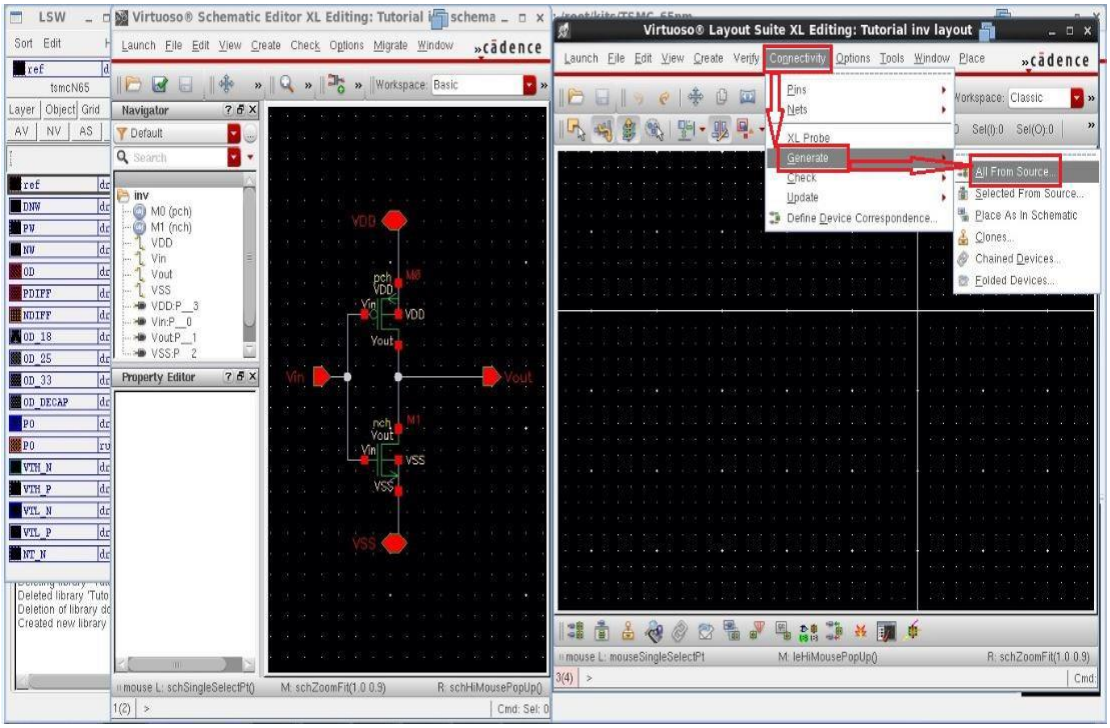
5- Change symbol block as you like as shown:



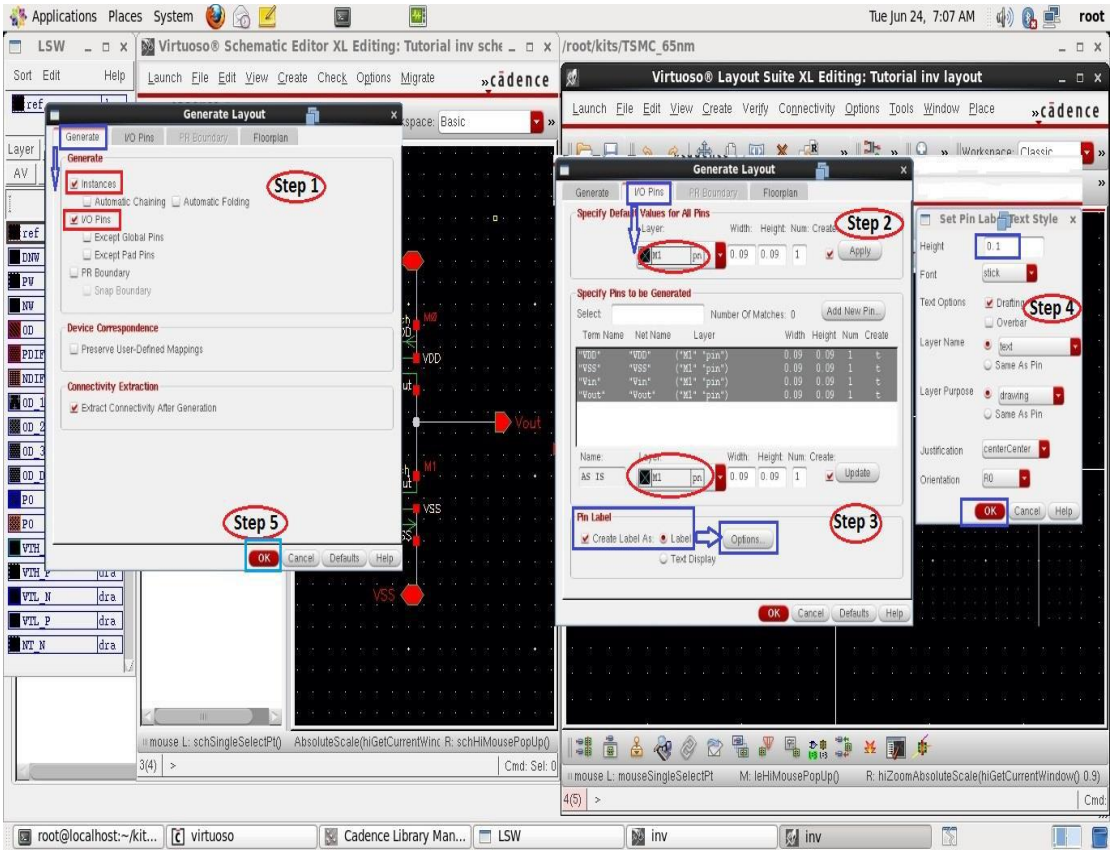
6- After creating symbol to use inverter as building block like any other component, we need to make layout for this schematic as shown:



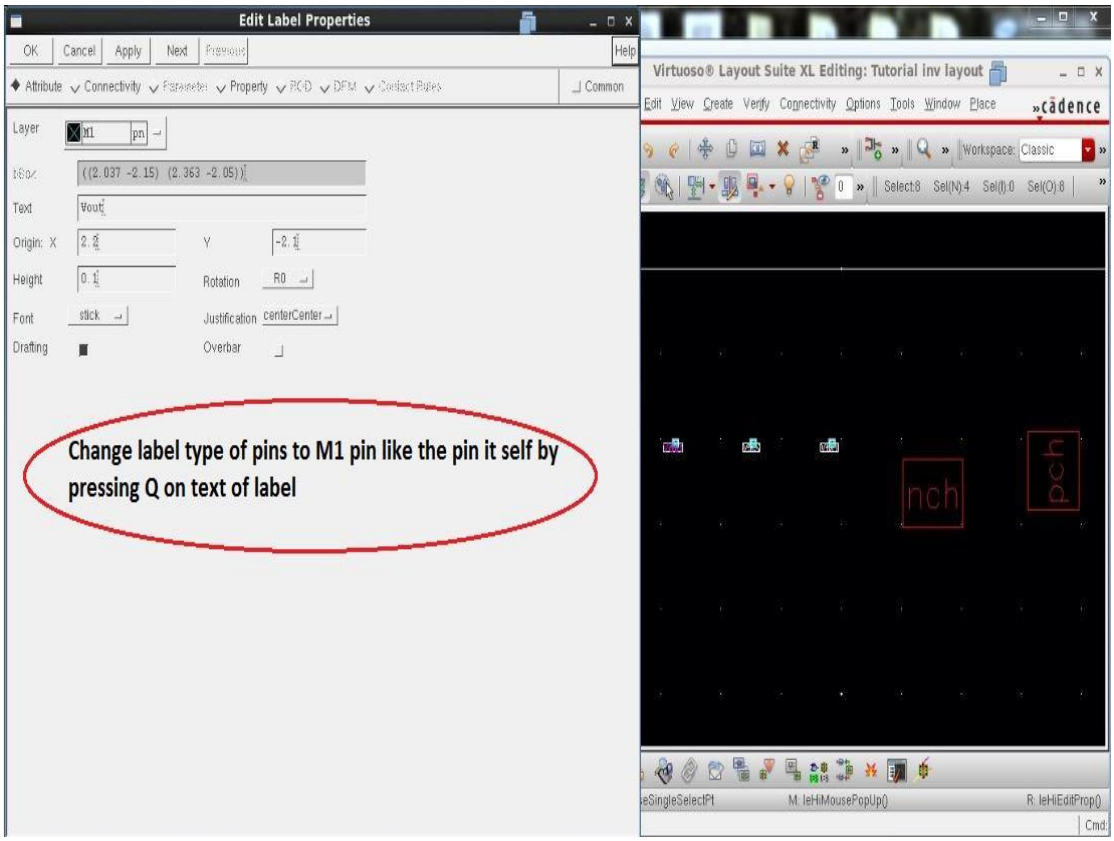
7- After, Layout extractor appear as shown we need to generate components like next step:



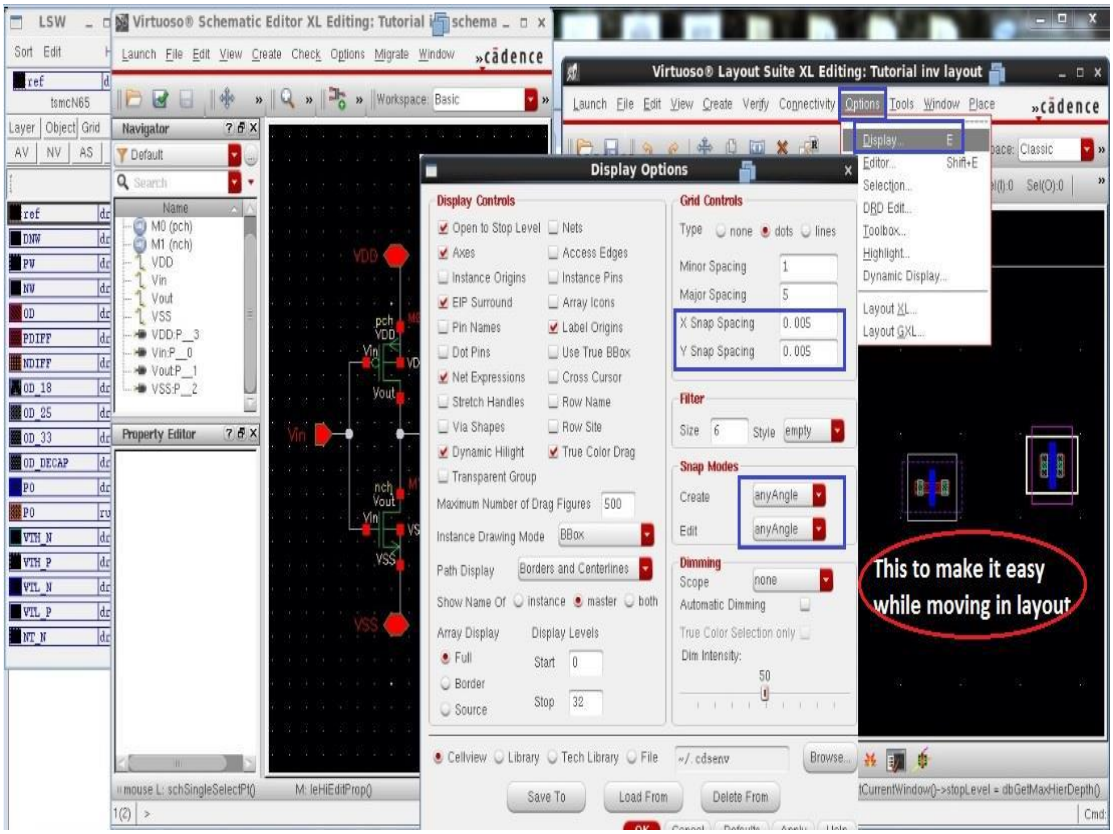
8- Now, We will generate our transistors and I/O pins



9- Label of I/O pins still need to be modified as shown:

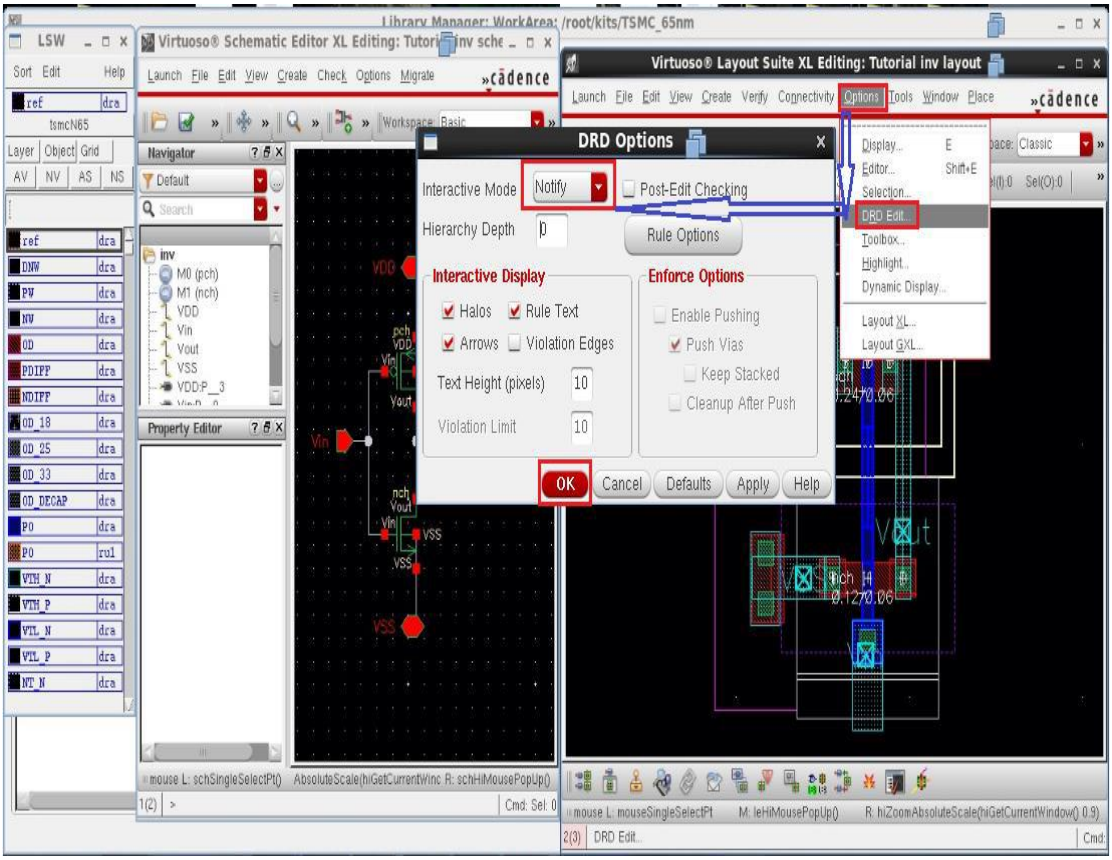


10- Transistor internal structure is hidden, in order to make it appear press “shift + f”. You will find some difficulty while moving components, so you need to edit display options as shown:



11- To make interconnections between transistors with each other and with I/O pins, we use following steps:

- To make a path from node to another, check that you select the required metal layer and beside it written that is used for drawing “dra”, then press ‘p’ and make your path.
- To zoom on components press “ctrl + z”.
- To stretch edge of path, just press ‘s’ followed by one left click, then start to move.
- For ruler to check spacing press ‘k’, after finishing press “shift + k”.
- To be notified in case of DRC violation takes place, activate this option as shown:



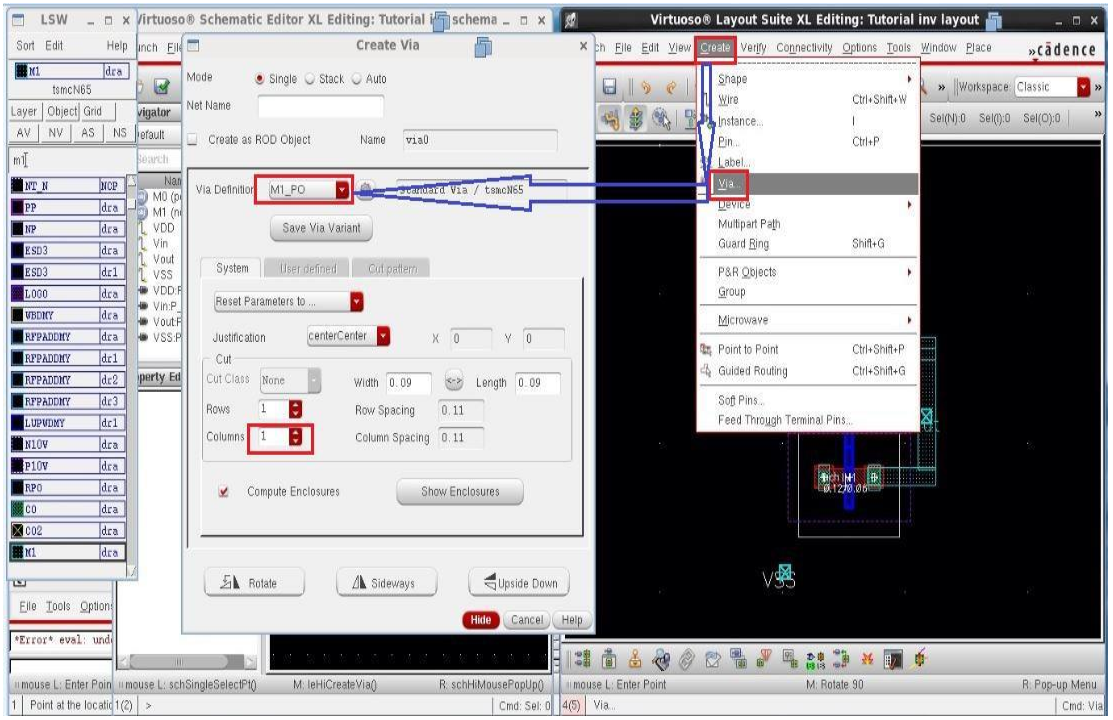
- To make a Via, follow shown figure; as we need to make it in these cases:

Between to different metal layers.

Between Metal 1 and Substrate: to create bulk terminal of NMOS transistor.

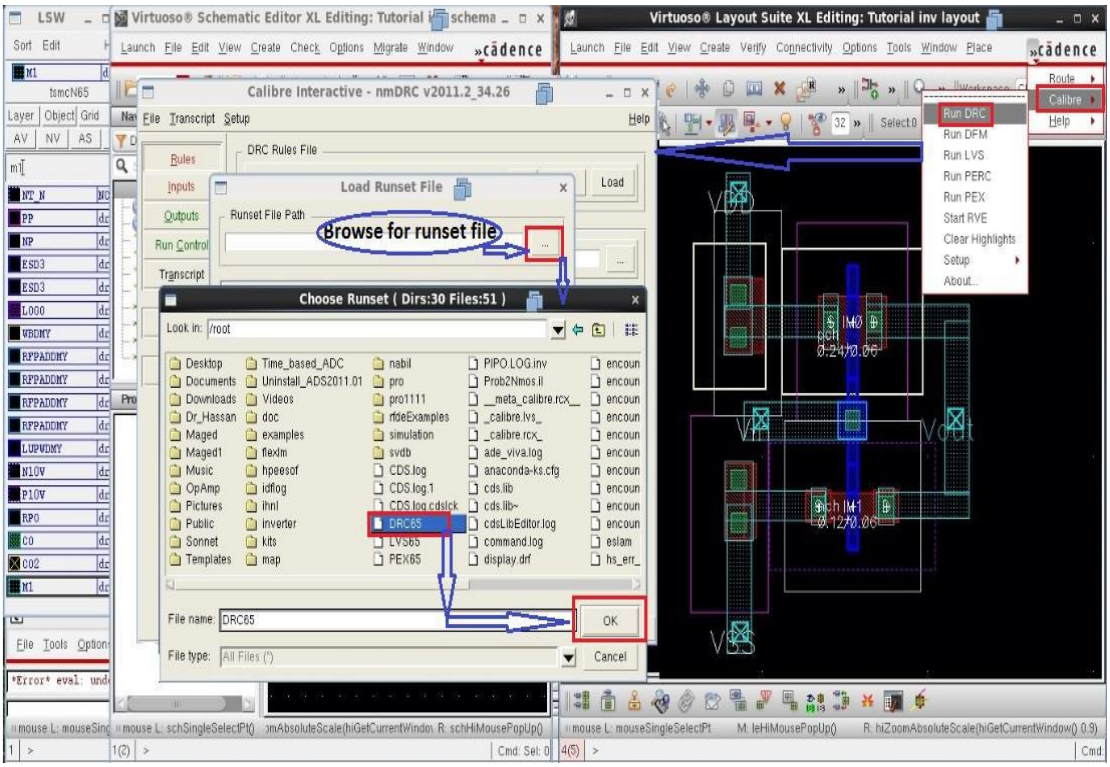
Between Metal 1 and N-Well: to create bulk terminal of PMOs transistor.

Between Metal layer and poly.



12- Now, after showing how to draw layout, we need to run different checks we mentioned before:

- First, DRC is done as shown figure:



Before we continue the other checks, we show here common errors that would appear with DRC:

Errors that can be neglected like the following:

Anything contain “Density” or “CSR”.

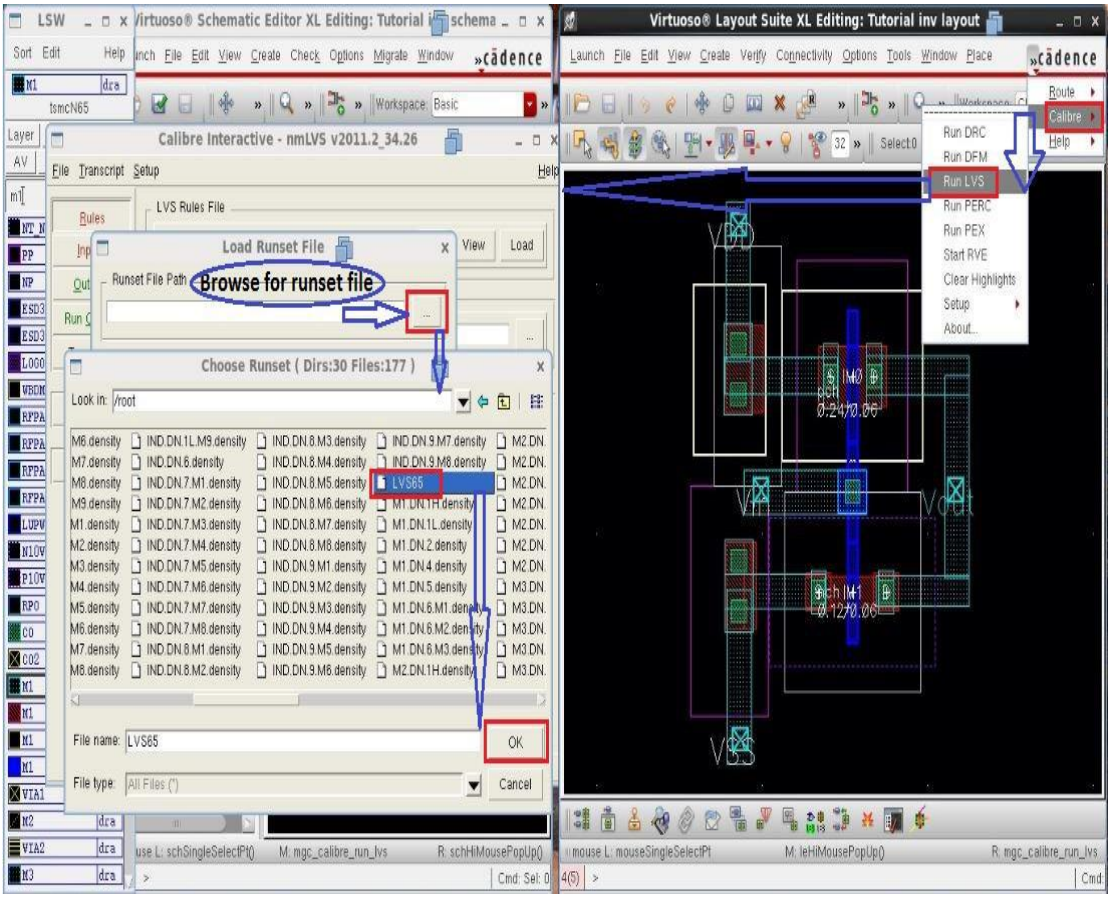
```
OD.DN.1L { @ (OD OR DOD) density across full chip >= 25%
  DENSITY ALL_OD CHIP < OD_DN_1L INSIDE OF LAYER CHIPx PRINT OD.DN.1L.density
  [ AREA(ALL_OD)/AREA(CHIP) ]
}
```

```
CSR.R.1.NWi { @ NWi is not allowed inside the empty area of chip corner.
  EMPTY_AREA AND NWi
}
```

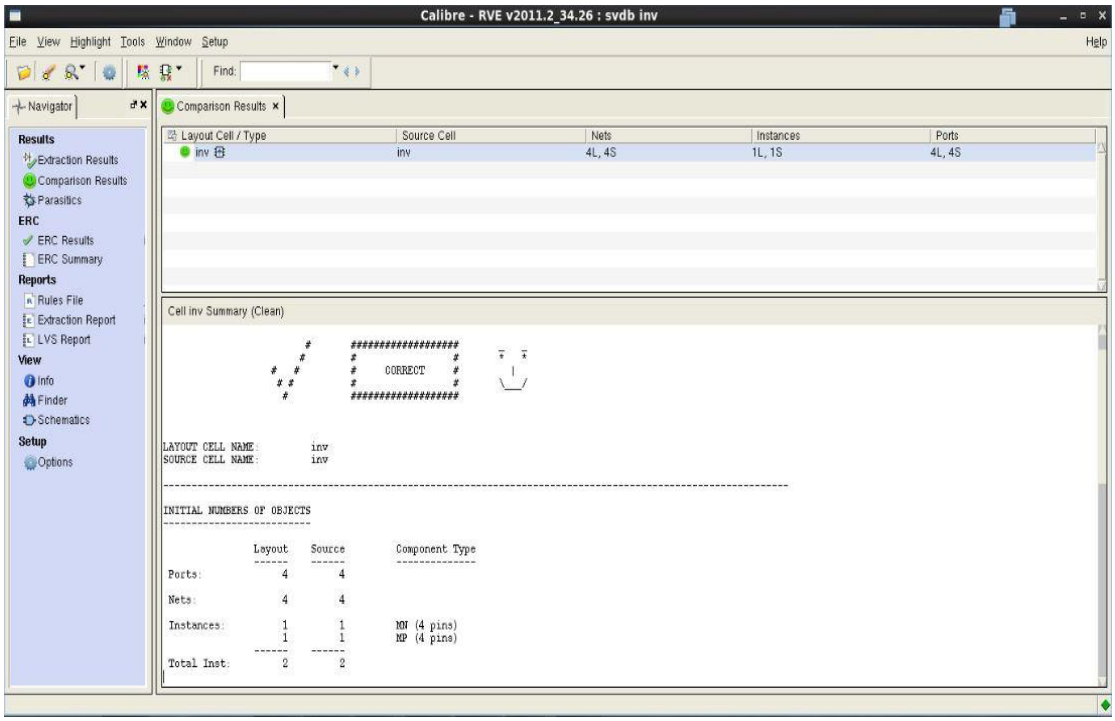
Errors that are related to dimensions like following cannot be neglected. Just close layout view and press on error, layout viewer will open automatically highlighting on error to be resolved.

```
NW.A.3 { @ Area (one of edge length < 0.8 um) >= 1 um2
  X = LENGTH NWEL < NW_A_3_L
  Y = NWEL WITH EDGE X
  AREA Y < NW_A_3
}
```

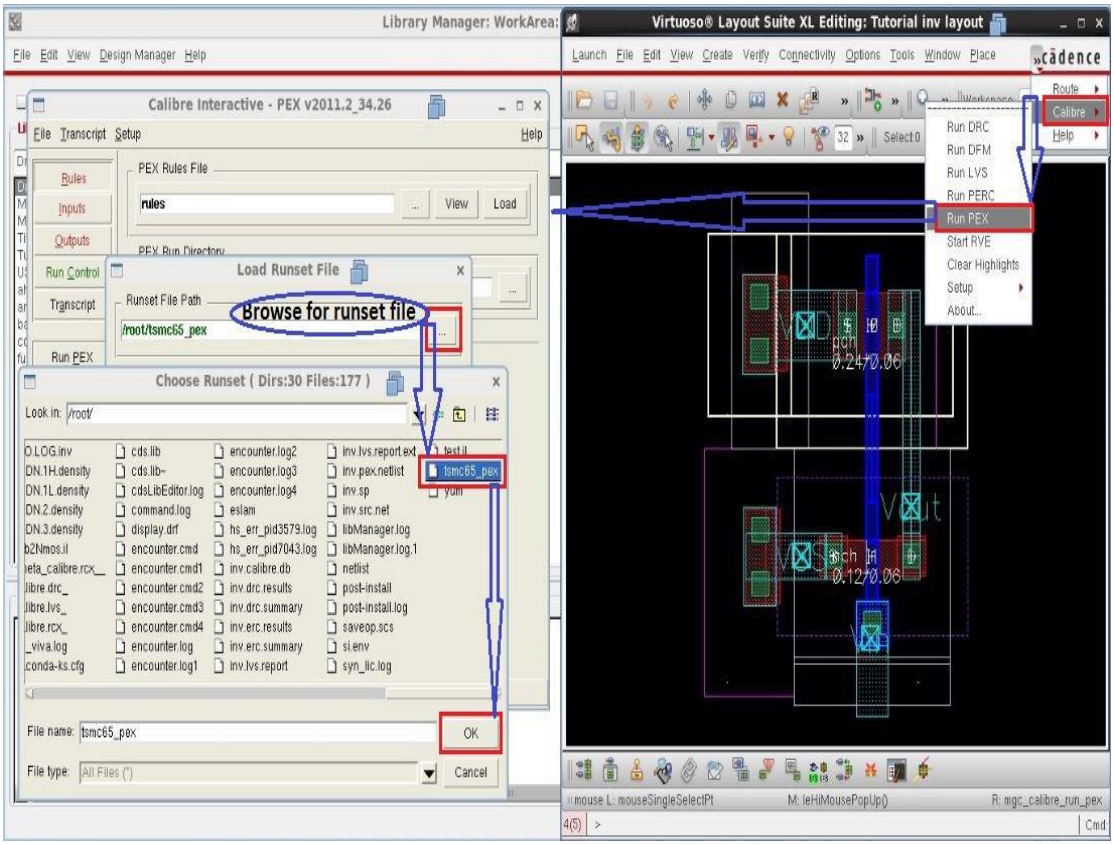
- Second, LVS is done as shown figure:



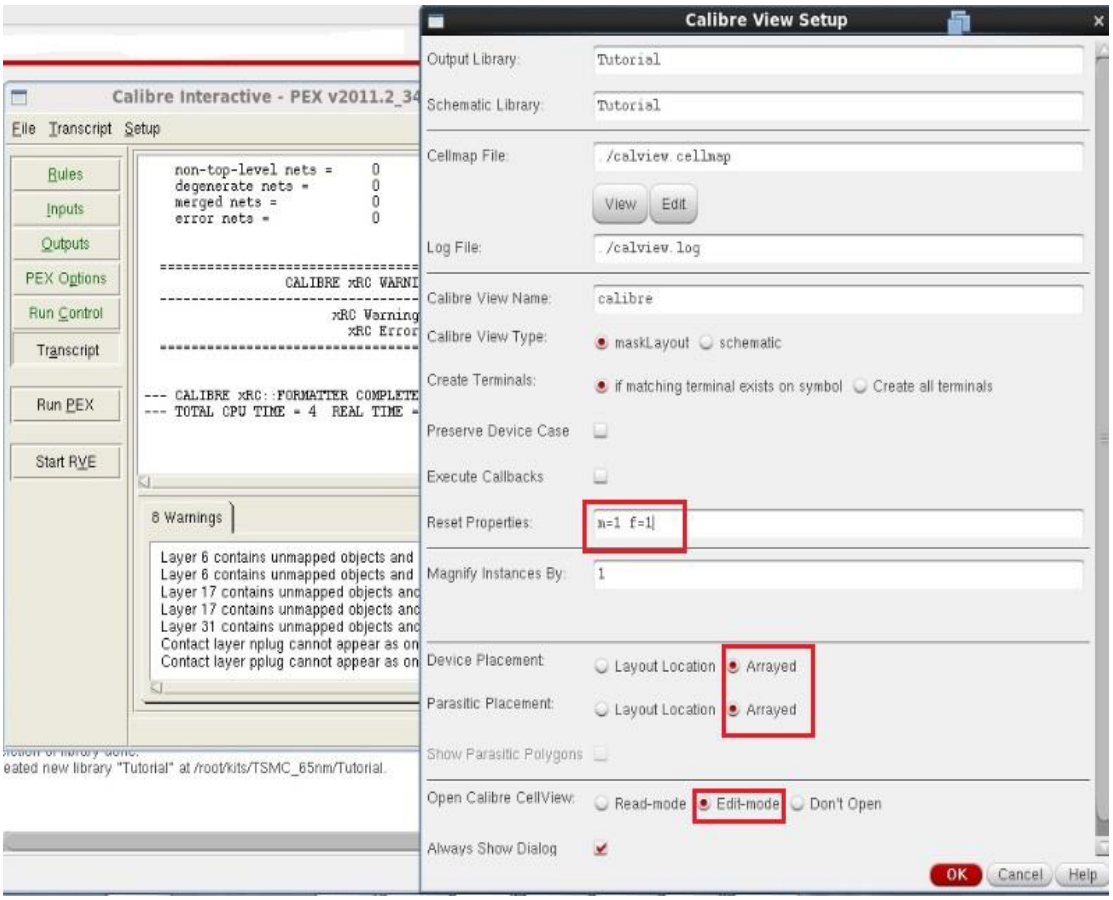
If your connections were right, you get this smiley face as shown:



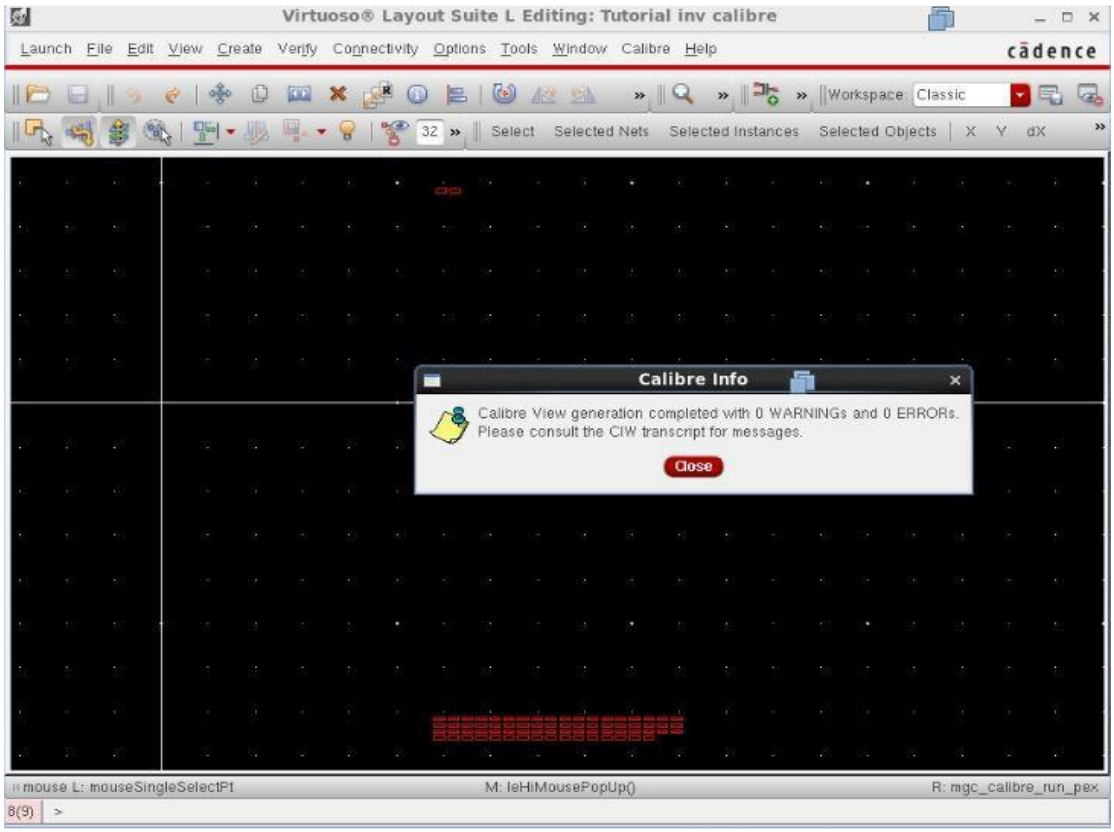
- Third, PEX is done as shown figure:



After running the PEX, you get the following window:

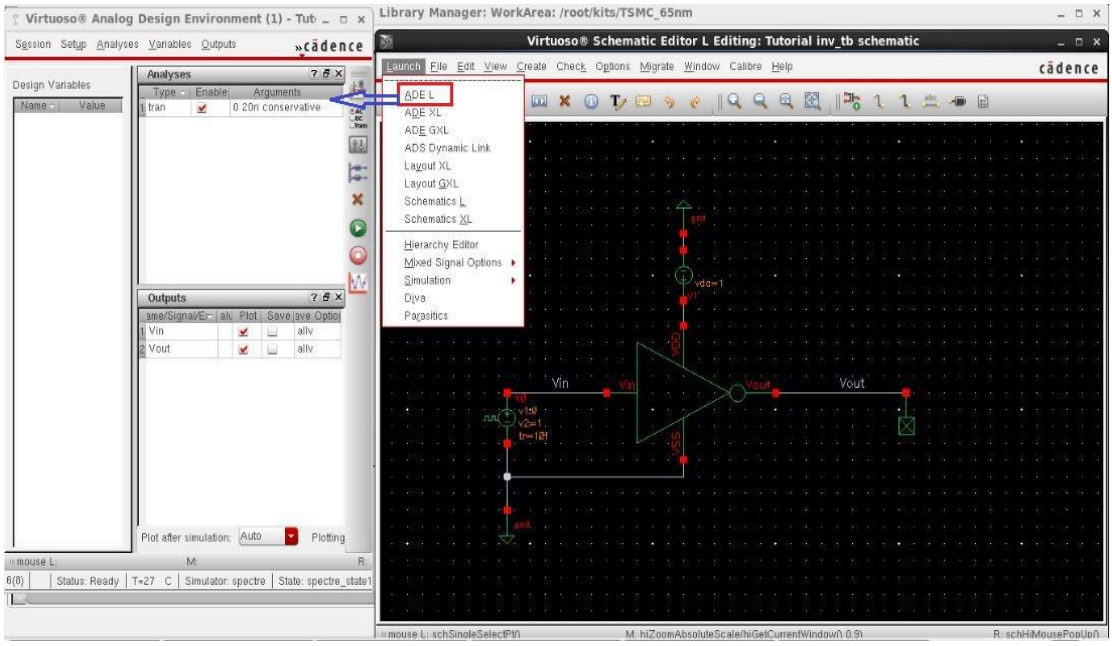


After PEX finish its work, you get this shown figure:

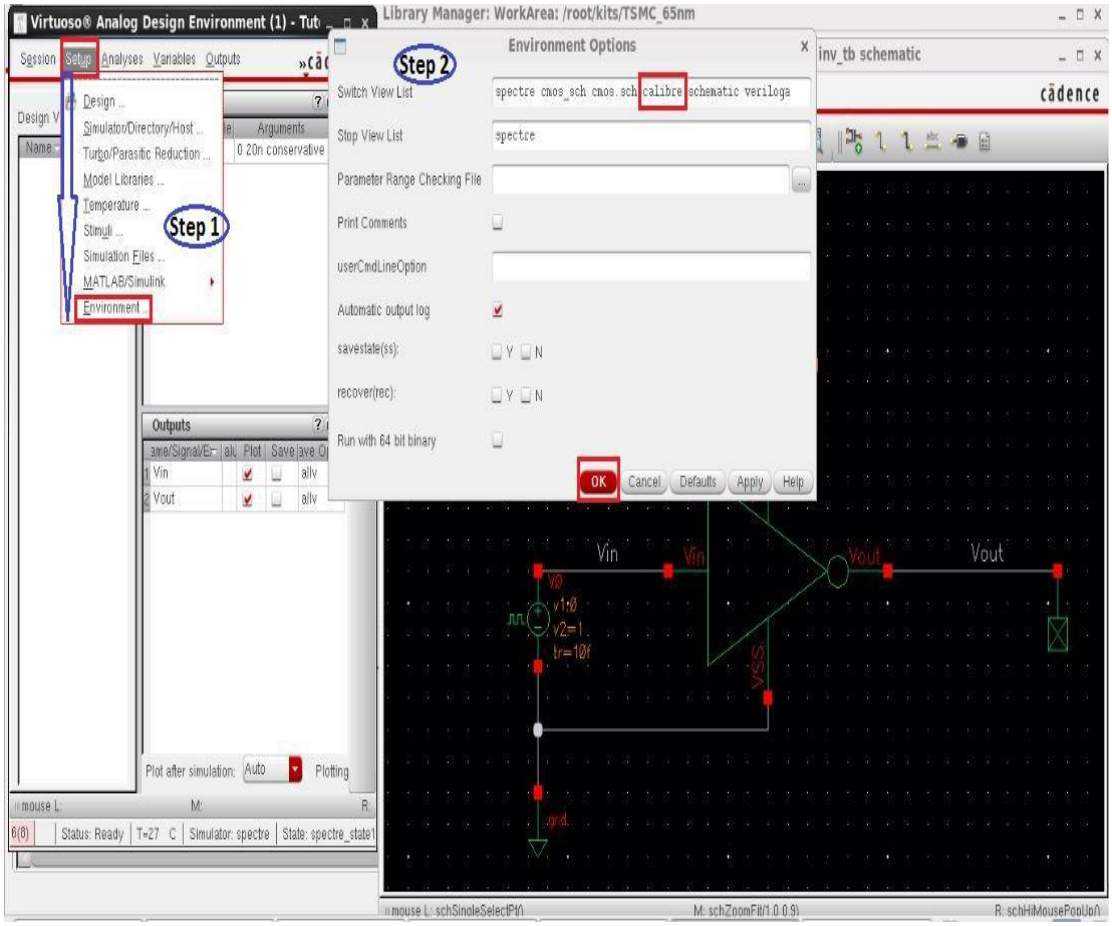


13- Now, We need to simulate parasitic components extracted from drawn layout to know its effect on the circuit functionality:

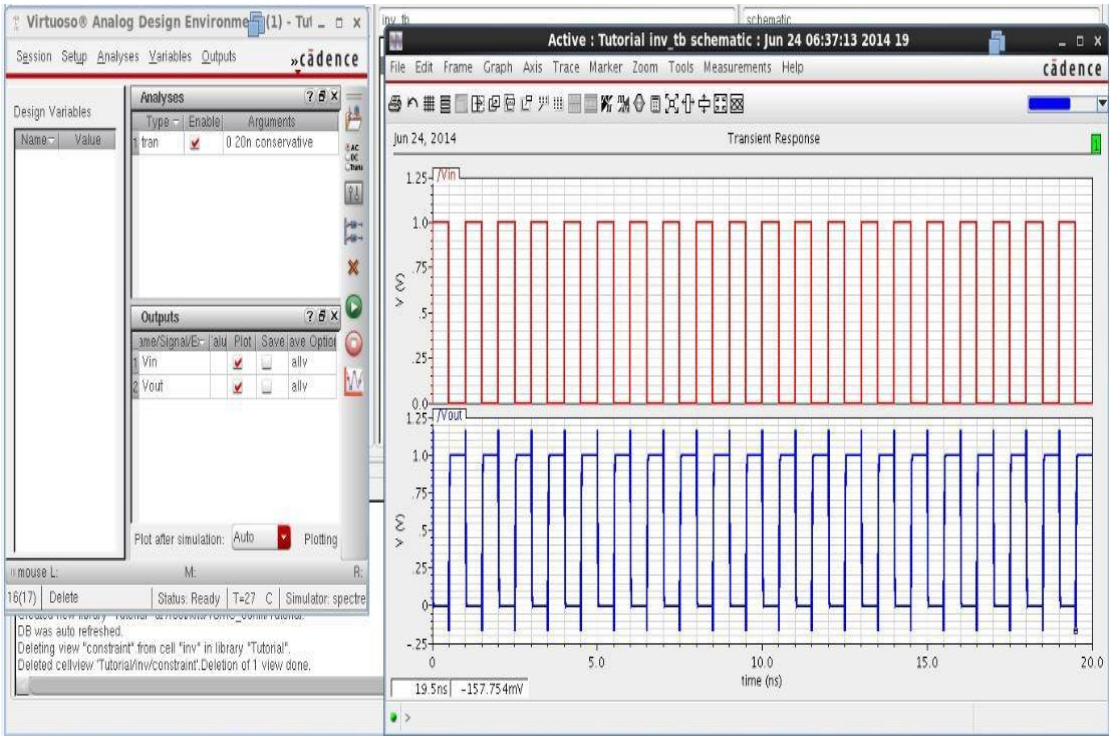
We assume, that you have drawn a testbench file before to test functionality, now we will test a file called calibre view as shown:



Then, edit options as shown:



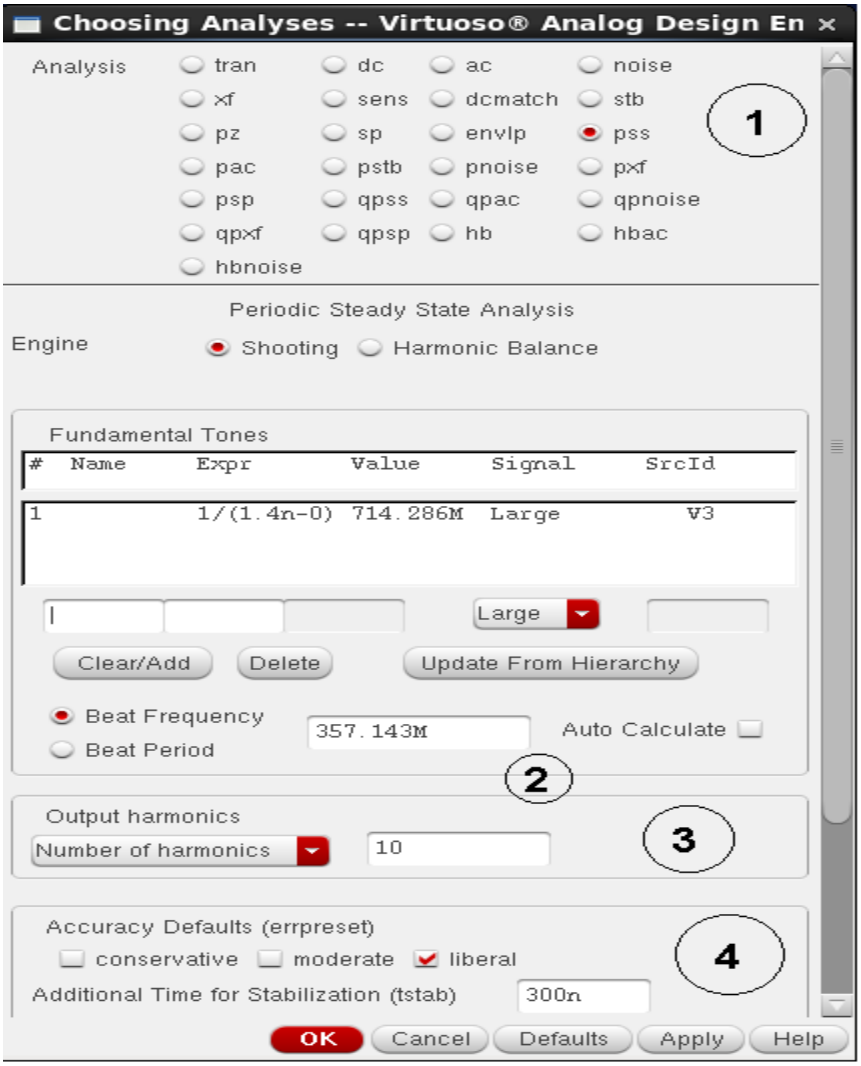
Finally, we run simulation and get shown figure:



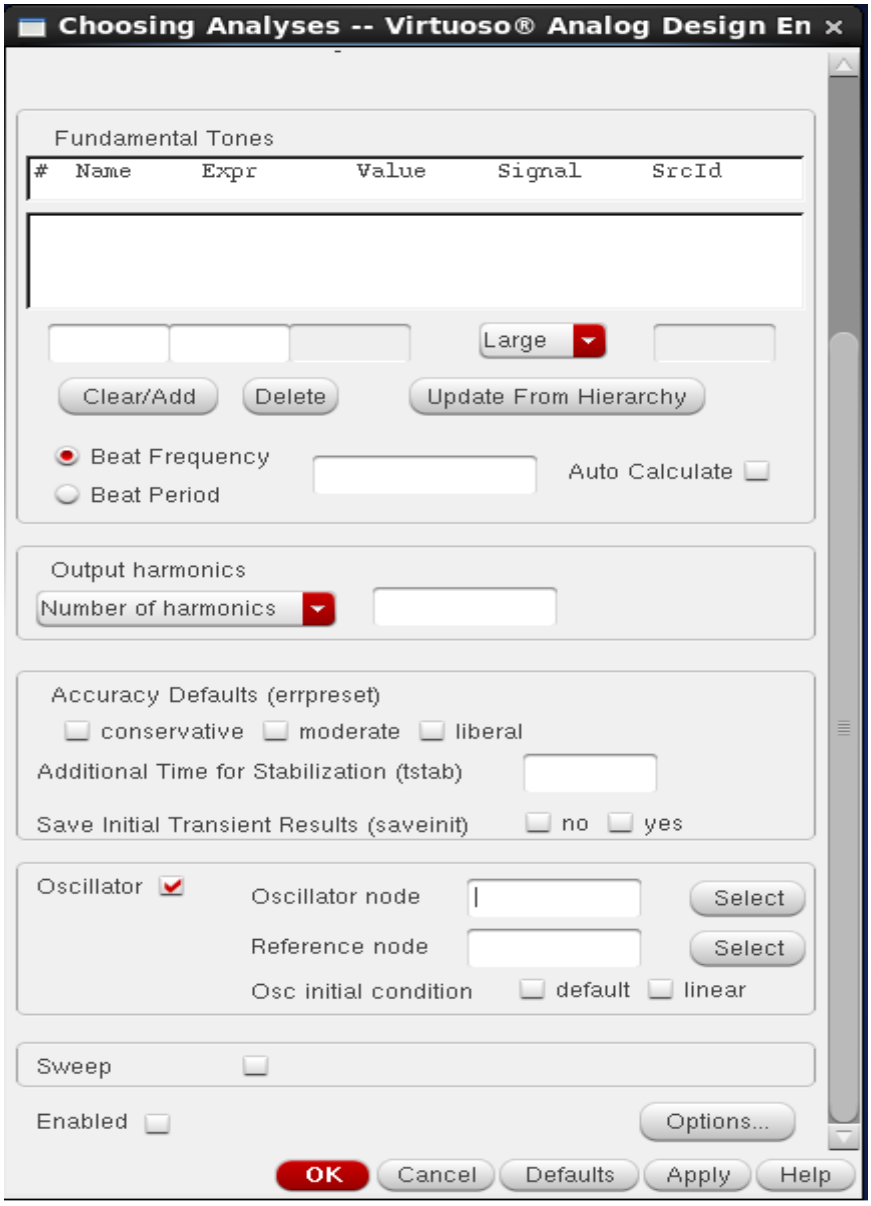
2- Jitter calculation

- PSS analysis for driven circuits, “Driven circuit”: is the circuit, which needs periodic signal to make it work. e.g. PLL needs input periodic signal to lock on its frequency.

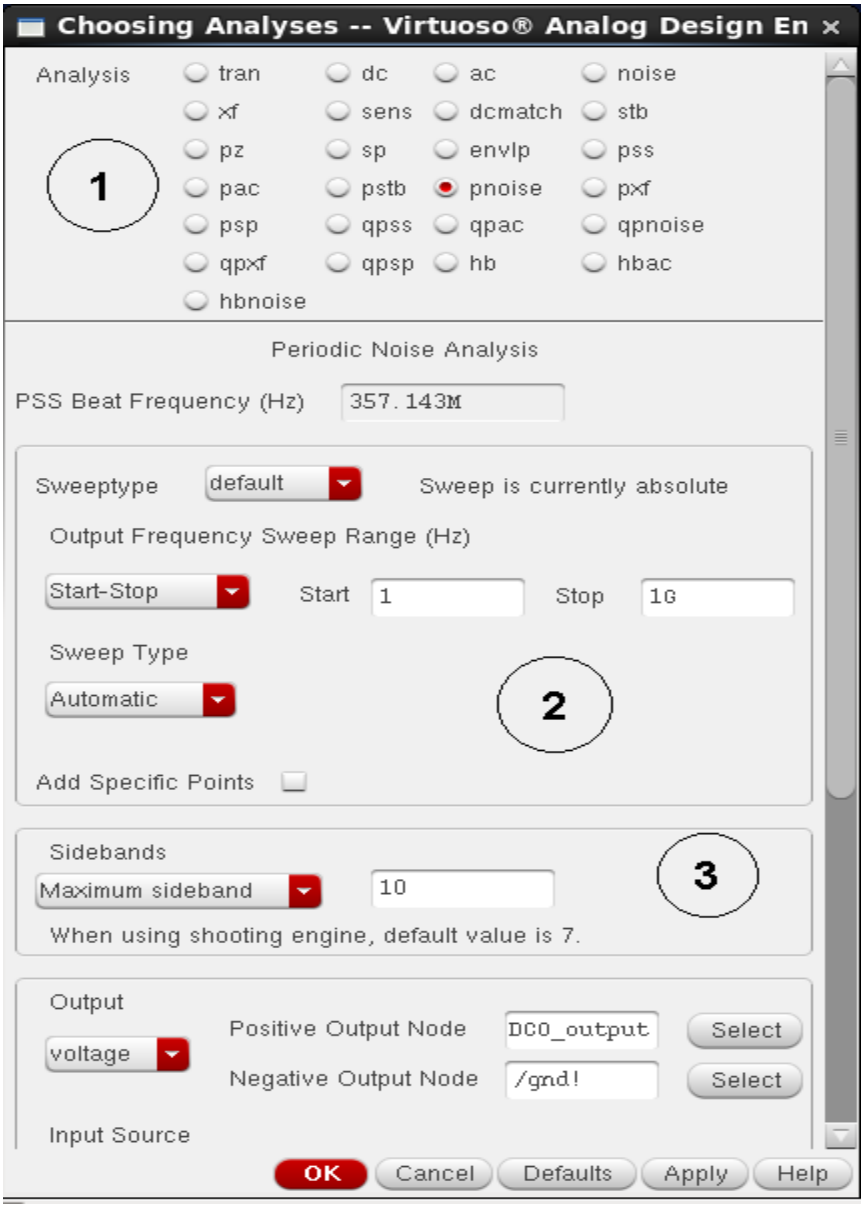
1. Select PSS.
2. In the fundamental tones, there will be the frequencies which are used in the circuit. If there is only one frequency, beat frequency should be half of the fundamental frequency. If there are more than one frequency, beat frequency must be selected to make each frequency as an integer multiple of the beat frequency (EX: if fundamental tones = 500MHz and 600MHz, beat frequency should be 100 MHz).
3. Choose number of harmonics. To increase the calculation accuracy, increase it.
4. Select the additional time for stabilization.



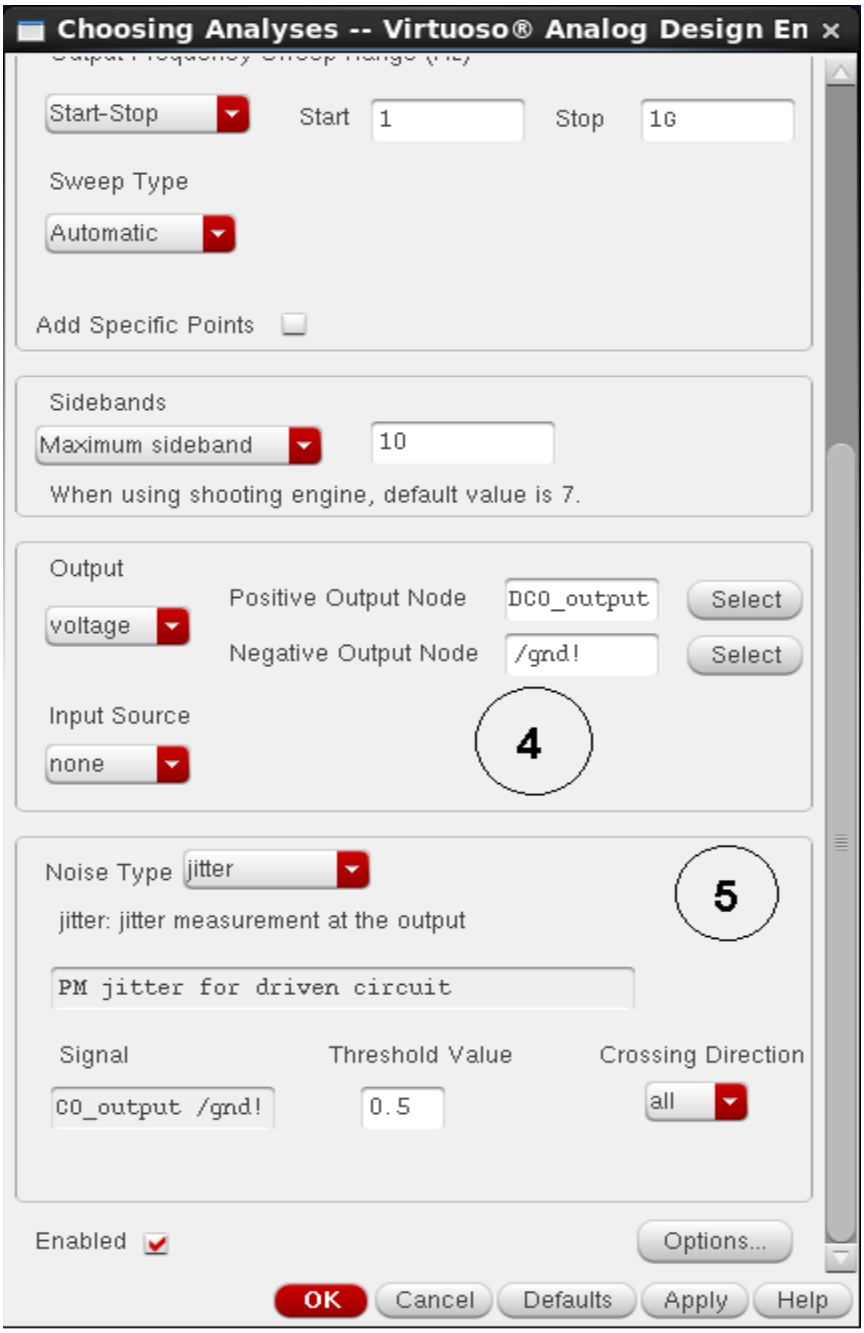
- PSS analysis for oscillators.
  1. Select PSS.
  2. Beat frequency should be the oscillation frequency.
  3. Select number of harmonics.
  4. Select accuracy defaults and the additional time for stabilization.
  5. Choose oscillator, select the node of the output of the oscillation for oscillator node, and select ground node for the reference node.



- Pnoise analysis.
    1. Select pnoise.
    2. Select the output frequency sweep range.
    3. Select maximum sideband. To increase the accuracy, increase it.



4. Select voltage for output. Select the output node for positive output node and ground node for negative output node. For input source, select none.
5. Choose jitter for noise type. And the threshold value should be half of the output range.



Output:

1. Run the simulation.
2. Select Results > Direct plot > Main Form.
3. Select pss for analysis, voltage for function, time for sweep and select the output node.

Select pnoise jitter for analysis, Jcc for function, 1 for number of cycles, RMS for signal level to calculate the RMS jitter or peak to peak for signal level to calculate the peak-to-peak jitter, second for modifier, 1 for frequency multiplier, choose integration limits for phase noise and select plot. The jitter calculation will be at the output graph, which created in step 3.

To communicate with the Team members:

Ayman Nabil Mohamed	ayman_n92@yahoo.com
Basma Mourad Mohamed	ba.soma_91@yahoo.com
Ehab Mahmoud Helmy	ehabhelmy7777@gmail.com
Hany Mohamed Amin	hanymohamin@gmail.com
Ingy Abdelhamid Mohamed	angie_sj_electro@hotmail.com